opentext™

ArcSight SmartConnectors

Software Version: 8.4.3

Performance Tuning Guide for SmartConnectors

Document Release Date: October 2023
Software Release Date: October 2023

Legal Notices

Open Text Corporation

275 Frank Tompa Drive, Waterloo, Ontario, Canada, N2L 0A1

Copyright Notice

Trademark Notices

Documentation Updates

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

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

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

https://www.microfocus.com/support-and-services/documentation

Contents

Performance Tuning Guide for SmartConnectors

This Performance Tuning document provides sizing and tuning recommendations to improve the performance and achieve optimal Events Per Second (EPS) results for ArcSight SmartConnector for Windows Event Log - Native and the Syslog SmartConnectors.

Intended Audience

This guide provides information for IT administrators who are responsible for managing the ArcSight software and its environment.

Additional Documentation

The ArcSight SmartConnector documentation library includes the following resources:

- Technical Requirements Guide for SmartConnector, which provides information about operating system, appliance, browser, and other support details for SmartConnector.
- Installation and User Guide for SmartConnectors, which provides detailed information about installing SmartConnectors.
- Configuration Guides for ArcSight SmartConnectors, which provides information about configuring SmartConnectors to collect events from different sources.
- Configuration Guide for SmartConnector Load Balancer, which provides detailed information about installing Load Balancer.

For the most recent version of this guide and other ArcSight SmartConnector documentation resources, visit the documentation site for ArcSight SmartConnectors 8.4.

Contact Information

We want to hear your comments and suggestions about this book and the other documentation included with this product. You can use the comment on this topic link at the bottom of each page of the online documentation, or send an email to MFI-Documentation-Feedback@opentext.com.

For specific product issues, contact Open Text Support for Micro Focus products.

Overview

Performance tuning requires an established performance baseline that can be used to compare if a performance issue arises. It is essential to collect data and analyze it effectively to identify and correct performance problems. Continuous monitoring of performance helps identify

possible symptoms and statistics, which can then be used to make configuration changes to correct performance issues.

This document addresses performance and stability issues that you have experienced with ArcSight SC 8.0.0.8322. A significant amount of code re-architecture was done for SmartConnector release 8.0.0.8322 to ensure that the connectors can achieve the maximum EPS.

The guidelines in this document are an attempt to simplify proactive monitoring and solve bottleneck scenarios.

The performance tuning guidance provided here is specific to Syslog and Windows Event Log - Native SmartConnectors. Most of the 200+ ArcSight SmartConnectors use the same connector framework, and any performance improvement changes might similarly impact other SmartConnectors.

Note: Performance tuning/monitoring needs careful study of possible scenarios, including input event size, input events per second, and connectors' hardware. Do not assume that updating parameters with higher numbers translates into better results.

Tuning SmartConnectors

The default configuration for each SmartConnector made in the `agent.properties` file might be sufficient for lower EPS. It might not utilize the full CPU capacity and might impede your hardware from achieving the best possible results. To reach higher EPS and optimal hardware utilization, you must tune parameters as per your requirement.

Ensure that the SmartConnector is tuned by an ArcSight administrator, who is aware of the types of events and the approximate EPS that the SmartConnector is expected to receive. The connector tuning also requires an understanding of SmartConnector logs. For more information, see "Reading SmartConnector Logs" on the next page.

You can tune the SmartConnector the first time after deployment, and any time after that if you see a drop in performance. You must monitor the input and the output EPS at regular intervals to see any drop in performance. The decrease in performance might be due to newer events coming to the connector or higher input EPS. Performance tuning is not required when input EPS for the connector is equal to output EPS..

Note: If you see a low input EPS while the number of events keep going up, the tuning exercise might not have been executed correctly. As a result, the event source throttles and makes it harder to debug. Hence, monitor the initial surge of EPS and wait to see a slow downward trend.

How Much to Tune?

Whenever you change the `agent.properties` file based on the tuning parameters, run multiple tests to ensure that you see sustained and desired EPS. While a small amount of data in cache is usually fine, ensure that you do not see any consistent build-up of data in cache or queue. Ensure that there are no errors in the logs.

Note: If you notice that the CPU utilization reaches 80% with a consistent cache or queue built-up, despite tuning parameters, then it might be an indicator that it is time to upgrade your hardware.

Reading SmartConnector Logs

When evaluating system performance, monitor the following parameter at peak event surges for approximately 20 to 30 minutes.

Parameter	Where to Find
Input EPS	Value of "Queue Rate(SLC)= " in agent.log
Queue Build-up	Browse to the <installation folder>/current/user/agent/agentdata/location, then count the number of files with suffix queue.syslogd
Cache	Value of "{C=" in agent.log
Output EPS	Value of "T=" in agent.log

For example, if the output EPS hovers around 21k (staying between 20k-22k) for 30 minutes, then the final output EPS can be considered as 21k.

Improving Memory Utilization

The minimum and maximum heap size in a connector are 1 GB, which allows more in-memory operations and faster execution. If there is more RAM available in the machine running the connector and you have high input EPS, OpenText recommends that you increase the heap size to 4 GB.

To increase the memory size in SmartConnectors running in stand-alone mode:

1. Open the following file:

   Windows: `ARCSIGHT_HOME\current\bin\scripts\connectors.bat`

   Linux: `$ARCSIGHT_HOME/current/bin/scripts/connectors.sh`
2. Change the following parameter:

```
ARCSIGHT_MEMORY_OPTIONS=" -Xms1024m -Xmx1024m "
```

to

```
ARCSIGHT_MEMORY_OPTIONS=" -Xms4096m -Xmx4096m "
```

To increase the memory size SmartConnectors runing as a service:

1. Open the `user/agent/agent.wrapper.conf` file.
2. Change the following line:

   ```
   wrapper.java.initmemory=1024 wrapper.java.maxmemory=1024
   ```

   to

   ```
   wrapper.java.initmemory=4096 wrapper.java.maxmemory=4096
   ```

Improving Disk Space Utilization and Network Usage

This section has the following information:

Improving Disk Space Utilization or Caching

Connectors use the disk to store cache and queue files. Disk space utilization is based on the number of cache and queue files and their respective sizes. The cache and queue files are created when the connector's various subsystems cannot process as fast as expected. For example, if the parsing is slow, queue files are saved in `<installation folder>/current/user/agent/agentdata/`.

If the destination cannot receive events at the rate the connector sends events, cache files build up. When the connector cannot send events to a destination, cache files are clustered in queue files. These queue files generate a cascading effect backwards.

Note: If the queue files or cache files reach their space limit, the newer events are dropped.

To avoid events getting dropped:

- If you see only cache files, increase the number of destination threads. If you see cache files and queue files, increase the number of destination threads and check if both clear up.
- If you see only queue files, increase the number of parser threads.
- If you have sudden surges and traffic peaks, consider creating more queue files so that events are not dropped and cached for later processing.

Improving CPU Usage

If the input EPS is high and there are enough free CPU cycles on the machine running the connector, you can increase the number of parsers and destination threads to improve the CPU

usage.

However, adding more threads than required is not recommended.

Improving Network Usage

Communication from the source to the connector and from the connector to the destination (along with all the hubs such as any routers or switches) must remain at the same network capacity.

- If the input EPS is higher than 50K, your network card must be 1 GB.
- If the input EPS is higher than 100K, your network card must be 10 GB.

Tuning Performance of Syslog SmartConnectors

You can improve the performance of SmartConnectors by implementing the following changes:

- Increasing the number of parser threads to improve parsing speed.
- Increasing heap memory to allow additional in-memory operations.
- Increasing destination write speed for CEF File, Transformation Hub and AWS s3 bucket to allow multiple parallel streams of write to the destination.

Syslog SmartConnectors can be configured with Logger, ESM, Transformation Hub, and files as single or multiple destinations.

Certain factors, such as configuring multiple destinations and the output EPS, might affect the performance of your SmartConnectors.

For example, with CEF files as a destination, the only limiting factor is disk speed. While with Logger as a destination, network latency and Logger hardware or performance affect the connector performance.

Enabling Transport Layer Security (TLS) decreases the throughput, though not significantly.

Configuration Parameters

Parameter	Description
Persistent Connection	Set `transport.loggersecure.connection.persistent`to **True** when using Logger as destination. It allows reuse of the existing HTTPS connections and not tear them down for every batch of events.
Custom SubAgent List	If you areaware of the types of events received by the connector, you can set `agents[0].usecustomsubagentlist` to **True**, and specify in the `agents[0].customsubagentlist` parameter, comma-separated values. For example: example linux_auditd_syslog, generic_syslog.
Parser threads	Increase the number of parser threads for better performance if the input EPS is high.
Destination Threads	If a queue is getting built up: CEF - `transport.ceffile.threads` Logger -`transport.loggersecure.thre`
Number of Kafka Threads	Set the producers`transport.cefkafka.multiplekafka` parameter to **True** to use as many threads as possible as `transport.cefkafka.threads`. If the parameter is set to false, even if the number of destination threads is set to a higher number, the condition is not overruled.
Time/size of buffers before events are sent to TH	`transport.cefkafka.buffer.bytes` and `transport.cefkafka.linger.ms`. The kafka threads wait for either the bytes to be full or the linger.ms to be completed before pushing events to the TH.

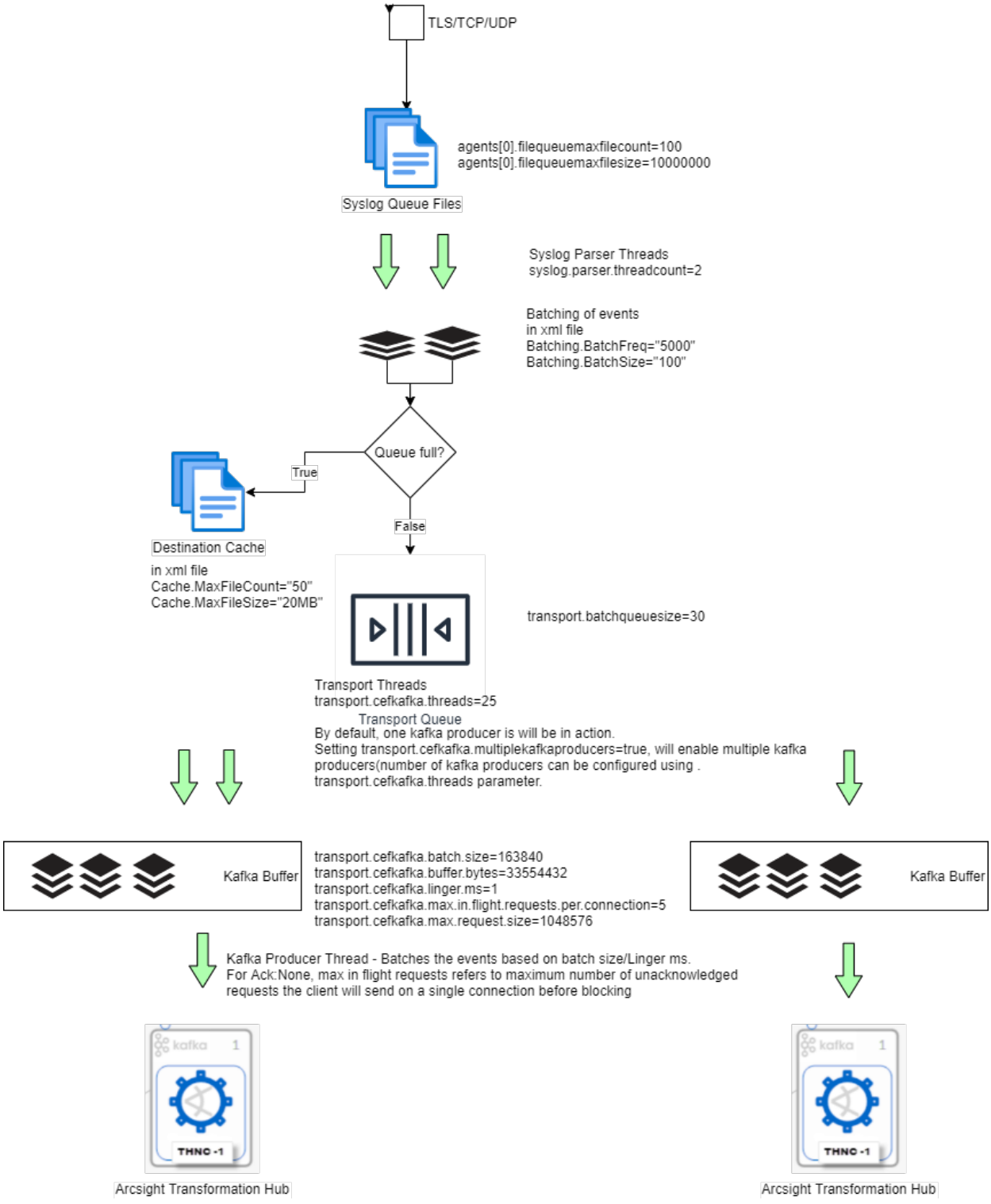
Syslog SmartConnector Parameters

Parameter	Description	Value
`agents[0].filequeuemaxfilecount`	Sets the maximum number of queues. Incoming events received by the Syslog connector are saved in queue files. Ensure enough storage available in the disk. If Ad hoc EPS inputs are expected, increase this value to avoid event drops.	100
`agents[0].filequeuemaxfilesize`	Maximum file size of a queue file (in bytes)	100
`Parsing: syslog.parser.threadcount`	Number of threads to process input raw events. (Syslog Parser Threads) **Batching of Events in the XML File**.	2

Parameter	Description	Value
`Batching.BatchFreq`	Batching frequency before sending events to a transport queue (in milliseconds)	5000
`Batching.BatchSize`	Number of events per batch before sending them to a transport queue	100
`transport.batchqueuesize`	Maximum queue size of batched events on hold. Syslog parser threads push processed events to this queue, and transport threads take the event batches to be sent to the destination.	30
`Cache.MaxFileCount`	Maximum number of cache files. If the destination is down or if there is event transfer latency, events are cached in the `current/user/agent/agentdata` folder.	50
`Cache.MaxFileSize`	Maximum size of a cache file	20 MB

Architecture Overview

The following diagram shows the architecture of Syslog SmartConnector that has Transformation Hub as a destination. The diagram also indicates parameters and values that can be set. However, you can tweak these values as per your requirements. For example, you can increase the thread count if you notice that the number of queue files increases. For more information about the parameters used in the diagram, see Syslog SmartConnector Parameters.


TLS/TCP/UDP
agents[0].filequeuemaxfilecount=100
agents[0].filequeuemaxfilesize=10000000
Syslog Queue Files
Syslog Parser Threads
syslog.parser.threadcount=2
Batching of events
in xml file
Batching.BatchFreq="5000"
Batching.BatchSize="100"
Queue full?
True
False
Destination Cache
in xml file
Cache.MaxFileCount="50"
Cache.MaxFileSize="20MB"
transport.batchqueuesize=30
Transport Threads
transport.cefkafka.threads=25
Transport Queue
By default, one kafka producer is will be in action.
Setting transport.cefkafka.multiplekafkaproducers=true, will enable multiple kafka producers(number of kafka producers can be configured using .
transport.cefkafka.threads parameter.
Kafka Buffer
transport.cefkafka.batch.size=163840
transport.cefkafka.buffer.bytes=33554432
transport.cefkafka.linger.ms=1
transport.cefkafka.max.in.flight.requests.per.connection=5
transport.cefkafka.max.request.size=1048576
Kafka Buffer
Kafka Producer Thread - Batches the events based on batch size/Linger ms.
For Ack:None, max in flight requests refers to maximum number of unacknowledged requests the client will send on a single connection before blocking
kafka
1
THNC -1
Arcsight Transformation Hub
kafka
1
THNC -1
Arcsight Transformation Hub

Tuning Transformation Hub Parameters

Parameter	Description	Value
`transport.cefkafka.threads`	Maximum threads that can be processed by the `transport.batchqueue` and sent to the destination.	25
`transport.cefkafka.batch.size`	The Kafka producer attempts to batch records together into fewer requests whenever multiple records are sent to the same partition. Performance improves for both client and server.	163840
`transport.cefkafka.buffer.bytes`	Controls the default batch size in bytes	33554432
`transport.cefkafka.linger.ms`	The producer groups records that arrive in between request transmissions into a single batch request.	1
`transport.cefkafka.max.in.flight.requests.per.connection`	The maximum number of unacknowledged requests a client sends on a single connection before blocking the operation.	5
`transport.cefkafka.max.request.size`	The maximum size of a request in bytes. Limits the number of record batches the producer sends in a single request to avoid sending multiple and heavier requests.	1048576
`transport.cefkafka.multiplekafkaproducers`	The maximum number of cache files. Creates different kafka producer for each transport.cefkafka.threads, The `multiplekafkaproducers` must be set to **true**.	False

Performance Statistics for Syslog SmartConnectors

Performance tuning and measurement must be done based on the requirement of the user's environment.

The performance results in the following sections are achieved in the OpenText Lab settings. These numbers must be used as guidance. OpenText strongly recommends that you run the test in your setup.

Performance Results Using a CEF File as a Destination

Configurable Section	Description
Hardware Configuration	Connector: G 10 Appliance: ProLiant DL360 G10, RAM: 64 GB CPU: 32, 16 cores
Heap size	Initial Java Heap Size (in MB) = 1024 Maximum Java Heap Size (in MB) =2048
Destination	CEF
Duration	10 Mins
Connector used	Syslog Deamon
Log file	Unix_OS_Events.txt

Agent.properties	`transport.ceffile.connection.persistent=True`
	`agents[0].usecustomsubagentlist=True`
	`agents[0].customsubagentlist=generic_syslog`

EPS	Parser Threads	Destination Threads
10k	5	5
20k	5	5
30k	10	10
40k	10	10
50k	10	10
60k	10	10
70	10	10
80k	10	10

EPS	Parser Threads	Destination Threads
90	15	15
100k	15	15
110k	151	15

Performance Results Using Transformation Hub as a Destination

Transformation Hub Leader ACK Performance Improvements:

Configurable Section	Description
Hardware Configuration	Connector:G10 Aappliance (ProLiant DL360 G10) RAM (64 GB) CPU (32) ,16 cores
Heap size	Initial Java Heap Size = 4 GB Maximum Java Heap Size = 4 GB
Destination	Transformation Hub
Duration	10 mins
Connector used	Syslog Deamon
Log file	Cisco Merraki

Results

100.04k None	None	99.95k
99.32k	Leader	99.36k
100.09k	All	99.99k

Tuning Performance of Windows Event Log - Native SmartConnectors

Windows Event Log - Native agent process has the following separate queues:

- **Processing queue:** It stores unprocessed events from Windows event logs.
- **Batching queue:** It stores the processed events that are ready to be batched.
- **Sending queue:** It stores event batches that are ready to be sent to the Windows Event Log - Native connector process.

A pool of threads monitors the processing queue for events, processes them, and puts them to the batching line.

Windows Event Log - Native SmartConnector Parameters

Parameter	Description
Queue parameters	Parameters related to Queue. agents[0].filequeuemaxfilecount: Maximum number of queue files to store the raw events. Default value is 100. If the queue is getting filled increase this value to have more number of queue files to avoid any event drop. agents[0].filequeuemaxfilesize: Maximum size of each queue file. Default value is 10MB. If the queue is getting filled increase the size of queue file to store more events.
Winc agent parameters	Modify these parameters to send more events to the connector. winc.winc-agent.eventBatchSize: The default number is 64. winc.winc-agent.processingThreadPoolSize: The default size is 5. winc.winc-agent.senderThreadPoolSize: The default size is 2.
Eventprocessing threadcount	syslog.parser.threadcount: Event processing thread count for high input EPS. The default count is 2. If the queue file is built up in the user\agent\agentdata folder, then increase the thread count.
Destination Threads	High cache CEF - transport.ceffile.threads transport.ceffile.connection.persistent=TRUE transport.ceffile.threads

Performance Statistics for Windows Event Log - Native SmartConnectors

Performance tuning and measurement must be done based on the requirement of the user's environment.

The Performance results in the following sections are achieved in the OpenText Lab settings. These numbers must be used as guidance. OpenText strongly recommends that you must run the test in your setup.

Performance Results Using Windows Event Log - Native SmartConnector 8.4

Details	Details
Connector Machine	• Gen 10 • 16 core / 64 GB • Windows Server 2022
Destination (CEF File)	• Gen 10 • 16 core / 64 GB • Windows Server 2022
Build	8.4.0.8949
Maximum log size in event viewer	10 GB
Log size(Mixed Events)	7 KB

agent.default.properties Values

Default	Custom
`winc.winc-agent.eventBatchSize=64`	`winc.winc-agent.eventBatchSize=59`
`winc.winc-agent.processingThreadPoolSize=5`	`winc.winc-agent.processingThreadPoolSize=30`
`winc.winc-agent.senderThreadPoolSize=2`	`winc.winc-agent.senderThreadPoolSize=7`

Heap Size

Default	Custom
`1GB-1GB`	`1GB-8GB`

agent.properties

Default	Custom
`agents[0].eventprocessorthreadcount=20`	`agents[0].eventprocessorthreadcount=40`

Event IDs Used to Execute Performance Tests:

Event ID	Event Description
4634	An account was logged off
4624	An account was successfully logged on.
4672	Special privileges assigned to new logon.
4768	A Kerberos authentication ticket (TGT) was requested.
4769	A Kerberos service ticket was requested.

Default Values

Input EPS (Connector)	EPS Processed by Connector	EPS Sent to Destination	Thread	Cache	System CPU (%)	System RAM Usage (Mb)	Queue File Count
17k	17k	17k	2	0	32.9	7.5	1
21k	21k	21k	2	0	38.3	7.7	1
25k	25k	25k	2	0	44.6	7.8	1
32k	24k	24k	2	0	51.9	8.2	84

Queue File Count is the number of files containing windows events to be processed.

Send Documentation Feedback

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

Feedback on Performance Tuning Guide for SmartConnectors (SmartConnectors 8.4.3)

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 MFI-Documentation-Feedback@opentext.com.

We appreciate your feedback!