Hewlett Packard
Enterprise

HPE Security ArcSight Data Platform Event Broker

Software Version: 1.0

Administrator's Guide

September 27, 2016

Legal Notices

Warranty

The only warranties for Hewlett Packard Enterprise 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. Hewlett Packard Enterprise shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

The network information used in the examples in this document (including IP addresses and hostnames) is for illustration purposes only.

HPE Security ArcSight products are highly flexible and function as you configure them. The accessibility, integrity, and confidentiality of your data is your responsibility. Implement a comprehensive security strategy and follow good security practices.

This document is confidential.

Restricted Rights Legend

Confidential computer software. Valid license from Hewlett Packard Enterprise required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notice

Follow this link to see a complete statement of copyrights and acknowledgements:
https://www.protect724.hpe.com/docs/DOC-13026

Support

Contact Information

Phone	A list of phone numbers is available on the HPE Security ArcSight Technical Support Page: https://softwaresupport.hpe.com/documents/10180/14684/esp-support-contact-list
Support Web Site	https://softwaresupport.hpe.com
Protect 724 Community	https://www.protect724.hpe.com

Contents

Chapter 1: Overview

The ArcSight Data Platform Event Broker (ADP Event Broker) centralizes event processing, helps you to scale your ArcSight environment, and opens up ArcSight events to third-party solutions. It enables you to take advantage of scalable, high-throughput, multi-broker clusters for publishing and subscribing to event data.

This chapter includes the following topics:

About ADP Event Broker

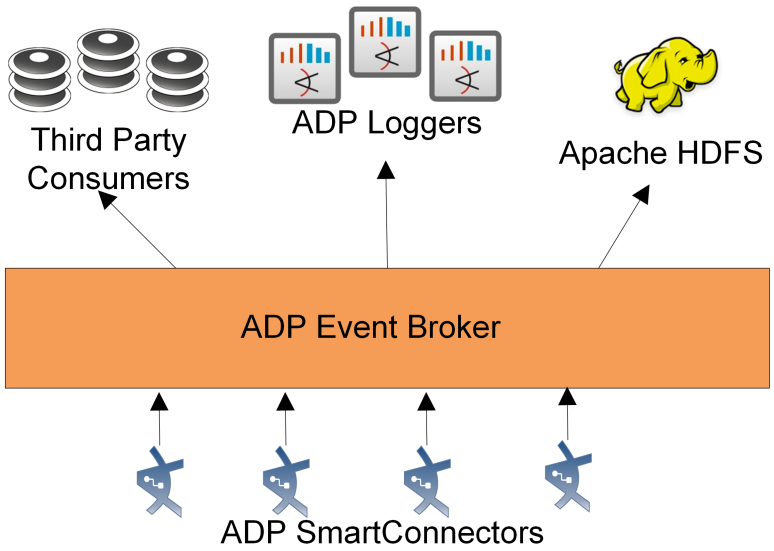
The ADP Event Broker (Event Broker) provides a packaged version of Apache Kafka. After you install and configure an Event Broker Kafka broker or cluster of brokers, you can use ADP SmartConnectors to publish data, and subscribe to that data with ADP Logger, Apache Hadoop, or your own consumer.

To help reduce the disk requirements and network load, events are compressed before being sent to Event Broker. They remain compressed until they reach the consumers.

Events are retained for however long you specify, up to the capacity of the underlying disks. If the disks aren't large enough to hold events to the configured duration, the oldest events are deleted without regard to whether they have been received by any or all consumers and new events are retained.

Event Broker Architecture

ArcSight SmartConnectors are the producers that publish data to the ADP Event Broker. Once the data is in Event Broker, you can subscribe to it with ADP Logger, Apache HDFS, or your own third-party consumer.



Setting up the ADP Event Broker in your Environment

Overview of how to set up ADP Event Broker (Event Broker) in your environment:

1. Consider your overall architecture:
   - You will need some ArcSight SmartConnectors to produce data for Event Broker. For more information, see "Producing Events with SmartConnectors" on page 32.
   - You will need at least one consumer (ADP Logger, Apache Hadoop or other third-party consumer). For more information, see "Consuming Events with Logger" on page 33 and "Consuming Events with Non-ArcSight Applications" on page 35.
   - For production environments, HPE ArcSight recommends that you set up at least three identical server nodes for Event Broker. For more information, see "System Requirements" on page 9.
   - DNS must be properly configured on all Event Broker nodes, and on all consumers and producers.

2. Install and configure Event Broker:
   - Deploy the appropriate hardware for Event Broker. For more information, see "System Requirements" on page 9.
   - Download and verify the Event Broker installation files. For more information, see "Before You Begin" on page 10.
   - Meet the installation prerequisites. For more information, see "Prerequisites for Installation" on page 11.
   - Install the Event Broker software on all Event Broker nodes. For more information, see "Installing Event Broker in GUI Mode" on page 12.
   - Configure Kafka and ZooKeeper on all Event Broker nodes. For more information, see "Configuring ZooKeeper and Kafka" on page 16.
   - Optionally, configure the Event Broker services on all Event Broker nodes. For more information, see "Managing Event Broker Services" on page 22.
   - Start Kafka and ZooKeeper on all Event Broker nodes. See "Starting and Stopping Event Broker" on page 24.
   - Configure Topics on any one node. For more information, see "Managing Kafka Topics in Event Broker" on page 25.
3. Set up your producers:
   - Deploy at least one SmartConnector SmartConnector (7.3.0 or higher) to send data to Event Broker.
   - Set up the SmartConnector(s) to publish to Event Broker.

   For more information, see "Producing Events with SmartConnectors" on page 32.
4. Set up your consumers:
   - Deploy at least one Logger (6.3 or higher) , Hadoop, or third-party consumer.
   - Set up your consumer(s) to receive events from Event Broker's Kafka cluster.

   For more information, see "Consuming Events with Logger" on page 33, "Consuming Event Broker Events with Apache Hadoop" on page 36, or "Consuming Events with Non-ArcSight Applications" on page 35.
5. Secure your environment:
   - Event Broker supports disk encryption and TLS. For more information, see "Securing Your Event Broker Deployment" on page 59.

Chapter 2: Installing Event Broker

The Event Broker installer will deploy Kafka, ZooKeeper, and the Event Broker Manager on your server. You can run the installer with a Command Line Interface (CLI) or a Graphical User Interface (GUI).

This chapter includes the following topics:

System Requirements

Before you can install Event Broker, you must set up the appropriate hardware. You will need to set up an odd number of servers for quorum-based replication. HPE ArcSight recommends that you set up at least three, with more for a large installation. Each will need to meet the requirements below.

Disk capacity and throughput requirements depend on average event size and compression ratio, as well as peak and average EPS. Large deployments could potentially require terabytes of storage per day and have hundreds of megabytes per second of disk throughput per node.

ZooKeeper requires low latency access and writes quite frequently. Ideally, this should be a solid state device, or at least a different physical disk than Kafka will use.

You can calculate the average bytes per day by using the following formula:

{average event size} x {average event rate}/{compression factor}= average bytes per second

{average bytes per second} x 86,400 = average bytes per day

For example, if you use 3 as a conservative compression factor, with 1000 byte average uncompressed event size at an average of 10,000 events per second over the course of a day, it would be (1000 bytes/event/3) * (10000 events/second) * (86400 seconds/day) / (1 gigabyte/10^9 bytes) = 288 GB/day of events.

With a replication factor of two, spread between three nodes, each node requires 192 GB/day of disk space. For further discussion and recommendations on server, disk and filesystem implementation, refer to https://kafka.apache.org/0100/ops.html.

Make sure you meet the following requirements:

- Minimum 10-gigabit Ethernet, with full-speed interconnects between all nodes in the cluster. These must be reachable from all consumers and producers.
- All machines involved in the Event Broker (nodes, consumers, producers) must have forward and reverse DNS entries.
- For each server node (minimum 3):
  - 8-core 64-bit server-grade processor
  - 32 GB RAM
  - 8 or more TB of disk space, depending on how long data should be retained and expected throughput, sourced from at least 2 disks, using either hardware RAID or LVM to create a JBOD or striped array. (Disk redundancy with RAID is not required since redundancy is provided by storing events on multiple nodes.)

    Note: To get good throughput, use multiple drives. To ensure good latency, do not share the same drives used for Kafka data with application logs or other OS filesystem activity.
- You must install on a supported OS platform. For information about the platforms on which you can install Event Broker, refer to the *ArcSight Data Platform Event Broker Release Notes* and *ArcSight Data Platform Support Matrix*. These documents are available for download from the ArcSight Product Documentation Community on Protect 724.

Before You Begin

The topics below describe how to get and verify the Event Broker installation package.

Downloading the Installation Package

The installation package is available for download from the HPE Software Depot.

Visit the following site to download your installation package:

https://h20392.www2.hpe.com/portal/swdepot/index.do

Verifying the Downloaded Installation Software

HPE provides a digital public key to enable you to verify that the signed software you received is indeed from HPE and has not been manipulated in any way by a third party.

Visit the following site for information and instructions:

https://h20392.www2.hpe.com/portal/swdepot/displayProductInfo.do?productNumber=HPLinuxCodeSigning

Prerequisites for Installation

Make sure these prerequisites are met before you install Event Broker:

- Ensure that you have met the "System Requirements" on page 9.
- Ensure that you are installing on a supported platform. Refer to the *ArcSight Data Platform Support Matrix*. This document is available for download from the ArcSight Product Documentation Community on Protect 724.
- Increase the user process limit, as described in "Increasing the User Process Limit and the Maximum Number of Open Files" below.
- HPE ArcSight recommends that you install as a non-root user. If none exists, create one.
- The directory you are installing into and the data storage directories must be writable by the user performing the installation.
- To use the GUI mode of installation, you must have an X Window System or similar GUI viewer installed on your server. If one is not installed, the installer will open in console mode.
  - If you will be installing Event Broker over an SSH connection and want to use the GUI mode of installation, make sure that you have enabled X window forwarding using the -X option so that you can view the screens of the installation wizard.
  - If you will be using PuTTY, you will also need an X client on the machine from which you are connecting to the machine onto which you want to install Event Broker.

Increasing the User Process Limit and the Maximum Number of Open Files

Before installing Event Broker on each node, you must increase the default user process limit while logged in as user `root`. This ensures that the system has adequate processing capacity.

To increase the default user process limit:

1. Open the file `/etc/security/limits.d/<NN>-nproc.conf`.
   (`<NN>` is 90 for RHEL or CentOS 6.X and 20 for RHEL and CentOS 7.X.)
   - If you do not already have a `/etc/security/limits.d/<NN>-nproc.conf file`, create one.
   - If the file already exists, delete all entries in the file.
2. Add the following lines:

```
* soft nproc 10240
* hard nproc 10240
* soft nofile 65536
* hard nofile 65536
```

Caution: Be sure to include the asterisk (*) in the new entries. It is important that you add all of the entries exactly as specified. Any omissions can cause system run-time errors.

3. Log out and log back in again.
4. Run the following command to verify the new settings: `ulimit -a`
5. Verify that the output shows the following values for “open files” and “max user processes”:

   ```
   open files 65536
   max user processes 10240
   ```

After you have increased the user process limit and met the other prerequisites, you are ready to install Event Broker.

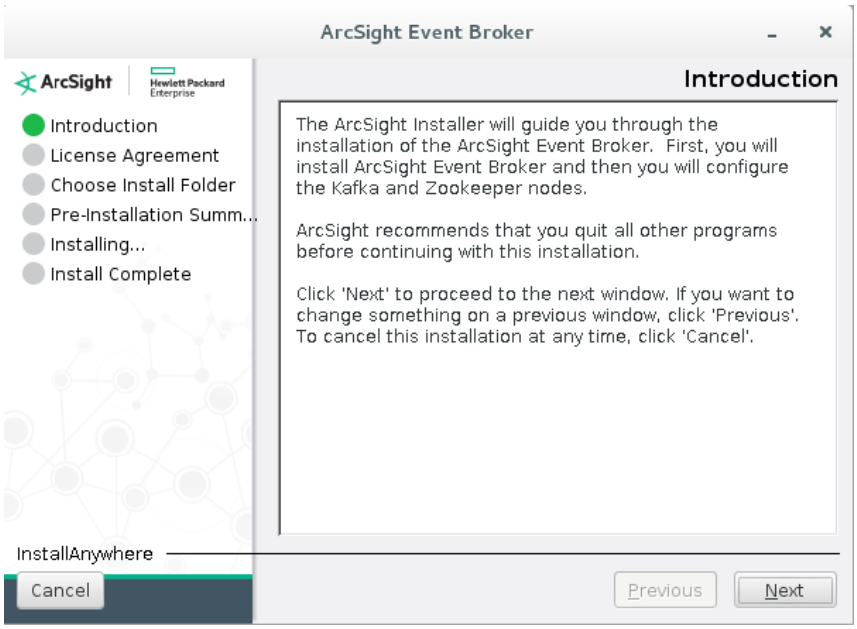
Installing Event Broker in GUI Mode

To install Event Broker in GUI mode:

1. Run these commands from the directory where you copied the Event Broker installation file:

   ```
   chmod +x ArcSightEventBroker-1.0.0.155.0.bin
   ./ArcSightEventBroker-1.0.0.155.0.bin
   ```

2. The installation wizard launches. Click **Next**.

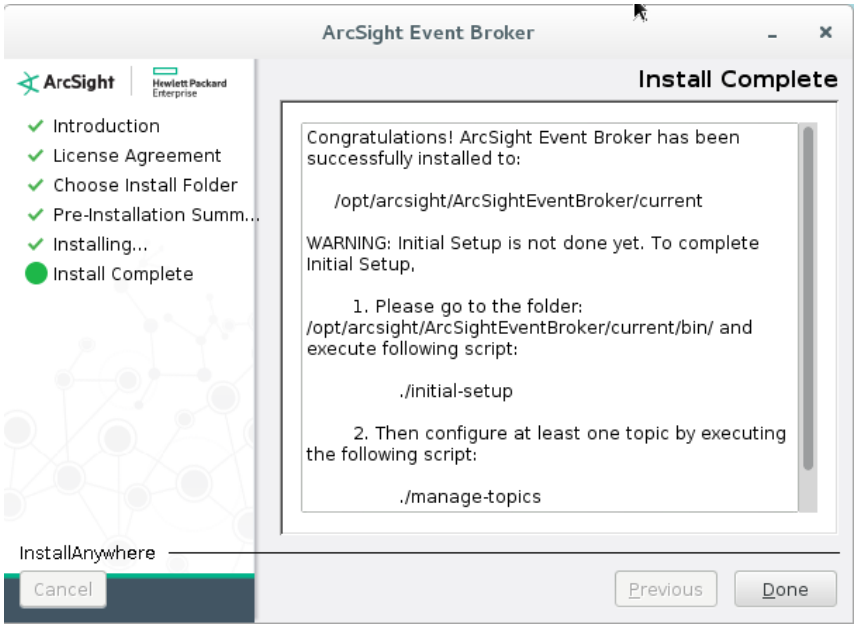


You can click **Cancel** to exit the installer at any point during the installation process.

3. The License Agreement screen is displayed. Scroll to the bottom of the license agreement to review the agreement and enable the "I accept the terms of the License Agreement" button.
4. Select **I accept the terms of the License Agreement** and click **Next**.
5. The installer checks that installation prerequisites are met. If a check fails, Event Broker displays a message. You will need to fix the issue before proceeding.
6. Navigate to or specify the location where you want to install Event Broker.

   The default installation path is `/opt/arcsight/ArcSightEventBroker`. You can install into this location or another location of your choice.
7. Click **Next** to install into the selected location.
   - If there is not enough space to install the software at the location you specified, a message is displayed. To proceed with the installation, specify a different location or make sufficient space available at the location you specified. Click **Previous** to specify another location or **Quit** to exit the installer.
   - If Event Broker is already installed at the location you specify, a message is displayed. Click **Previous** to specify another location or **Quit** to exit the installer.
8. Review the pre-install summary and then click **Install**.

   Installation may take a few minutes. Please wait.
9. Once the installation is complete, the installer displays the name and location of the `initial-setup` and `manage-topic` tools.



   Make a note of these. You will need to run them to complete your setup.
10. Click **Done** to exit the installer.

Next step: After installing Event Broker, you must configure it. See "Configuring ZooKeeper and Kafka" on page 16.

Installing Event Broker in CLI Mode

To install Event Broker in CLI mode:

1. Run these commands from the directory where you copied the Event Broker installation file:

   ```
   chmod +x ArcSightEventBroker-1.0.0.155.0.bin
   ./ArcSightEventBroker-1.0.0.155.0.bin -i console
   ```

2. The installation wizard launches in command-line mode, as shown below. Press **Enter** to continue.

   ```
   Introduction
   ------------

   The ArcSight Installer will guide you through the installation of ArcSight
   Event Broker.

   ArcSight recommends that you quit all programs before continuing with this
   installation.

   Respond to each prompt to proceed to the next step in the installation. If
   you want to change something on a previous step, type 'back'. To cancel
   this installation at any time,  type 'quit'.

   PRESS <ENTER> TO CONTINUE:
   ```

3. The next several screens display the end user license agreement. Press Enter to display each part of the license agreement, until you reach the following prompt:

   ```
   DO YOU ACCEPT THE TERMS OF THIS LICENSE AGREEMENT? (Y/N):
   ```

4. Type Y and press **Enter** to accept the terms of the License Agreement.

   You can type `quit` and press **Enter** to exit the installer at any point during the installation process.

5. The installer checks that installation prerequisites are met. If a check fails, Event Broker displays a message. You will need to fix the issue before proceeding.

   Once all checks complete, the installation continues, and the Choose Install Folder screen is displayed.

6. The default installation path is `/opt/arcsight/ArcSightEventBroker`. You can press **Enter** to install in this location or enter another location of your choice and then press **Enter**.

7. If there is not enough space to install the software at the location you specified, a message is displayed.

- To proceed with the installation, specify a different location or make sufficient space available at the location you specified. Type **quit** and press **Enter** to exit the installer.
- If Event Broker is already installed at the location you specify, a message is displayed. Type **quit** and press **Enter** to exit the installer or type **back** and press **Enter** to specify another location and uninstall the previous version.

8. Review the pre-install summary and press **Enter** to install Event Broker.

   Installation may take a few minutes. Please wait.
9. Once the installation is complete, the installer displays the name and location of the `initial-setup` and `manage-topic` tools. Make a note of these. You will need to run them to complete your setup.
10. Press Enter to exit the installer.

Next Step: After installing Event Broker, you must configure it. See "Configuring ZooKeeper and Kafka" on page 16.

Uninstalling Event Broker

Uninstalling Event Broker does not delete files created after installation, including your event data. If you configured Event Broker services, remove them before uninstalling.

To remove the Event Broker services:

Run the following command from the *`install_dir`*`/current` directory:

```
./bin/manage-service remove
```

To uninstall Event Broker:

Run the following command from the *`install_dir`*`/current` directory:

```
./UninstallerData/Uninstall_ArcSight_EventBroker
```

The uninstall wizard launches. Click **Uninstall** or press **Enter** to start uninstalling Event Broker.

Chapter 3: Configuring Event Broker

The Event Broker installer installs Apache Kafka and Apache ZooKeeper on your system. To use Event Broker, you must configure ZooKeeper ensemble and a Kafka cluster. For more information about Kafka and ZooKeeper, refer to the Apache Kafka documentation.

The Event Broker Kafka cluster must have an odd number of nodes. HPE ArcSight recommends using three or more in production environments. ZooKeeper runs on these same nodes. The nodes should be dedicated to the Event Broker, as high throughput and low latency are required.

This chapter includes the following topics:

Configuration Tools

The ADP Event Broker provides several tools that you can use to manage your Kafka cluster and topics. These, and other Event Broker tools are located in *install_dir*`/current/bin`.

- **initial-setup**: You must run this tool as part of the installation process to configure ZooKeeper and Kafka on your system. Optionally, you can record the configuration and use the autoconfig tool to copy it to other nodes. For more information, see "Configuring ZooKeeper and Kafka" below.
- **manage-topics**: You must run this tool as part of the installation process to configure at least one topic for your cluster. You can run it again as needed to manage your topics. For more information, see "Managing Kafka Topics in Event Broker" on page 25.
- **autoconfig**: You can use this tool to copy the configuration of one node, recorded by the `initial-setup` tool, to other nodes in your cluster. For more information, see "Auto-Configuring Event Broker Nodes" on page 21.
- **manage-service**: You can use this tool to set up ZooKeeper and Kafka to run as services. For more information, see "Managing Event Broker Services" on page 22.

Configuring ZooKeeper and Kafka

The ZooKeeper and Kafka configuration in your deployment is controlled through configuration files.

- `user/config/kafka.properties`
- `user/config/zookeeper.properties`

The user you installed Event Broker with must have write permissions on these directories.

This section includes the following topics:

Initial Configuration

Immediately after installation, run the `initial-setup` tool to configure ZooKeeper and Kafka. This tool will create directories for ZooKeeper and Kafka. If you need to update the configuration, see "Updating the Configuration" on page 20.

To configure ZooKeeper and Kafka:

1. Run the following command from the *`install_dir`*`/current` directory:

   `./bin/initial-setup`

   Alternatively, for proof-of-concept demos only, run the following command:

   `ALLOW_SINGLE_NODE=poc ./bin/initial-setup`

   Caution: Do not use a single node in a production environment. Single node setup has no redundancy. If the system goes down, all data that has not been consumed yet will be lost.

2. The tool guides you through configuring a ZooKeeper ensemble and Kafka cluster. Follow the prompts to complete the configuration.

Prompt	What to enter
Preparing for new ZooKeeper configuration (step 1)	
On how many servers will ZooKeeper and Kafka be running?	Enter the number of nodes there will be in this cluster. Valid values: An odd number. HPE ArcSight recommends that you use 3 or greater. **Note:** Do not use an even number of nodes. Example: 3
Each node in the cluster needs a unique positive integer to identify itself. These start at 1 and go up to the number of nodes in the cluster (in this case, 3). Which node is this?	Enter the identifying number of the node you are currently configuring. Valid values: A number that identifies one of the nodes of the cluster. This tool will configure this node only. You must run the tool once for every node in the cluster. Example: 1

Prompt	What to enter

ZooKeeper needs to know where every other ZooKeeper server is located. These addresses must be specified in the same order on each server. They may be IP addresses or hostnames that can be resolved globally.
Address of server 1:
Address of server 2:
Address of server 3:
....
Address of server <N>:
	Enter the hostname or IP address of the server for each node.
	Valid values: A valid hostname or IP address
	Example:
	`eventbroker1.example.com`
	`eventbroker2.example.com`
	`eventbroker3.example.com`
	Enter more hosts if you specified a number greater than 3.

This machine has <NN> megabytes of RAM. Based on that, we recommend that ZooKeeper uses <nn> megabytes. Use this value? [Y/n]	Enter y to use the suggested amount of RAM or enter n to specify another amount. **Tip:** The upper case letter signifies the default, rather than the case of the letter you should type. Simply press **Enter** to use the default value. If you enter n, you are asked to specify an amount of usable RAM in megabytes. Example: `740`
Please review the following information regarding ZooKeeper configuration Number of nodes: 3 This is node number: 1 Node 1 address: eventbroker1.example.com Node 2 address: eventbroker2.example.com Node 3 address: eventbroker3.example.com Node <N> address: ... ZooKeeper memory in megabytes: 740 Is the above information correct? [y/N]	Enter n to change the configuration, or enter y accept the configuration and proceed.
Preparing for new ZooKeeper configuration (step 2)	
ZooKeeper only requires about 1 gigabyte of disk space on a fast disk. Enter the location on disk where ZooKeeper should store data:	Enter the location where ZooKeeper should store data. Valid values: A directory on the file system. The user you installed Event Broker with must have write permissions on this location. Use a different physical location than the one on which you plan to store Kafka data. Example: `/ArcSight/ZooKeeperData`

Prompt	What to enter
Please review the following information regarding ZooKeeper configuration: /ArcSight/ZooKeeperData Is all of the above information correct? [y/N]	Enter n to change the configuration, or enter y accept the configuration and proceed.
Preparing for new Kafka configuration	
Kafka requires a large amount of space on fast disks. This location should be different than the physical medium you specified for ZooKeeper. Enter the location on disk where Kafka should store data.	Enter the location where Kafka should store data. Valid values: A directory on the file system. The user you installed Event Broker with must have write permissions on this location. Example: `/ArcSight/KafkaData`
Kafka retains data for a set amount of time, or until a set amount of disk space is consumed, whichever occurs first. It then starts deleting the oldest data. Enter the maximum number of days to retain data.	Enter a number of days. Valid values: Any integer greater than 0. Example: 3
Set this correctly even if you want to dedicate the disk to Kafka. Leave at least 1 GB of space available on the disk. Enter the maximum amount of disk space to use (in GigaBytes):	Enter the maximum disk space to use for Kafka data retention. Be sure to leave at least one Gigabyte unused. Valid values: A number of GB (Maximum total disk space minus at least 1 GB.) Example: For a 600 GB disk, enter 599 or less.
This machine has <NN> megabytes of RAM. Based on that, it is recommended that Kafka uses <nn> megabytes. Use this value? [Y/n]	Enter y to use the suggested amount of RAM or enter n to specify another amount. If you enter n, you are asked to specify an amount of available RAM in megabytes. Example: 5927
Please review the following information regarding Kafka configuration: Kafka data directory: /ArcSight/KafkaData Data retention time: 3 days (72 hours) Data retention size: 599 gigabytes (643171352576 bytes) Kafka memory: 5927 megabytes Is all of the above information correct? [y/N]	Enter n to change the configuration, or enter y accept the configuration and proceed.
Do you want to create a configuration bundle to copy to other nodes in the cluster? [y/N]	Enter y to create a configuration bundle that you can use to configure ZooKeeper and Kafka on other Event Broker nodes.

Prompt	What to enter
Copy this file to the same location on the other nodes in the cluster, then run *install_dir*/current/bin/autoconfig to automatically configure those nodes to use the same values as this node.	• If you entered n, the setup tool configures ZooKeeper and Kafka as specified, and then exits. • If you entered y, the setup tool creates the configuration bundle, *install_dir*/current/user/config.tgz, configures ZooKeeper and Kafka as specified, and then exits. You can use the configuration bundle to automatically configure the rest of your nodes. See "Auto-Configuring Event Broker Nodes" on the next page.
ZooKeeper and Kafka have been configured. If you want to run them as a service, on reboot, then run as root: *install_dir*/current/bin/manage-service install Alternatively you can use the tools provided in *install_dir*/current/bin to run them on demand.	After this tool exits, you can set up Event Broker's ZooKeeper and Kafka to run as services. See "Managing Event Broker Services" on page 22. Start and stop tools are also provided. See "Starting and Stopping Event Broker" on page 24.
After running this tool and starting Kafka and ZooKeeper on every node in your cluster, run *install_dir*/current/bin/manage-topics to configure topics from any single node in the cluster.	After running this tool and starting Kafka and ZooKeeper on every node in your cluster, the next step is to configure topics for your cluster. See "Managing Kafka Topics in Event Broker" on page 25.

Updating the Configuration

You can update the configuration files manually or by running the `initial-setup` tool again.

You must perform these steps on each Event Broker node in your cluster. To keeping your cluster online and operational when updating your configuration, perform a rolling restart. Stop Kafka and ZooKeeper on one node, make the changes, restart Kafka and ZooKeeper. Do same on each node, until you have made the changes on all.

Caution: Never stop more than 50% of your ZooKeepers and Kafkas at the same time.

To update the files by rerunning the initial-setup tool:

1. Stop Kafka and ZooKeeper. For instructions, see "Starting and Stopping Event Broker" on page 24.
2. Delete the following files.
   - `user/config/kafka.properties`
   - `user/config/zookeeper.properties`
3. Run the `initial-setup` tool, as described in "Initial Configuration" on page 17.

4. Restart Kafka and ZooKeeper.
5. Repeat these steps on each Event Broker node in your cluster.

To update the configuration files manually:

1. Stop Kafka and ZooKeeper. For instructions, see "Starting and Stopping Event Broker" on page 24.
2. Edit the following files to make the appropriate changes.
   - `user/config/kafka.properties`
   - `user/config/zookeeper.properties`
3. Start Kafka and ZooKeeper.
4. Repeat these steps on each Event Broker node in your cluster.

Auto-Configuring Event Broker Nodes

If you created a configuration bundle when you set up your first Event Broker node, you can run the `autoconfig` tool to set up the same ZooKeeper and Kafka configuration on each of the other nodes in your cluster.

This tool cannot change an existing configuration. If you want replace the current configuration with an automatic configuration, delete the following files and run the tool.

- `user/config/kafka.properties`
- `user/config/zookeeper.properties`

Prerequisites:

- Event Broker must already be installed on the current node.
- Hardware specifications for each node must be identical to that of the original node, including:
  - Storage locations
  - Storage capacity
  - Memory capacity
  - Disk mount locations
- The user running the tool must have write permissions for the directory the tool will create.
- A configuration bundle, `config.tgz`, must have been created during initial setup of the original node, and you must have access to this bundle.

To automatically configure ZooKeeper and Kafka on other nodes:

1. Copy the configuration bundle from *`install_dir`*`/current/user/config.tgz` on the original node to *`install_dir`*`/current/user/config.tgz` on this node.

2. Run the following command from the *install_dir*/current directory:
   `./bin/autoconfig`
3. The `autoconfig` tool opens. Respond to the prompts to configure the node.

Prompt	What to enter
The configuration bundle has the following node ID to address mapping: 1: eventbroker1.example.com 2: eventbroker2.example.com 3: eventbroker3.example.com This node's hostname is eventbroker2.example.com Each node in the cluster needs a unique positive integer to identify itself. These must start at 1 and go up to the number of nodes in the cluster (in this case, 3). Which node is this?	Enter the number that identifies the current host. It should be one of the hosts in the list above. Example: 2
ZooKeeper and Kafka have been configured. If you want to run them as a service on reboot, run, as root: *install_dir*/current/bin/manage-service install Alternatively, you can run them on demand by using the start and stop scripts provided in *install_dir*/current/bin.	After this tool exits, you can set up Event Broker's ZooKeeper and Kafka to run as services. See "Managing Event Broker Services" below. Start and stop tools are also provided. See "Starting and Stopping Event Broker" on page 24.
After configuring and starting the software on every node in your cluster, configure topics from any node in the cluster by running *install_dir*/current/bin/manage-topics.	After running this tool and starting Kafka and ZooKeeper on every node in your cluster, the next step is to configure topics for your cluster. See "Managing Kafka Topics in Event Broker" on page 25 for instructions.

4. Repeat the process for the remaining nodes.

Managing Event Broker Services

You can use the `manage-service` tool to set up the Event Broker services and to uninstall the services.

- Running `manage-service install` installs three services, `arst-kafka`, `arst-kafka-manager`, and `arst-zookeeper`.
- Running `manage-service remove` uninstalls all three services.

To start or stop the services, see "Starting and Stopping Event Broker" on page 24.

To set up the Event Broker services, you must run the `manage-service install` tool as root on each node. You cannot complete the process on any node until you have started it on all nodes.

Prerequisite: ZooKeeper and Kafka must be configured and running before you can use this tool. For more information, see "Configuring ZooKeeper and Kafka" on page 16 and "Starting and Stopping Event Broker" on page 24.

To install the Event Broker services:

1. Log in as the root user.
2. Run the following command:
   `./install_dir/current/bin/manage-service install`
3. Respond to the prompts to install the Event Broker services.

   When the process is complete, the Event Broker services (`arst-kafka`, `arst-kafka-manager`, and `arst-zookeeper`) have been installed and start automatically.

   The Event Broker services are installed.

To remove the Event Broker services:

1. Log in as the root user.
2. Run the following command:
   `./install_dir/current/bin/manage-service remove`
3. Respond to the prompts to remove the Event Broker services.

   When the process is complete, the Event Broker services `arst-kafka`, `arst-kafka-manager`, and `arst-zookeeper`) have been stopped and removed. Configuration files and event data are not deleted.

Chapter 4: Starting and Stopping Event Broker

ArcSight Event Broker provides several start and stop tools. These, and other Event Broker tools are located in *install_dir*/current/bin.

This section documents how to start and stop Event Broker's ZooKeeper and Kafka. For information on how to start and stop Event Broker Manager, see "Starting and Stopping the Event Broker Manager" on page 42.

The way to start and stop Event Broker depends on whether the Event Broker services were configured.

To start or stop the Event Broker services:

Note: These tools must be run as root.

1. Run the appropriate command from the *install_dir*/current directory:
   - service arst-kafka (stop, start, restart)
   - service arst-zookeeper (stop, start, restart)

If the Event Broker services were not configured, use the tools below to start and stop Kafka and ZooKeeper.

To start or stop Kafka:

Note: These tools must be run as the user that installed Event Broker.

1. Log in as the user that installed Event Broker:
2. Run the appropriate command from the *install_dir*/current directory:
   - ./bin/start-kafka
   - ./bin/stop-kafka

To start or stop ZooKeeper:

Note: These tools must be run as the user that installed Event Broker.

1. Log in as the user that installed Event Broker:
2. Run the appropriate command from the *install_dir*/current directory:
   - ./bin/start-zookeeper
   - ./bin/stop-zookeeper

Chapter 5: Managing Kafka Topics in Event Broker

Once Event Broker is configured and running on all nodes in your cluster, you can run the `manage-topics` tool to configure topics from any node in the cluster. You can run it again as needed to view existing topics, create new topics, and increase the number of partitions in managed topics.

This chapter includes the following topics:

Topic Configuration

Each topic can be configured with its own partition count and replication factor, depending on traffic levels and durability requirements. HPE ArcSight recommends using different topics for categorization, for example, firewall events in one topic and anti-virus events in another.

- Configure the number of topics based on data isolation requirements or categorization.
- Configure partition count based on throughput and number of consumers.
- Configure the replication factor based on how important the events are. While you can replicate every topic to every node in the cluster, we do not recommend it, because the extra traffic reduces throughput and increases disk space requirements.

Data Redundancy and Topic Replication

When setting up Event Broker, you can specify the number of copies (replicas) of each topic Event Broker should distribute. When you add new consumers, you don't need to update your producers. Event Broker handles the distribution and replication for you.

Kafka automatically distributes each event in a topic to the number of brokers indicated by the topic replication level specified during Event Broker configuration. While replication does decrease throughput slightly, HPE ArcSight recommends that you configure a replication factor of at least two. You need at least one node for each replica. A topic replication level of three requires at least three nodes.

A topic replication level of one means that only one broker will receive that event. If that broker goes down, that event data will be lost. However, a replication level of two means that two brokers will receive that same event. If one goes down, the event data would still be there on the other, and would be restored to the first broker when it came back up. Both brokers would have to go down simultaneously for the data to be lost. A topic replication level of three means that three brokers will receive that event. All three would have to go down simultaneously for the event data to be lost.

Creating Topics in Event Broker

You can run the `manage-topics` tool to configure topics from any node in the cluster. Your consumers and producers will use these topics. See "Producing and Consuming Event Data" on page 32 for more information.

Prerequisite: Before you can create topics, Event Broker must be installed and ZooKeeper and Kafka must be configured and running. For more information, see "Configuring ZooKeeper and Kafka" on page 16 and "Starting and Stopping Event Broker" on page 24.

To create new topics:

1. Run the following command from the *`install_dir`*`/current` directory:
   `./bin/manage-topics`
2. The configuration tool guides you through configuring a Kafka topic. Respond to the prompts to complete the configuration.

Prompt	What to enter
What is the name for the topic? (Only letters, numbers, dashes, and underscores are permitted.)	Enter a unique name for the topic. You will use this name to find the topic later, so you should create an identifying name. Valid Values: letters, numbers, dashes, and underscores The name is case sensitive. Example: `Firewall`

Prompt	What to enter
How many replicas of each partition should be made? This number must not be larger than the number of nodes in the Kafka cluster, cannot be less than 1 (the primary replica counts,) and is recommended to be 2.	Enter the number of replicas you want to create. This number must not be larger than the number of nodes in the Kafka cluster. There must be at least 1, the original, or primary replica. HPE ArcSight recommends accepting the default value, 2. This creates the primary replica and one duplicate to protect against data loss. Valid values: Any integer, 1 or greater, up to the number of nodes. Press **Enter** to accept the default of 2, or enter a different value. For more information, see "Data Redundancy and Topic Replication" on page 25.
How many Connectors are going to be sending events to this topic? Include growth estimates for the next two years, if possible.	Enter a number of connectors. Include growth estimates for the next two years, if possible. Valid values: Any integer, 1 or greater. Example: `20`
What is the expected average combined EPS for all Connectors?	Enter a number that represents your expected Events Per Second (EPS) for all connecters sending events to this topic. Example: `500000` Valid values: Any non-negative integer.
Please review the following information regarding topic configuration: Topic name: Firewall Number of replicas: 2 Number of Connectors: 20 Average EPS: 500000 Is all of the above information correct? [y/N]	Enter y to continue or enter n to start over. **Tip:** The upper case letter signifies the default, rather than the case of the letter you should type. Simply press **Enter** to use the default value.
What is the average consumption rate per consumer thread, for the slowest consumer group, in EPS? Be somewhat generous with this; if the consumers are unable to keep up, events will eventually become extremely delayed and potentially lost. If you are using Logger, the value to enter is 10000.	Enter the average Events Per Second (EPS) for the consumers in the slowest consumer group.

Prompt	What to enter
Based on the information provided, it is required to have at least <N> consumer threads for the slowest consumer group consuming from this topic. The topic will be created with <M> partitions. This is the maximum number of consumer threads that will be supported per consumer group. Do you require support for more consumer threads for any given consumer group? [y/N]	Enter n to accept the suggested number, or y to specify a greater number.
If you entered y, the following prompt is displayed. How many consumer threads are going to be receiving events from the largest consumer group for this topic?	Enter a number of consumer threads. This is the largest number of threads expected. You cannot enter a number smaller than the suggested number. Example: 75
Please review the following information regarding topic configuration: Number of partitions (and maximum number of consumer threads in largest group): 40 Is all of the above information correct? [y/N]	Enter y to accept this figure and continue or enter n to start over.
Ready to create topic by executing the following command: `./kafka/bin/kafka-topics.sh --zookeeper <ZooKeeperHost-1>:2181, <ZooKeeperHost-2>:2181, <ZooKeeperHost-3>:2181 --create --if-not-exists --topic Firewall --replication-factor 2 --partitions 40 --config cleanup.policy=delete` Press enter to continue.	Press Enter to create the topic and display its configuration, or press **CTRL+C** to exit.
Created topic "Firewall". The following topics are defined in Kafka: 1. Firewall Main menu Please choose an operation: 1. Create new topic 2. Edit or view details of existing topic 3. Exit Enter your choice [1/2/3]	Enter the number that indicates the action you want to take. For example, to exit enter 3.

Next Step: Set up ArcSight SmartConnectors to send data to the Event Broker topic. For more information, see "Producing Events with SmartConnectors" on page 32.

Viewing and Updating Existing Topics

After running the `manage-topics` tool the first time, you can run it again as needed to view and create new topics and increase the number of partitions in existing topics.

Prerequisite: These options only appear when there are existing topics.

To manage existing topics:

1. Run the following command from the *`install_dir`*`/current` directory:
   `./bin/manage-topics`
2. The configuration tool guides you through configuring Kafka topics. Respond to the following prompts to complete the configuration.

Prompt	What to enter
This configuration tool will guide you through configuring Kafka topics. Checking for existing configuration... Verifying communication with ZooKeeper (this may take a while if it doesn't work)... Main menu Please choose an operation: 1. Create new topic 2. Edit or view details of existing topic 3. Exit Enter your choice [123]	To add a topic, enter 1. To edit or view a topic, enter 2. If you entered 1, go to step 2 of "Creating Topics in Event Broker" on page 26. If you entered 2, continue here.
The following topics are defined in Kafka: 1. MyTopic 2. MyTopic_2 3. MyTopic_3 Enter the number of the topic you want to modify:	Enter the number that identifies the topic you want to view or update. Example: 3
Information about topic MyTopic_3: Topic:MyTopic_3 PartitionCount:5 ReplicationFactor:2 Configs:cleanup.policy=delete Topic: MyTopic_3 Partition: 0 Leader: 1 Replicas: 1,2 Isr: 1,2 Topic: MyTopic_3 Partition: 1 Leader: 2 Replicas: 2,3 Isr: 2,3 Topic: MyTopic_3 Partition: 2 Leader: 3 Replicas: 3,1 Isr: 3,1	The tool displays information from Kafka's topic description. **Caution:** Do not edit this information using outside tools. This will cause data loss. This information includes a summary at the top, containing: **Topic:** Topic name **PartitionCount:** Number of partitions currently specified.

Prompt	What to enter
Topic: MyTopic_3 Partition: 3 Leader: 1 Replicas: 1,3 Isr: 1,3 Topic: MyTopic_3 Partition: 4 Leader: 2 Replicas: 2,1 Isr: 2,1 Please choose an operation for topic MyTopic_3: 1. Increase number of partitions 2. Back to main menu Enter your choice [12]	**ReplicationFactor:** Number of replicas specified. **Configs:cleanup.policy=delete:** When retention time or size is reached, old data is deleted. (This policy is not configurable.) Beneath the summary is information about each partition. **Partition:** Identifies the partition. **Leader:** Indicates which broker contains the main replica. **Replicas:** Lists the brokers that contain replicas of this partition, including any currently-down brokers. **Isr:** (In Sync Replicas) Indicates which of the replicas are in sync and available for automatic fail-over. **Note:** The Leader manages the data in the partition. The other brokers, following the leader, get a copy of that data for replication. Another broker is automatically promoted to be the leader for the partition if the current leader fails. Enter the number that indicates the action you want to take. Example: 1
Topic MyTopic_3 currently has 5 partitions. To how many partitions should it be increased? **Note:** It is not possible to decrease the number of partitions. To cancel, enter 0.	Enter the number of partitions you need. For example, if you added five more Loggers, you would increase the number of partitions by five. Example: 10 Existing data are not re-balanced, new partitions will contain only new data. Connectors will automatically start sending data to new partitions without needing a configuration update.
Ready to alter topic by executing the following command: `./kafka/bin/kafka-topics.sh --zookeeper <ZooKeeperHost-1>:2181,<ZooKeeperHost-2>:2181,<ZooKeeperHost-3>:2181 --alter --if-exists --topic MyTopic_3 --partitions 10` Press enter to continue. Adding partitions succeeded!	Press **Enter** to make the configured changes.
Please choose an operation for topic MyTopic_3: 1. Increase number of partitions 2. Back to main menu	Enter the number that indicates the action you want to take. For example, to return to the main menu enter 2.

Prompt	What to enter
Enter your choice [1/2]	
Main menu Please choose an operation: 1. Create new topic 2. Edit or view details of existing topic 3. Exit Enter your choice [1/2/3]	Enter the number that indicates the action you want to take. For example, to exit enter 3.

Chapter 6: Producing and Consuming Event Data

Event Broker's publish-subscribe messaging system uses ArcSight SmartConnectors to produce event data, and supports ArcSight Logger as well as Apache Hadoop and other third-party consumers.

This chapter includes the following topics:

Producing Events with SmartConnectors

ArcSight SmartConnectors, version 7.3.0 or later, can publish events to Event Broker Topics. Event Broker supports all SmartConnector types.

To publish events, you must configure your SmartConnectors to use the Event Broker (CEF Kafka) destination. To send events to multiple topics, you can configure multiple destinations with the same Event Broker hosts and different topics. The SmartConnectors can send data to other destinations concurrently.

Once configured with an Event Broker (CEF Kafka) destination, the SmartConnector sends events to Event Broker's Kafka cluster, which can then further distribute events to real-time analysis and data warehousing systems. Once events are in Event Broker, other applications, including Logger, and any third-party application that supports retrieving data from Kafka can receive them, for example, Apache Hadoop.

Tip: You may need to tune your SmartConnectors based on expected throughput.

Event Broker balances events sent by the SmartConnector between nodes by distributing them evenly between the partitions in the configured topic.

Acknowledgments ensure that Event Broker has received the event before the SmartConnector removes it from its local queue. You can disable acknowledgments, require acknowledgment only from the primary replica, or require every replica to acknowledge the event.

The SmartConnector encodes its own IP address as meta-data in the Kafka message for consumers that require that information, such as Logger Device Groups.

For more information about SmartConnectors and how to configure a Event Broker (CEF Kafka) destination, refer to the CEF Destinations chapter of the SmartConnector User's Guide, available for download from the ArcSight Product Documentation Community on Protect 724.

Consuming Events with Logger

ADP Logger version 6.3 or later can subscribe to Event Broker events. To subscribe to Event Broker topics, you must configure an Event Broker receiver on Logger. Logger's Event Broker receivers are consumers for Event Broker's publish-subscribe messaging system. They receive events in Common Event Format (CEF) from Event Broker's topics. An Event Broker receiver connects to ArcSight Event Broker's Kafka and consumes all events for the topics it subscribes to.

When configuring an Event Broker receiver, specify the consumer group and topic. You can configure multiple Loggers to consume from the same topic as a part of a consumer group. The events in the topic will be distributed among the Loggers in the group based on the replication factor you specified for the topic.

For more information about Logger and how to configure an Event Broker receiver on it, refer to the **Configuration > Receivers** section of the ArcSight Logger Administrator's Guide, available for download from the ArcSight Product Documentation Community on Protect 724.

Sending Event Broker Data to Logger

For Logger to be able to consume Event Broker events, the SmartConnectors that send data to Event Broker must have an Event Broker (CEF Kafka) destination and the Logger must have an Event Broker receiver configured with the Event Broker hosts, consumer group and event topic list.

A group of Loggers, called a pool, can be configured to receive and distribute events between themselves. This works similarly to the Logger pool created by using the ArcSight Logger Smart Message Pool destination on the SmartConnectors.

The difference is that when the SmartConnectors have an ArcSight Logger Smart Message Pool destination, the event load is balanced by each SmartConnector, but when the SmartConnectors have an Event Broker (CEF Kafka) destination, the event load is balanced by the Loggers. Additional Loggers can be added to the pool simply by configuring the same Event Broker hosts, Consumer Group, and Event Topic List in the new Logger's Event Broker receivers, without having to reconfigure either the existing Loggers or any SmartConnectors.

The events sent to the Logger pool are distributed among the Loggers in the pool. If one Logger is down, events are sent to another Logger. When a Logger is added or removed from the Consumer Group, the event load is distributed across the pool of Loggers.

To send events from a group of SmartConnectors to a pool of Loggers, configure their Event Broker (CEF Kafka) destinations to send events to the topic that the Logger pool is consuming.

To make Logger subscribe to event data from specific SmartConnectors, you can do either of the following:

- Configure all the SmartConnectors to publish events to the same topic, then configure the Logger's Event Broker receiver to subscribe to this event topic.
- Configure each SmartConnector to publish events to different topics and then configure the Event Broker receiver on the Logger to subscribe to multiple event topics. The SmartConnector will fail destination verification if the topic name provided does not exist.

Tip: Loggers in the same Logger pool do not consume the same events, since they are in the same Consumer Group. In high availability situations, you need events to be stored on two different Loggers. To store the same events on two Loggers, configure the Loggers to have different Consumer Group names, but subscribe them to the same event topic.

The number of Loggers in a Logger pool is restricted by the number of event topic partitions. If there are only five partitions, only five Loggers will receive the events.

If you have more than five Loggers configured in the same Consumer Group, they will not normally receive events, but will be available as hot spares. When adding receivers, be sure to increase the number of event topic partitions.

Sending Event Broker data to Logger (Overview):

1. Configure the SmartConnector:
   - Setup a SmartConnector to publish to a particular Event Topic. Connectors can only send to a single topic for each destination. Additional destinations need configured if each event needs to go to multiple topics.
     Note the number of partitions in the Event Topic.
   - Configure the SmartConnector to have an Event Broker (CEF Kafka) destination.

     For more information about SmartConnectors and how to configure a Event Broker (CEF Kafka) destination, refer to the CEF Destinations chapter of the SmartConnector User's Guide, available for download from the ArcSight Product Documentation Community on Protect 724.
2. Configure Logger:
   - Create an Event Broker receiver for each Logger in the Logger pool.
   - Configure each receiver to subscribe to the Event Topics to which the SmartConnectors are publishing data. To subscribe to multiple topics, indicate the topics by specifying them in the Event Topic List parameter while configuring the Event Broker receiver.
   - Configure each receiver to be in the same Consumer Group.

     For more information on how to configure Logger to receive Kafka events by using Event Broker, see "Consuming Events with Logger" on the previous page, and refer to the Receivers section in the Configuration chapter of the *ArcSightLogger Administrator's Guide*, available for download from theArcSight Product Documentation Community on Protect 724.

Example Setup with Multiple Loggers in a Pool

You can set up your Logger pools to subscribe to events from a particular device type, such as "Firewall." To do this, you would:

1. Configure all the SmartConnectors that handle firewall events to publish these events to topic "Firewall."
2. Configure the Loggers in the Logger pool:
   - Create an Event Broker Receiver for each Logger in the pool.
   - Configure the receivers to subscribe to the event topic “Firewall,” and include them in the “Logger_Firewall” Consumer Group.

Once all the configuration is set up properly, the Logger pool will subscribe to device type Firewall.

Consuming Events with Non-ArcSight Applications

The Event Broker is designed with support for third-party tools. You can write a standard Kafka consumer and configure it to subscribe to Event Broker topics. By doing this you can pull Event Broker events into your own non-ArcSight data lake.

- The Kafka client libraries transparently handle the event compression, so anything that can communicate with Kafka should be able to handle it.
- Events are sent in standard CEF. Anything that can consume from Kafka and understand CEF can process events.
- You can set up multiple consumer groups, and each group will get a copy of every event. Therefore you can have Logger and Apache Hadoop configured to consume from the same topic and they’ll each get a copy of every event. This allows fanning out multiple copies of events without reconfiguring SmartConnectors or using additional CPU or network resources for them.
- DNS must be properly configured on all Event Broker nodes, and on all consumers and producers.

This section includes the following topics:

Using Apache Flume to Transfer Events

One of the applications you could use to transfer Event Broker events into your data lake is Apache Flume. Flume is designed to push data from many sources to the various storage systems in the Hadoop ecosystem, such as HDFS and HBase.

This section describes how to use Apache Flume as a data transfer channel to transfer events from Event Broker to Apache Hadoop or other storage systems.

- Event Broker installed—See "Installing Event Broker" on page 9.
- Flume installed—For information on how to install and configure Flume, refer to the Flume documentation, available at https://flume.apache.org/releases/content/1.6.0/FlumeUserGuide.pdf.
- Storage system installed—Refer to your storage system documentation.

Flume is controlled by an agent configuration file. You must configure Event Broker as the source agent, and your Storage system and the sink agent, and ZooKeeper as the channel agent in this file.

To configure Event Broker as the source:

Edit the agent configuration file to include the required properties, as in the table below. Configure other properties as needed for your environment.

Required Kafka Source Configuration

Property	Description
type	Set to `org.apache.flume.source.kafka.KafkaSource`.
topic	The Event Topic from which this source reads messages. Flume supports only one topic per source.

zookeeperConnect	The URI of the ZooKeeper server or cluster Kafka is using.
	POC single node example:
	`zk01.example.com:2181`
	Comma-separated list of nodes in a ZooKeeper cluster example:
	`zk01.example.com:2181,zk02.example.com:2181,zk03.example.com:2181`

To configure the sink:

The required configuration varies. Refer to the Flume documentation for details on your storage system. The section "Consuming Event Broker Events with Apache Hadoop" below provides an example of how to configure Apache Hadoop as the sink.

Consuming Event Broker Events with Apache Hadoop

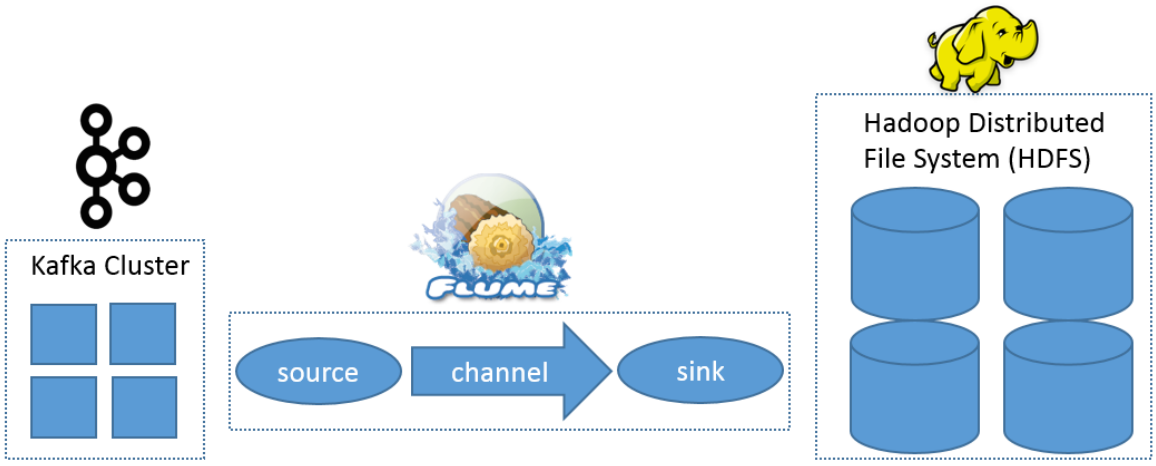
Apache Hadoop is a software framework that enables the distributed processing of large data sets across clusters of computers. You can send Event Broker events to Hadoop by using Apache Flume.

This section describes how to set up the Apache Flume agent to transfer Common Event Format (CEF) events from an Event Broker Kafka cluster to Hadoop Distributed File System (HDFS).

It includes the following topics:

Architecture for Kafka to Hadoop Data Transfer

Apache Flume uses a source module to read a Kafka topic that has raw CEF events and it then transfers the events using a memory channel, and persist them to HDFS using a sink module. The CEF files are stored on HDFS by time, in a year/month/day/hour directory structure.



Setting Up Flume to Connect with Hadoop

In the simplest deployment model, you need to deploy the Apache Flume agent on a Hadoop node server to pull events, and send them to Hadoop Distributed File System (HDFS).

Prerequisite: Hadoop must be installed before you can connect it with Flume. If you do not already have your own Hadoop deployment, you can deploy Hadoop on a Red Hat Enterprise Linux 7.2 machine.

To deploy Flume:

1. Log into your Hadoop server as the user "hadoop".
2. Download Flume from the Apache download site.
3. Uncompress the ".gz" file to your preferred deployment directory.
4. In the configuration file, add your ZooKeeper address and port, Kafka topic, and HDFS address and port.

   By default, this configuration persists a CEF file every hour. Alternatively, you could choose to roll using events count or file size. If you have high volume of events, HPE ArcSight recommends using the event count option instead of time, to avoid running out of memory. For more information, refer to Flume HDFS sink in the Flume Users' Guide.

5. Execute the following commands to create the Hadoop cefEvents directory:
   ```
   hadoop fs -mkdir /opt
   hadoop fs -mkdir /opt/hadoop
   hadoop fs -mkdir /opt/hadoop/cefEvents
   ```
6. Create a configuration file in the Flume conf directory, `bin/flume/conf/`, following the template in "Sample Flume Configuration File" below. In our example we named the file *kafka.conf*. You can name it whatever is appropriate.
   a. Copy `flume-env.sh.template` as `flume-env.sh`.
   b. Edit `flume-env.sh` file and make the following changes:
      - Set JAVA_HOME to the directory where Java was installed on your system.
      - Uncomment the line for JAVA_OPTS:

        ```
        export JAVA_OPTS="-Xms100m -Xmx2000m -Dcom.sun.management.jmxremote"
        ```
      - Set `FLUME_CLASSPATH=<Flume install directory>/lib`.
   c. Copy the common jar files from the Hadoop install directory to Flume lib directory:

      ```
      cp <Hadoop install directory>/share/hadoop/common/*.jar /<Flume Install directory>/lib
      ```

      ```
      cp <Hadoop install directory>/share/hadoop/common/lib/*.jar /<Flume Install directory>/lib
      ```
   d. Copy hadoop-hdfs-2.7.2.jar from Hadoop install directory to Flume lib directory.

      ```
      cp <Hadoop install directory>/share/hadoop/hdfs/hadoop-hdfs-2.7.2.jar /<Flume Install directory>/lib
      ```
7. Execute the following command to start Flume from its home directory:
   ```
   bin/flume-ng agent --conf conf/ --conf-file conf/kafka.conf --name tier1 -Dflume.root.logger=INFO,console
   ```
8. After you start Flume, you can find the files on HDFS by running the following command:
   ```
   hadoop fs -ls -R /opt/hadoop/cefEvents
   ```
   This path has to match the HDFS directory path, created in the Hadoop configuration section.

   The files are stored in following structure: "year/month/day/hour".

Sample Flume Configuration File

Before starting Apache Flume, create a configuration file based on the template below. The configuration file should reside in `bin/flume/conf/`. This file is called *kafka.conf* in our example. You can name your own configuration file whatever is appropriate.

```
#################################################

#Sample Flume/Kafka configuration file

#################################################

#defines Kafka Source, Channel, and Destination aliases
```

```
tier1.sources = source1
tier1.channels = channel1
tier1.sinks = sink1

#Kafka source configuration
tier1.sources.source1.type = org.apache.flume.source.kafka.KafkaSource
tier1.sources.source1.zookeeperConnect = <zookeeperAddress>:Port
# Example Address of ZooKeeper on an Event Broker node, with port 2181:
# zk01.example.com:2181
tier1.sources.source1.topic = <Kafka_topic>
tier1.sources.source1.groupId = flume
tier1.sources.source1.channels = channel1
tier1.sources.source1.interceptors = i1
tier1.sources.source1.interceptors.i1.type = timestamp
tier1.sources.source1.kafka.consumer.timeout.ms = 150
tier1.sources.source1.kafka.consumer.batchsize = 100

#Kafka Channel configuration
tier1.channels.channel1.type = memory
tier1.channels.channel1.capacity = 10000
tier1.channels.channel1.transactionCapacity = 1000

#Kafka Sink (destination) configuration
tier1.sinks.sink1.type = hdfs
tier1.sinks.sink1.channel = channel1
tier1.sinks.sink1.hdfs.path = hdfs://localhost:9000/opt/\
hadoop/cefEvents/year=%y/month=%m/day=%d
tier1.sinks.sink1.hdfs.rollInterval = 360
tier1.sinks.sink1.hdfs.rollSize = 0
tier1.sinks.sink1.hdfs.rollCount = 0
tier1.sinks.sink1.hdfs.fileType = DataStream
tier1.sinks.sink1.hdfs.filePrefix = cefEvents
tier1.sinks.sink1.hdfs.fileSuffix = .cef
```

```
tier1.sinks.sink1.hdfs.batchSize = 100
tier1.sinks.sink1.hdfs.timeZone = UTC
```

Chapter 7: Managing Event Broker

The ArcSightEvent Broker provides the Event Broker Manager, a version of Yahoo Kafka Manager, to help you monitor and manage its Kafka services. For more information about Yahoo Kafka Manager, refer to https://github.com/yahoo/kafka-manager. For more information, about Kafka monitoring, refer to the monitoring section of the Apache Kafka documentation.

This chapter includes the following topics:

About the Event Broker Manager

The Event Broker Manager enables you to manage your clusters, topics and partitions. It enables the following monitoring and management options.

- Viewing and managing cluster states, including topics, consumers, offsets, brokers, replica distribution, and partition distribution.
- Creating and updating topics.
- Generating partitions and adding partitions to a topic.
- Batch generating partition assignments, and batch reassign partitions.
- Generating partition assignments while selecting brokers, and reassigning partitions base on generated assignments.
- Running your preferred replica election.
- Managing JMX polling for broker-level and topic-level metrics (on by default.)

Starting and Stopping the Event Broker Manager

Event Broker provides several start and stop tools. These, and other Event Broker tools are located in *install_dir*/current/bin.

This section documents how to start and stop the Event Broker Manager. For information on how to start and stop Event Broker's ZooKeeper and Kafka, see "Starting and Stopping Event Broker" on page 24.

The way to start and stop the Event Broker Manager depends on whether the Event Broker services were configured. When the services have been configured, the Event Broker Manager is running by default.

To start or stop the Event Broker Manager service:

Note: This tool must be run as root.

1. Log in as the root user.
2. Run the appropriate command from the *install_dir*/current directory:

   ```
   service arst-kafka-manager (stop, start, restart)
   ```

If Event Broker services were not configured, use the tools below to start and stop Event Broker Manager.

To start or stop the Event Broker Manager:

Note: These tools must be run as the user that installed Event Broker.

1. Log in as the user that installed Event Broker:
2. Run the appropriate command from the *install_dir*/current directory:
   - `./bin/start-kafka-manager`
   - `./bin/stop-kafka-manager`

Connecting to the Event Broker Manager

Only users that can log into the Event Broker server can access the Event Broker Manager. These users can access the Event Broker Manager by using their local web browser directly from any of the Event Broker nodes or by using SSH forwarding from `127.0.0.1:8080`.

You can connect to the Event Broker Manager with most browsers, including Chrome, Firefox and Internet Explorer. For a list of browsers supported in this release, refer to the ADP Support Matrix, available for download from the ArcSight Product Documentation Community on Protect 724.

To connect directly from an Event Broker server node:

1. Log into the Event Broker server.
2. With a supported browser, connect by using the following URL:

   `http://127.0.0.1:8080`

Once you connect, the browser displays the Clusters page. See "Managing Clusters" below.

To connect from your local machine:

1. From your local system, set up SSH forwarding and connect by using a command like the following:

   `ssh -L 8080:127.0.0.1:8080 eb1.example.com`
2. With a supported browser, connect by using the following URL:

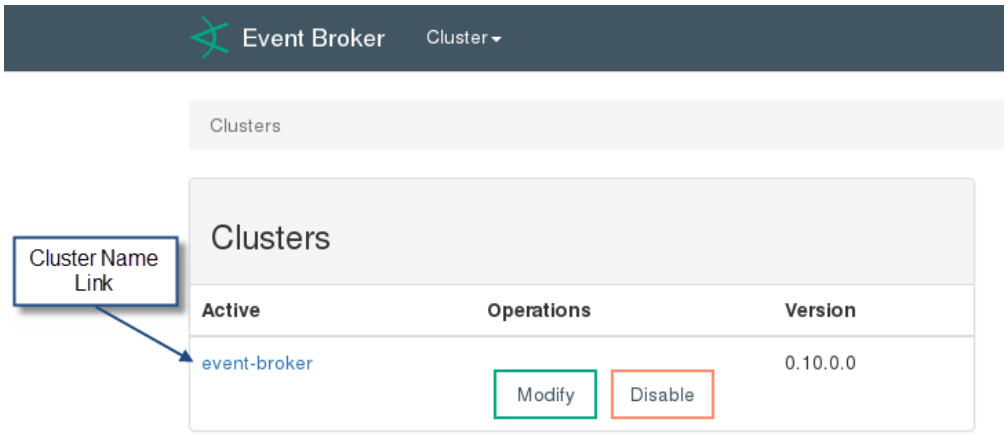
   `http://127.0.0.1:8080`

Once you connect, the browser displays the Clusters page. See "Managing Clusters" below.

Managing Clusters

The **Clusters** page is the Event Broker Manager's home page. From here you can modify, disable or delete a cluster from view in the Event Broker Manager (the cluster itself is not deleted), or drill down into the cluster for more information.

Location: Clusters



Note: The Event Broker cluster is included automatically. HPE ArcSight recommends that you only use the provided cluster. Do not add another.

To view details of a specific cluster:

Click the *Cluster Name* link. The Event Broker Manager displays the Cluster Summary page. See"Viewing Information About a Cluster" below.

To edit the cluster:

1. Click **Modify**. The Event Broker Manager displays the **Update Cluster** page.
2. Update the appropriate fields, and click **Save**.

To disable the cluster:

Click **Disable**. Once a cluster has been disabled, a **Delete** button is displayed.

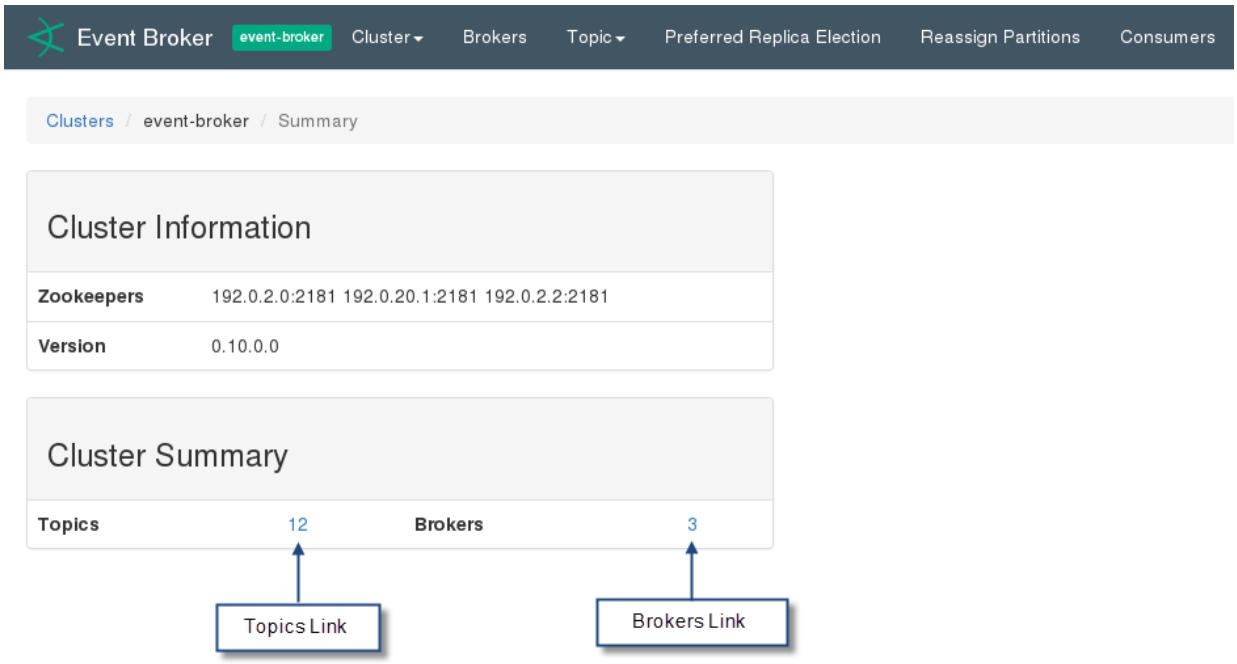
To delete the cluster:

Click **Delete**.

Viewing Information About a Cluster

On the **Summary** page, you can view the ZooKeeper processes in your cluster and drill down into its topics and brokers for more information.

Location: Clusters > *Cluster Name* > Summary



To view information about your cluster:

- If the cluster is not yet open, click **Cluster > List** in the navigation bar. Then click the *Cluster Name* link.
- If the cluster is already open, click **Clusters > *Cluster Name* > Summary**

To view or edit the topics in your cluster:

Click the **Topics** link. See "Managing Topics" on page 48.

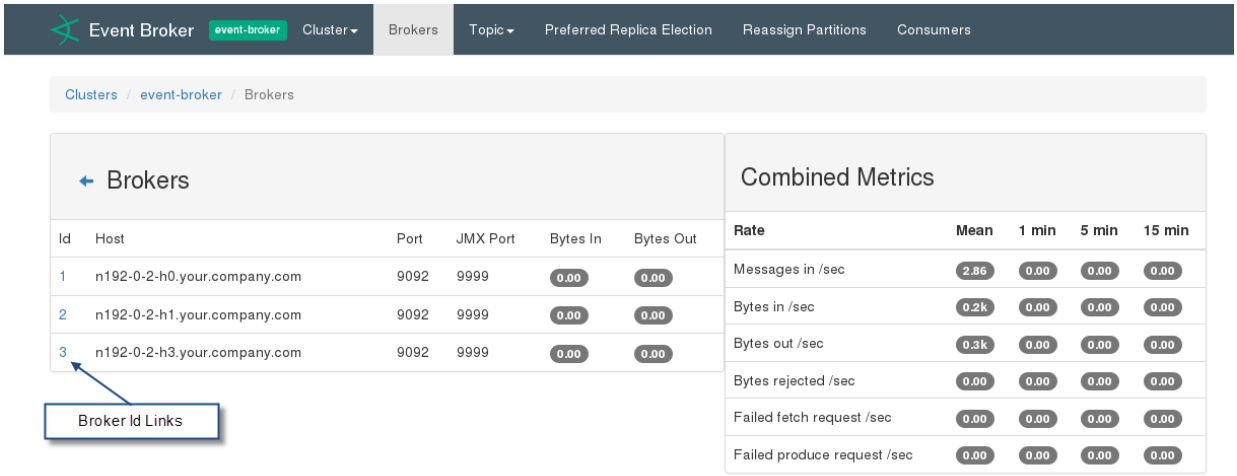
To view or edit the brokers in your cluster:

Click the **Brokers** link. See "Managing Brokers" below.

Managing Brokers

On the **Brokers** page, you can view overview information on all of your brokers and drill down into a broker for more information.

Location: Clusters > *Cluster Name* > Brokers



To view the brokers in your cluster:

Click **Brokers** in the navigation bar. The **Brokers** page opens.

To see more information about a specific broker:

Click the broker's *Id* link. The *Broker Name* ID opens. See "Viewing Broker Details" on the next page.

Viewing Broker Details

You can view detailed information about a broker from the *Broker Name* details page.

Location: Clusters > *Cluster Name* > Brokers > *Broker Name*

To view information on a specific broker:

1. Click **Brokers** in the navigation bar.
2. Click the *Broker Name* link. The *Topic Name* page opens.

From here you can view the following:

- "Summary" below
- "Metrics" on the next page
- "Messages count" on the next page
- "Per Topic Detail" on page 48

Summary

In the **Summary** section, you can see an overview of your broker, including the number of topics and partitions located on it.

Summary	
# of Topics	11
# of Partitions	142
% of Messages	58.065
% of Incoming	57.314
% of Outgoing	57.281

Metrics

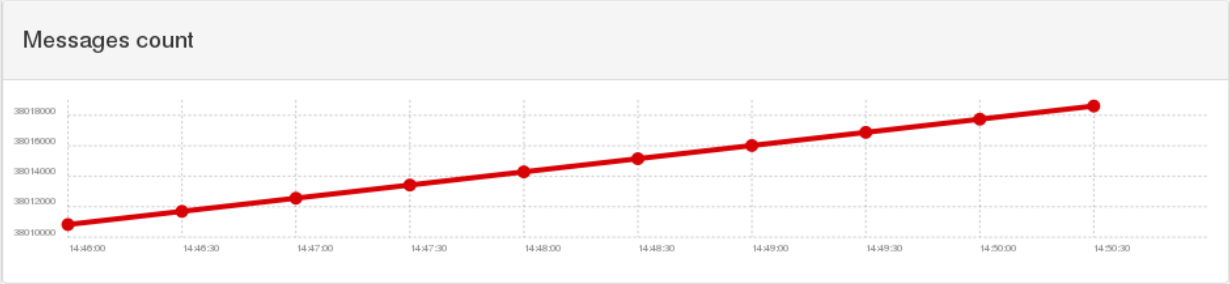
In the **Metrics** section, you can view information about the data flow.

Metrics

Rate	Mean	1 min	5 min	15 min
Messages in /sec	23.14	28.80	28.80	28.80
Bytes in /sec	2.1k	2.6k	2.6k	2.6k
Bytes out /sec	6.4k	10k	10k	10k
Bytes rejected /sec	0.00	0.00	0.00	0.00
Failed fetch request /sec	0.00	0.00	0.00	0.00
Failed produce request /sec	0.00	0.00	0.00	0.00

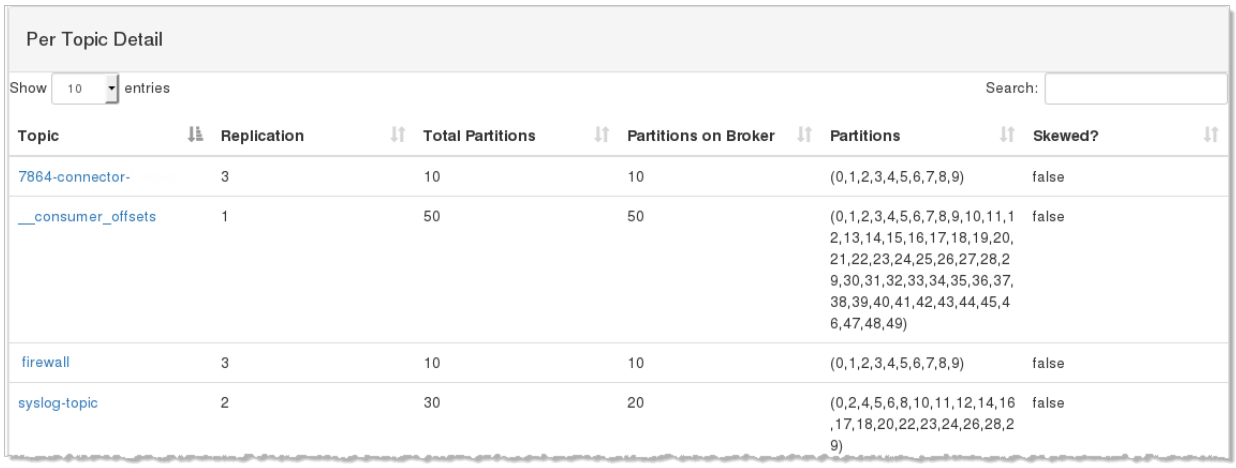
Messages count

In the **Messages** section, you can view a message view chart.



Per Topic Detail

In the **Per Topic Detail** section, you can view topic replication and partition information and drill down to view more information on each topic.


Per Topic Detail

Show 10 entries Search:

Topic	Replication	Total Partitions	Partitions on Broker	Partitions	Skewed?
7864-connector-	3	10	10	(0,1,2,3,4,5,6,7,8,9)	false
__consumer_offsets	1	50	50	(0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49)	false
firewall	3	10	10	(0,1,2,3,4,5,6,7,8,9)	false
syslog-topic	2	30	20	(0,2,4,5,6,8,10,11,12,14,16,17,18,20,22,23,24,26,28,29)	false

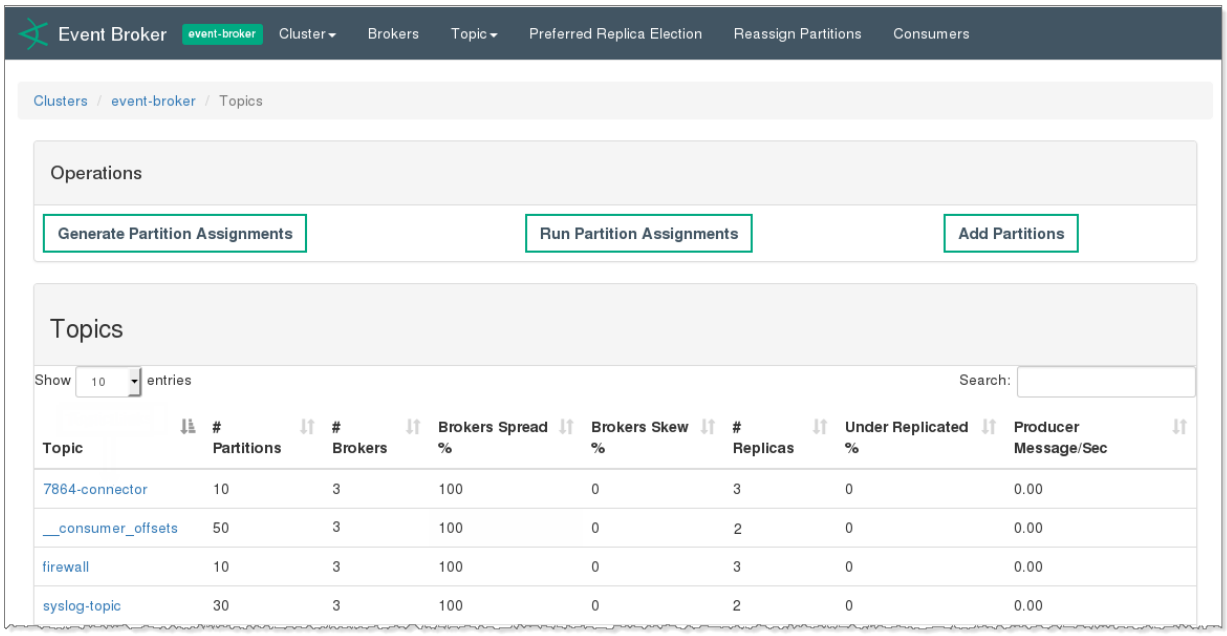
To see more information about a specific topic:

Click the *Topic Name* link in the **Per Topic Details** section. See "Viewing Topic Details" on page 51.

Managing Topics

On the **Topics** page, you can run or generate partition assignments, add a new partition, and drill down into individual topics for more information.

Location: Clusters > *Cluster Name* Topic > List



Note: A default topic, __consumer_offsets, comes with your installation. This is used internally by Kafka to manage consumers and should not be modified.

To manage the topics in your cluster:

Click **Topic > List** in the navigation bar.

To view information on a topic:

Click the *Topic Name* link. The *Topic Name* page displays the topic's summary, metrics, consumers, and partitions. See "Viewing Topic Details" on page 51.

To generate partition assignments:

1. Click **Generate Partition Assignments**.
2. Select the topics and brokers to reassign.
3. Click **Generate Partition Assignments**.

To assign partitions as generated:

1. Click **Run Partition Assignments**.
2. Select the topics to reassign.
3. Click **Run Partition Assignments**.

To add a partition:

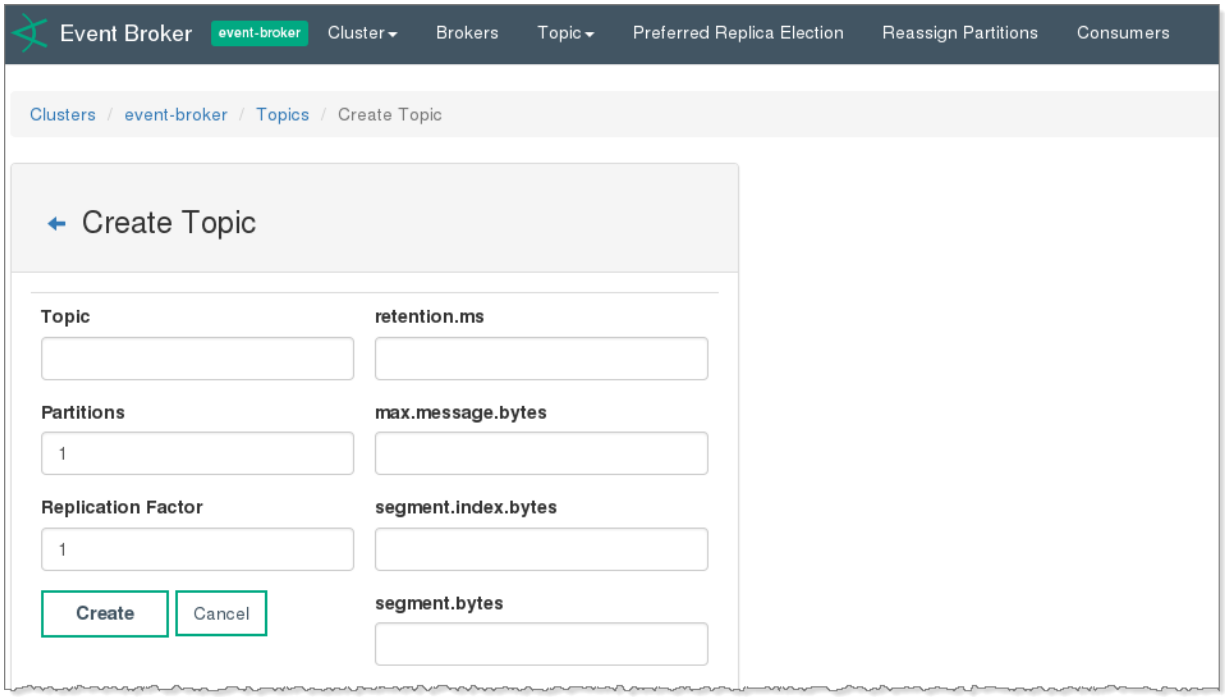
1. Click **Add Partition**.
2. Enter the new number of partitions.
3. Select the topics and brokers.
4. Click **Add Partitions**.

Creating Topics

You can create a new topic on the **Create Topic** page.

Caution: While the Event Broker Manager provides this option, HPE ArcSight recommends that you use the `manage-topics` tool to create new topics instead. The `manage-topics` tool will calculate the partition number based on traffic levels and set best-practice configuration options. For information and instructions, see "Managing Kafka Topics in Event Broker" on page 25.

Location: Clusters > *Cluster Name* Topics > Create Topic



Note: You cannot delete topics once they have been created.

To open the Add Topic page:

Click **Topic > Create** in the navigation bar.

To create a new topic:

Fill in the fields and click **Create**.

Viewing Topic Details

You can see details about a topic, including information about the summary, metrics, consumers, and partitions from the *Topic Name* details page.

Location: Clusters > *Cluster Name* Topics > *Topic Name*

To view information on a specific topic:

1. Click **Topic > List** in the navigation bar.
2. Click the *Topic Name* link. The *Topic Name* page opens.

From here you can view the following:

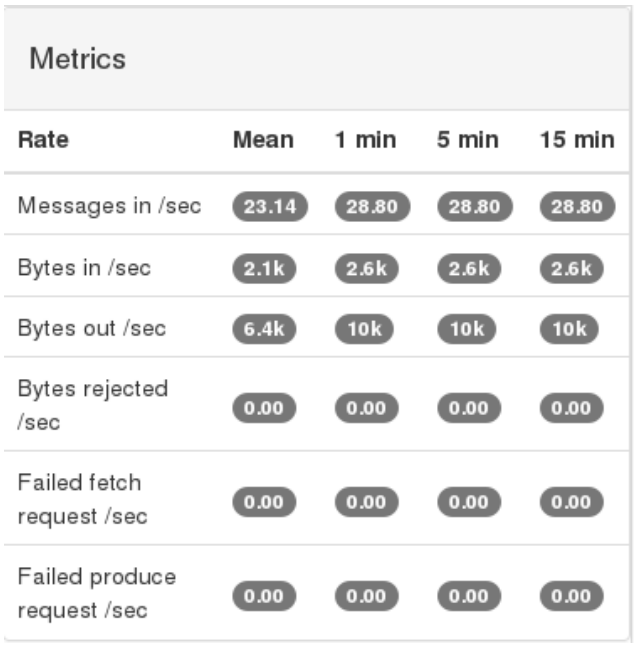
Topic Summary

In the **Topic Summary** section, you view information on the topic's replicas, partitions, and brokers.

Topic Summary	
Replication	3
Number of Partitions	10
Sum of partition offsets	0
Total number of Brokers	3
Number of Brokers for Topic	3
Preferred Replicas %	100
Brokers Skewed %	0
Brokers Spread %	100
Under-replicated %	0
Config	**Value**
cleanup.policy	delete

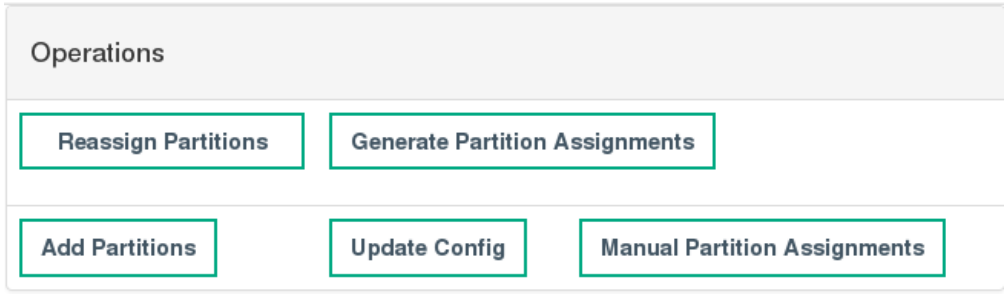
Metrics

In the **Metrics** section, you can view information about the data flow.



Operations

In the **Operations** section, you can reassign partitions, generate partition assignments, add partitions, update the topic configuration and manually assign topics to brokers.



To reassign partitions:

Click **Reassign Partitions**.

To generate partition assignments:

1. Click **Generate Partition Assignments**.
2. Select the topics and brokers to reassign.

3. Click **Generate Partition Assignments**.

To add a partition:

1. Click **Add Partitions**.
2. Enter the new number of partitions.
3. Select the topics and brokers.
4. Click **Add Partitions**.

To update the topic's configuration:

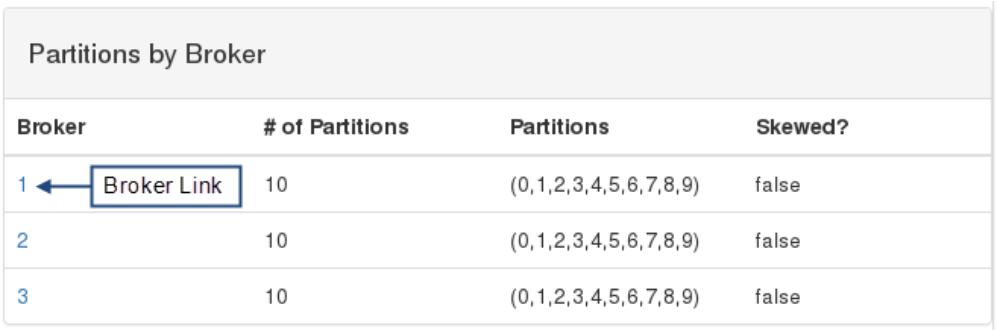
1. Click **Update Config**.
2. Edit the configuration fields.
3. Click **Update Config**.

To specify partition assignments:

1. Click **Manual Partition Assignment**.
2. Select the desired assignments.
3. Click **Save Partition Assignment**.

Partitions by Broker

In the **Partitions by Broker** section, you can see topic partition information and drill down to see details for each broker.

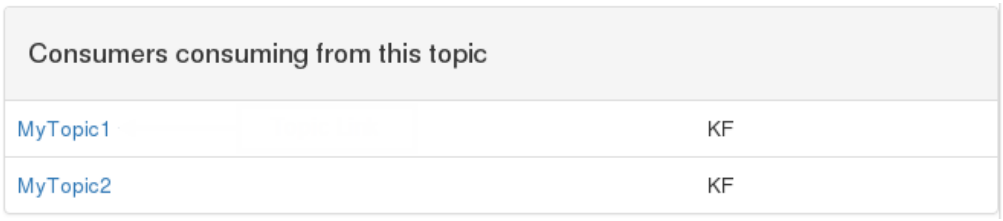


To view details on a broker:

Click the Broker link. The Topic Summary page displays information on the topic's lag, partitions, and consumer offset.

Consumers consuming from this topic

In the **Consumers consuming from this topic** section, you can drill down to see details on each consumer.

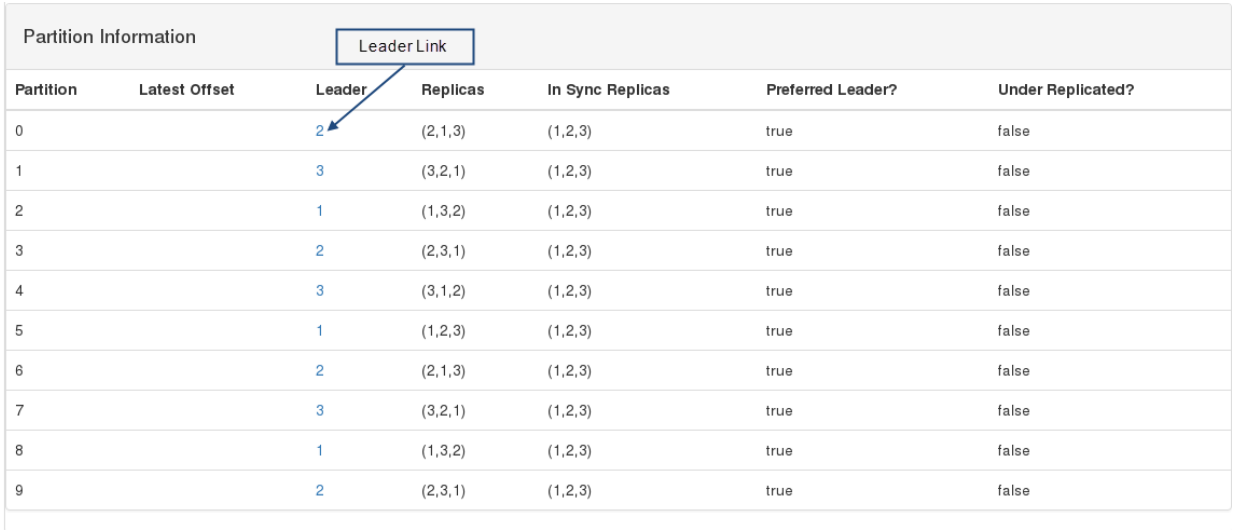


To view details on a consumer:

Click the *Topic Name* link. The Topic Summary page displays information on the topic's lag, partitions, and consumer offset.

Partition Information

In the **Partition Information** section, you can view information about the topic's partitions and drill down for more information on each leader.



Partition	Latest Offset	Leader	Replicas	In Sync Replicas	Preferred Leader?	Under Replicated?
0		2	(2,1,3)	(1,2,3)	true	false
1		3	(3,2,1)	(1,2,3)	true	false
2		1	(1,3,2)	(1,2,3)	true	false
3		2	(2,3,1)	(1,2,3)	true	false
4		3	(3,1,2)	(1,2,3)	true	false
5		1	(1,2,3)	(1,2,3)	true	false
6		2	(2,1,3)	(1,2,3)	true	false
7		3	(3,2,1)	(1,2,3)	true	false
8		1	(1,3,2)	(1,2,3)	true	false
9		2	(2,3,1)	(1,2,3)	true	false

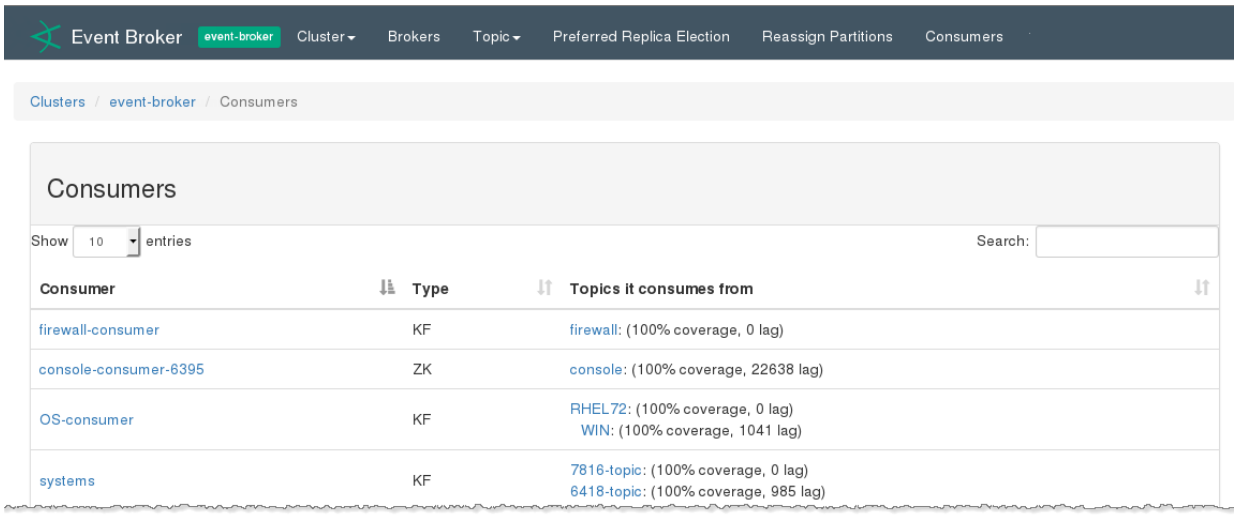
To view details on a Leader:

Click Leader link. The *Broker Name* ID page displays the broker's summary, metrics, message count, and topic details. See "Viewing Broker Details" on page 46.

Managing Consumers

On the **Consumers** page, you can see a list of consumers, view their type, the topics they consume, and drill down into each consumer and topic for more information.

Location: Clusters > *Cluster Name* > Consumers



To view or edit the consumers in your cluster:

Click **Consumers** in the navigation bar.

To view more details on a specific consumer:

Click the *Consumer Name* link. The *Consumer Name* page displays details about the consumer. You can drill down further for more information.

To view more details on the topic it consumes:

Click the *Topic Name* link. The *Topic Name* page displays details about the topic. You can drill down further for more information.

Viewing Consumer Details

You can see a information about a consumer and drill down on the topics it consumes from the *Consumer Name* details page.

Location: Clusters > *Cluster Name* Consumer > *Consumer Name*

To view information on a consumer:

1. Click Clusters > *Cluster Name* Consumer.
2. Click the *Consumer Name*.

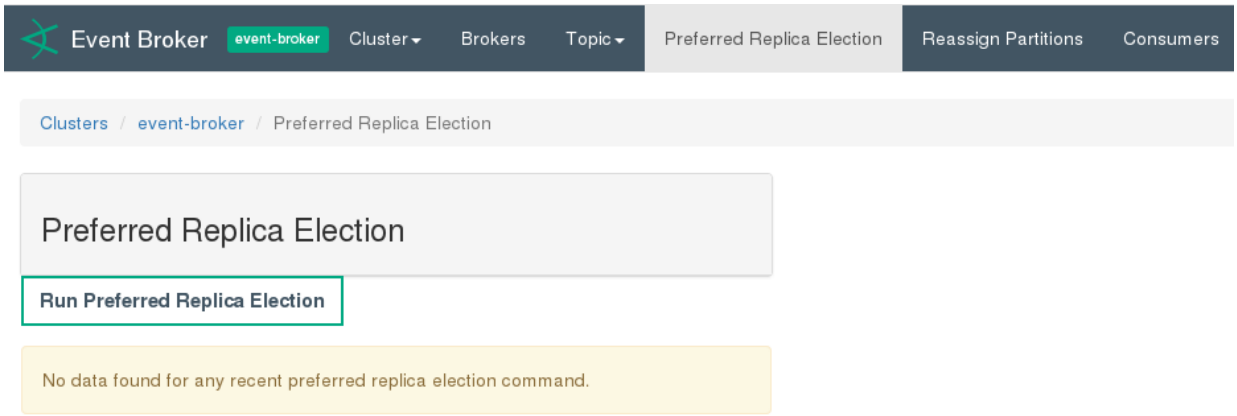
To view information on the consumed topic:

1. Click the *Topic Name*. The Consumed Topic Information page displays information about the topic. Click the topic name for more information.

Managing Preferred Replicas

You can update the replicas for each cluster on the **Preferred Replica Election** page.

Location: Clusters > *Cluster Name* > Preferred Replica Election



To open the Preferred Replica Election page:

Click **Preferred Replica Election** in the navigation bar.

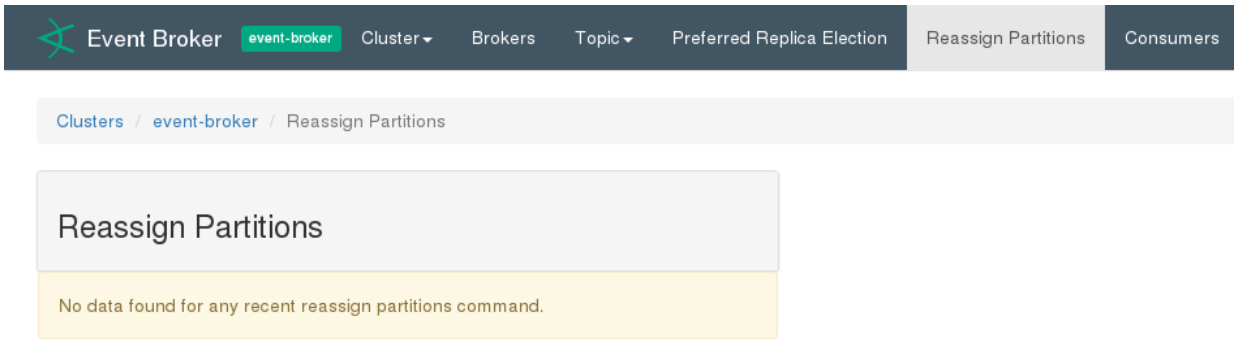
To run the Preferred Replica Election for your topic:

Click **Run Preferred Replica Election**.

Managing Partitions

You can reassign partitions for your cluster on the **Reassign Partitions** page.

Location: Clusters > *Cluster Name* > Reassign Partitions



To open the Reassign Partitions page:

Click **Reassign Partitions** in the navigation bar.

To reassign the partitions for your topic:

Click **Reassign Partitions**.

Chapter 8: Securing Your Event Broker Deployment

You are responsible for configuring your Event Broker environment securely according to your business needs and requirements. To help you do this, the Event Broker supports Transport Layer Security (TLS) and Hardware or volume-level software disk encryption. (Event data are not encrypted.) Ensure that you have your firewalls configured appropriately for your business needs and deployment.

Note: Federal Information Processing Standards (FIPS) is not supported in this release.

This chapter includes the following topics:

Setting Up Transport Layer Security

The Event Broker allows clients and brokers to communicate over Transport Layer Security (TLS) using a dedicated port, although this is not enabled by default.

To use TLS in your Event Broker environment requires the following:

- Your firewall must be configured so that port 9093 must also be reachable by all Event Broker nodes, consumers, and producers. See "Firewall Configuration" on page 63 for more information.
- You must have a digital certificate, either self-signed or signed by a certificate authority (CA). The root certificate of the CA that signed your system's certificate must be trusted on all SmartConnector and Logger servers.

  Note: HPE ArcSight strongly recommends using a CA-signed certificate.

- All producers (SmartConnectors) and consumers (Logger and Apache Hadoop) must be configured to support TLS connections. Each Event Broker node needs its own key and a certificate signed by the CA. The SmartConnectors, Loggers, and all other Event Broker consumers must have this CA imported to their trust store, or have another trust store that contains it.

To set up Transport Layer Security (TLS) in your Event Broker environment:

1. Create a directory for the certificates, such as *`install_dir`*`/current/user/certs`.

Navigate to that directory and add the Event Broker's JRE to your PATH. Do this even if your system has its own JRE installed to ensure that the correct security algorithms are used. For example, you could run the following command to add the JRE to the path.

```
PATH="/opt/arcsight/ArcSightEventBroker/current/jre/bin:$PATH"
```

2. Generate a CA certificate on one Event Broker node in the Kafka cluster. All Event Broker nodes, consumers, and producers will use this certificate.

   Note: Skip this step if your company provides you with a CA certificate.

   For example, you could run the following command to generate a CA certificate good for 365 days.

   ```
   openssl req -new -x509 -keyout ca-key -out ca-cert -days 365
   ```

3. Import the CA certificate to the trust store. You will later copy this trust store to all Event Broker nodes.

   For example, you could run the following keytool command from the *install_dir*`/current/user/certs` directory to import the CA certificate.

   ```
   keytool -keystore kafkaServer.truststore.jks -alias kafka -import -file
   ca-cert
   ```

   Note: HPE ArcSight strongly recommends against including the password in the command.

4. Perform the following steps on each Event Broker node in the cluster.
   a. Generate a TLS key and certificate.

      For example, you could run the following keytool command from the *install_dir*`/current/user/certs` directory to generate the key and certificate.

      ```
      keytool -keystore kafkaServer.keystore.jks -alias kafkaServer -validity
      days -genkey -dname
      "CN=hostname,OU=Example,O=Example,L=AnyTown,ST=CA,C=US"
      ```

      Where *hostname* is a Fully-Qualified Domain Name (FQDN) and *days* is the number of days until this certificate expires.

      Note: You will need to regenerate and redistribute the certificate in advance of its expiration.

   b. Generate the certificate signing request.

      For example, you could run the following keytool command from the *install_dir*`/current/user/certs` directory to generate the certificate signing request:

      ```
      keytool -keystore kafkaServer.keystore.jks –alias kafkaServer -certreq
      -file kafkaServer.csr
      ```

   c. Sign the generated certificate.

- For example, you could run the following command to sign the certificate.

  ```
  openssl x509 -req -CA ca-cert -CAkey ca-key -in kafkaServer.csr -out kafkaServer-cert-signed -days days -CAcreateserial
  ```

  Where *days* is the number of days until this certificate expires.
- Alternatively, if required by your organization, send the .csr file to your certificate administrator for signing, and use the kafkaServer-cert-signed you get back.

You will need to regenerate, resign, and redistribute the certificate in advance of its expiration.

d. Import the CA certificate and signed certificate to the KeyStore as a trusted certificate.

   For example, you could run the following keytool commands from the *install_dir*/current/user/certs directory to import the CA certificate and signed certificate.

   ```
   keytool -keystore kafkaServer.keystore.jks -alias EBCA -import -file ca-cert
   ```

   ```
   keytool -keystore kafkaServer.keystore.jks -alias kafkaServer -import -file kafkaServer-cert-signed
   ```

e. Make the following changes to the *install_dir*/current/user/certs file on each Event Broker server:

   - Modify the existing listeners line with the appropriate values:
     ```
     listeners=PLAINTEXT://:9092,SSL://:9093
     ```
   - Add these lines:

     ```
     ssl.keystore.location=/opt/arcsight/ArcSightEventBroker/current/user/certs/kafkaServer.keystore.jks
     ```

     ```
     ssl.keystore.password=test1234
     ```

     ```
     ssl.key.password=test1234
     ```

     ```
     ssl.truststore.location=/opt/arcsight/ArcSightEventBroker/current/user/certs/kafkaServer.truststore.jks
     ```

     ```
     ssl.truststore.password=test1234
     ```

     ```
     ssl.client.auth=none
     ```

5. Copy the kafkaServer.truststore.jks generated in the previous step to all Event Broker nodes.

SmartConnector Configuration for TLS

To configure your SmartConnectors for Transport Layer Security (TLS), you will need to copy the JKS files you created to the SmartConnectors and use that JKS location when configuring the SmartConnector destinations.

You can use ArcSight Management Center (ArcMC) to both push the JKS files to your SmartConnectors and to configure their Event Broker (CEF Kafka) destinations.

To use ArcMC to push the JKS to the various SmartConnectors:

1. Log into ArcMC.
2. Create a new repository for the JKS files.
3. Upload the JKS files to this repository.
4. Push the uploaded files to the applicable connectors, as described in the ArcSight Management Center Administrator's Guide.

After the push is complete, they will be in the connector folder you specified.

To use ArcMC to configure the SmartConnector Event Broker (CEF Kafka) destination to use this JKS path:

1. Log into ArcMC.
2. Open **Node Management > Connectors**.
3. Select the connectors you want to update.
4. Add or Update the Destination, as described in the ArcSight Management Center Administrator's Guide.

Logger Configuration for TLS

To configure your Logger for TLS, you will need to copy the Java KeyStore (JKS) TrustStore that contains the CA certificate you generated to the `user/logger` directory on all Logger servers. Be sure to give user and group ownership permissions on this file to the user who installed Logger.

When configuring the receiver, enable TLS and parameters to use that JKS TrustStore.

To copy the JKS TrustStore that contains the CA certificate to Logger:

- For software Logger, copy the JKS TrustStore to the following directory: `/<installdir>/current/arcsight/logger/user/logger`
- For Logger appliances, copy the JKS TrustStore to the following directory: `/opt/arcsight/logger/user/logger/`

To add or edit your Logger's Event Broker receivers:

1. Log into Logger.
2. Open **Configuration > Receivers**.
3. Add or edit a receiver, as described in the Logger Administrator's Guide.

4. Edit the receiver parameters as follows:
   - Set **Use SSL/TLS** to `true` to enable TLS communication with Event Broker.
   - Provide the JKS TrustStore location in the **Event Broker CA or Self-Signed Certificate File Location** field.

Firewall Configuration

You can configure your firewall rules to allow access to only the services that are required.

The Event Broker environment requires the following access:

- Kafka uses port 9092. This port must be reachable by all Event Broker nodes, consumers, and producers. If you are using TLS, port 9093 must also be reachable.
- ZooKeeper uses ports 2181, 2888, and 3888. These ports must be mutually reachable between all Event Broker nodes.
- The Event Broker Manager uses port 9999 to monitor Kafka. This port must be mutually reachable between all Event Broker nodes.

Glossary

A

Apache Flume
A service for efficiently collecting, aggregating, and moving large amounts of log data.

Apache Hadoop
A software framework that enables the distributed processing of large data sets across clusters of computers. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage. You can configure Hadoop as an Event Broker consumer.

Apache Kafka
An open source distributed publish-subscribe messaging system. Kafka is installed as part of the Event Broker.

Apache ZooKeeper
A centralized service for maintaining configuration information, naming, providing distributed synchronization, and providing group services. ZooKeeper's architecture supports high availability through redundant services. ZooKeeper is installed as part of Event Broker and it used to coordinate the Kafka cluster.

B

broker
An instance of the Kafka server software.

C

CEF
Common Event Format (CEF) is an extensible, text-based, high-performance format designed to support multiple device types in the simplest manner possible. Various message syntaxes are reduced to one-matching ArcSight Enterprise Security Manager (ESM) normalization. Specifically, CEF defines a syntax for log records comprised of a standard header and a variable extension, formatted as key-value pairs. This format contains the most relevant event information, making it easy for event consumers to parse and use them.

channel
In Apache Flume, a buffer that stores events, until a sink has successfully written the them.

cluster
A collection of brokers working together to increase throughput and durability.

consumer
A process that subscribes to one or more topics and processes the feed of messages.

consumer group
A logical grouping of several consumers, where only one consumer in the group will process each message.

consumer offset
The read position for a consumer in a partition.

D

device group
In Logger, a category of named source IP addresses called devices. Device groups can be associated with storage rules that define the storage group where events from specific devices are stored. Refer to your Logger documentation for complete details.

H

HDFS
Hadoop Distributed File System (HDFS) is a Java-based file system that provides scalable and

reliable data storage, and was designed to span large clusters of commodity servers.

L

leader
The broker containing the original replica of a partition, and manages that data.

Logger
An ArcSight product that receives event data and stores it for retrieval and analysis. You can configure Logger as an Event Broker consumer. Refer the Logger Administrator's Guide for complete details.

N

node
The machine a Kafka instance is running on.

O

offset
A sequential number identifying the location of a message in a partition. The sum of partition offsets is the total number of events in the topic.

P

partition
A segment of a topic. There can be one or more partitions for each topic. The number of partitions limits the maximum number of consumers in a consumer group.

pool
A logical grouping of Loggers. Loggers in a pool belong to the same consumer group and subscribe to the same topics.

producer
A process that publishes messages to a topic.

publish
The action of sending topics to the Event Broker. A producer publishes event on a given topic.

Q

quorum
The set of all in-sync replicas for a particular partition. Replicas are considered in-sync if they are caught-up to the leader. The leader waits until a majority of replicas have received the data before considering it to be committed. On leader failure, a new leader is elected through the coordination of a majority of the followers. If there is an odd number of replicas, a majority is ensured. Any replica in the quorum can become the leader. This enables the producer to continue to publish messages and the consumer continues to receive the correct messages, even when there is failure.

R

receiver
In Logger, the process that receives events, captures event data, and populate each event with information about its origin. Refer to the Logger Administrator's Guide for complete details.

replica
A copy of a partition. There can be one or more per partition; even if there is no redundancy, the original is still called a replica.

S

sink
In Apache Flume, the sink forwards events to the storage destination.

SmartConnector
An ArcSight product that collects event data from objects on your network. They normalize the data in two ways: normalizing values (such as severity, priority, and time zone) into a common format,

and normalizing the data structure into a common schema. You can configure SmartConnectors as Event Broker producers. Refer to the SmartConnector User's Guide for complete details.

source
In Apache Flume, a source sends events to Flume.

subscribe
The action a consumer takes in order to receive the events that are published to a topic. A subscriber can receive the events published while the subscriber is active, or it can request events "from the beginning of time" the first time its consumer group is seen. From then on it will retrieve events since the last time a consumer for its group retrieved events.

T

topic
A feed of messages relating to the same category.

Y

Yahoo Kafka Manager
An open source tool for managing Apache Kafka. Event Broker includes a version of Yahoo Kafka Manager

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 system, click the link above and an email window opens with the following information in the subject line:

Feedback on Administrator's Guide (Event Broker 1.0)

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 arc-doc@hpe.com.

We appreciate your feedback!