ONE IDENTITY™

syslog-ng Premium Edition 7.0.10

Performance Guideline for syslog-ng Premium Edition

syslog-ng PE Performance Guideline for syslog-ng Premium Edition
Updated - August 2018
Version - 7.0.10

Contents

Preface

This document summarizes the findings of several performance tests and measurements carried out on different versions of syslog-ng Premium Edition. The main focus was syslog-ng PE version 7.0, but similar performance tests are performed and published regularly with every major syslog-ng PE release.

Executive summary

The main findings of the performance tests are as follows:

- The fastest way the syslog-ng PE application can receive log messages from the network is using plain TCP transport with the `network()` source driver.

- By default, syslog-ng PE runs in multithreaded mode to scale to multiple CPUs or cores for increased performance.

  Depending on the exact syslog-ng PE configuration, environment, and other parameters, syslog-ng PE is capable of processing:

  - over 635,000 messages per second (over 235 MB of data per second) when receiving messages from multiple connections and storing them in text files,

  - over 615,000 messages per second (over 230 MB of data per second) when receiving messages from multiple secure (TLS-encrypted) connections and storing them in text files.

- The tests were performed using real log messages of 135-2100 bytes and of varying complexity, obtained from real machines. The tests ran for 10 minutes.

- PatternDB only works on one thread. Therefore, using PatternDB can significantly decrease performance depending on how many messages match a pattern in the specified ruleset. If a large amount of messages are being received but the specified pattern matches a small number of messages, there is no significant performance degradation.

- Using disk buffer on the client side can significantly degrade performance. This is especially the case if relays that are heavily loaded are used together with disk buffer enabled on the relays. Performance can also be degraded if lots of messages are received on the syslog-ng server, and are forwarded through the network (for example, to a log analyzer or SIEM), and therefore disk buffer is enabled on the server as well.

- Higher `stats_level` decreases the performance. For example, `stats_level(2)` means -10% in performance.

- Simple filtering (for example, filtering on facility or tag) has no significant impact on performance. However, regular expressions, even simple ones, slightly decrease the message-processing rate, by about 15%.

- Results of the performance tests show a standard deviation of +-5%. This means that when comparing measurements of a particular dimension (for example, non-encrypted versus encrypted log processing), a difference of 5% does not necessarily indicate a real difference in performance.

Optimizing multithreaded performance

Sources:

File sources scale based on the number of files that the syslog-ng PE is reading. If there are 10 files all coming to the same source, then that source can use 10 threads, one thread for each file.

> ⓘ NOTE:
>
> When collecting log messages from multiple files, the file source is a `wildcard-file()` source.

TCP-based network sources scale based on the number of active connections. This means that if there are 10 incoming connections all coming to the same source, then that source can use 10 threads, one thread for each connection.

> ⓘ NOTE:
>
> UDP-based network sources do not scale by themselves because they always use a single thread. If you want to handle a large number of UDP connections, it is best to configure a subset of your clients to send the messages to a different port of your syslog-ng server, and use separate source definitions for each port.

Figure 1: How multithreading works — sources


```
log {
    source(s_tcp);
    source(s_udp);
    source(s_file);
    filter(f_filter);
    .
    .
    .
}
```
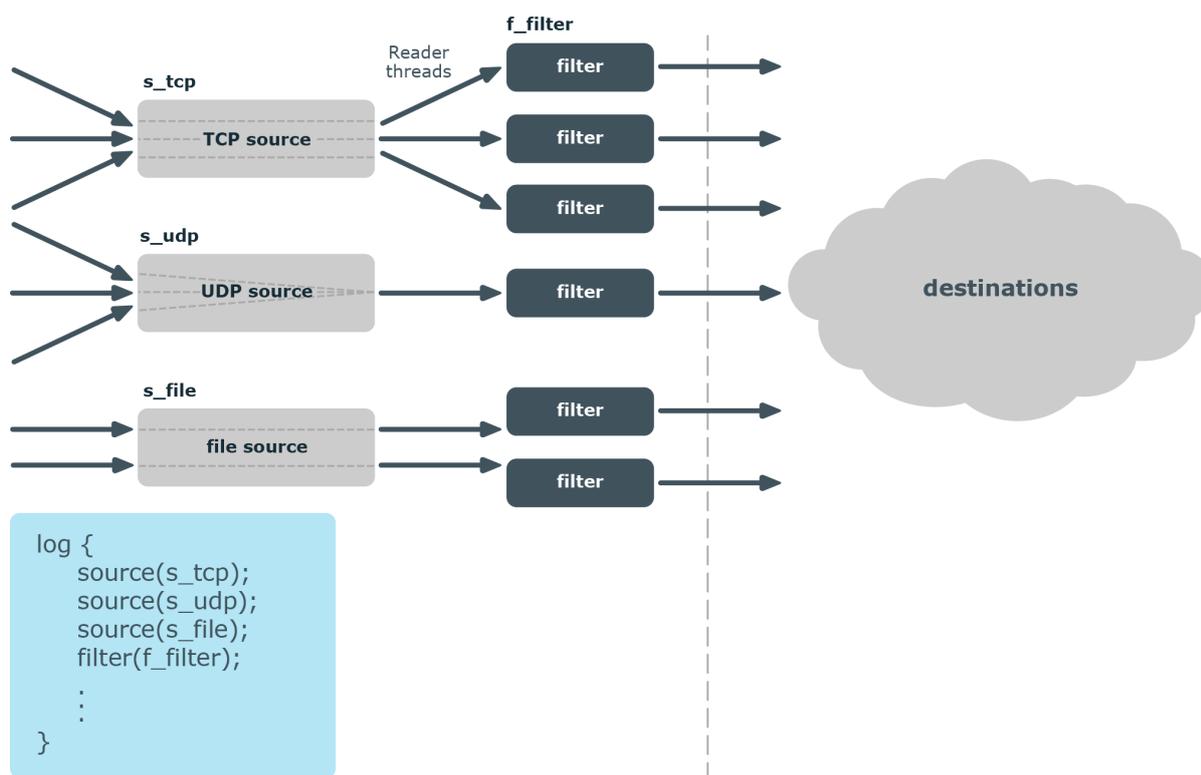
Message processors:

Message processors — such as filters, rewrite rules, and parsers — are executed by the reader thread in a sequential manner.

For example, if you have a log path that defines two sources and a filter, the filter will be performed by the source1 reader thread when log messages come from source1, and by the source2 reader thread when log messages come from source2. This means that if log messages come from both source1 and source2, they will both have a reader thread and that way filtering will be performed simultaneously.

> ❶ NOTE:
>
> This is not true for PatternDB because it uses message correlation. When using PatternDB, it runs in only one thread at a time, and this significantly decreases performance.

Destinations:

In syslog-ng, every destination has a writer thread. To improve scaling on the destination side, use multiple destinations instead of one.

For example, when sending messages to a syslog-ng server, you can use multiple connections to the server if you configure the syslog-ng server to receive messages on multiple ports, and configure the clients to use both ports.

When writing the log messages to files, use macros in the filename to split the messages to separate files (for example, using the ${HOST} macro). Files with macros in their filenames are processed in separate writer threads.

Figure 2: How multithreading works — destinations
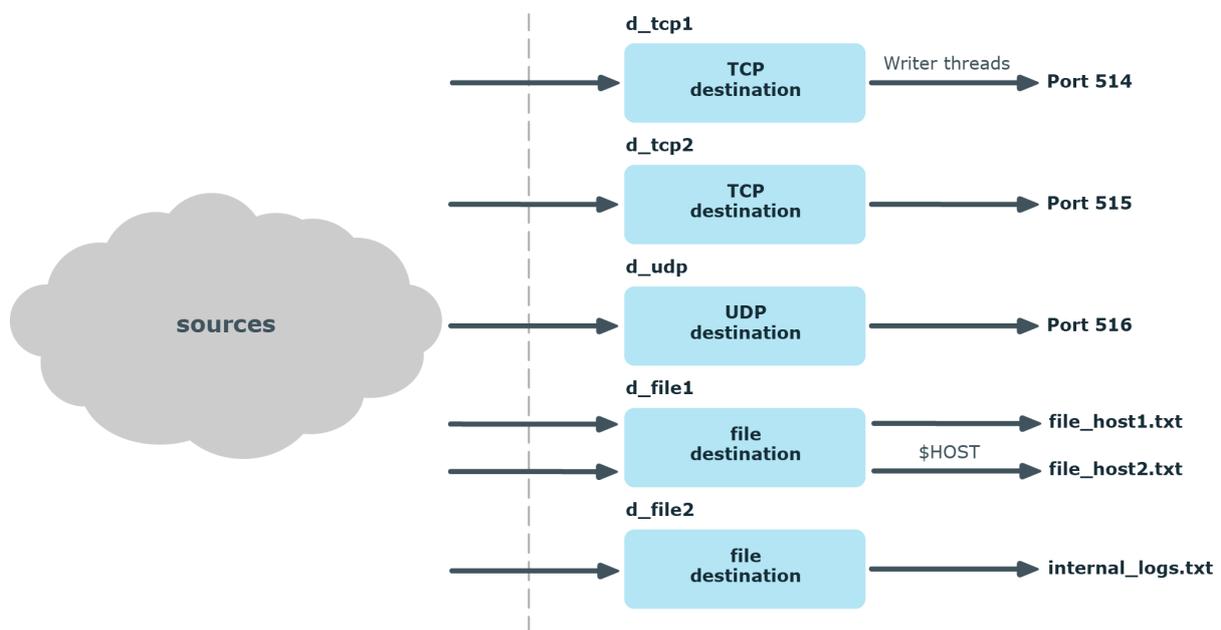
Use cases for performance improvement

Improving performance with lots of connections:

If there are several thousand active connections simultaneously, it is advised to place relay syslog-ng-s on another computer in front of the syslog-ng server. Switching between active connections is time-consuming, while the amount of incoming messages is usually not significant. This problem is solved by using relays, since they are collecting the logs. The syslog-ng solution can handle lots of log messages sent from a few connections easily.

In the case of non-encrypted connections, it is best to handle no more than 5,000 connections per syslog-ng PE application. While in the case of secure (TLS-encrypted) connections, a single syslog-ng PE application should handle no more than 1,000 connections.

When your environment has more connections than the numbers advised above, then our recommendation is that those should be handled by more than one syslog-ng PE application. For example, in the case of secure connections, when there are 2,000 connections, employ two relays, each of them running one syslog-ng PE application and handling 1,000 connections.

Note, however, that the exact capacity of syslog-ng PE depends on your particular circumstances (such as hardware performance, number and complexity of log messages, and so on), and needs to be measured on an individual basis. The recommendations indicated here are valid in an environment that handles a maximum of 200,000 EPS.

Filtering messages:

It is advised to use the simplest filters when filtering incoming messages. If a message can be filtered with several types of filters, check the measured data. For example, if a message is filtered with a `regexp`, the performance of syslog-ng can drop down to 85% of the original performance level. Whereas if the `tag` or `facility` filters are used, the decrease in performance is between 1-5%.

When using multiple filters one after the other, or connecting filters with the logical AND/OR operators, the order of filters has a significant impact on performance. We recommend to put those filters in front that are the most likely to match the incoming log messages (if known).

Identifying bottlenecks:

It is advised to first identify the bottlenecks if the performance of syslog-ng seems to be too low. For example, if syslog-ng is writing on slow disks at the destination side, upgrading the source-side device will not help increasing the performance.

Configuration guidelines

Log messages can be collected and processed at a faster rate in the latest version of syslog-ng Premium Edition compared to version 6 LTS and earlier versions but several configuration aspects will affect the rate at which log messages are collected and stored. The following tables show the results of tests performed with syslog-ng PE version 7.0. Taking the following factors into consideration will optimize syslog-ng Premium Edition performance:

Number of network connections:

In a multithreaded environment, an increase in connections will have no significant impact on the rate at which syslog-ng PE processes log messages.

Table 1: Number of network connections

Number of Connections	Messages Per Second	Average Data Rate (MB/sec)
10	640,000	240
50	550,000	205
100	530,000	200
200	545,000	205

Configuration: path – TCP, destination – multiple files (using macros), message size: 400 bytes

Encrypted log transfer:

The syslog-ng PE application uses the Transport Layer Security (TLS) protocol to encrypt the communication. TLS also allows the mutual authentication of the host and the server using X.509 certificates.

Table 2: Encrypted log transfer — 10 connections

	Legacy syslog		IETF syslog	
	Not Encrypted	TLS Encryption	Not Encrypted	TLS Encryption
Messages per second	640,000	620,000	65,000	65,000
Average data rate (MB/sec)	240	230	35	35

Configuration: path – TCP, multithreaded, 10 connections, destination – multiple files (using macros), message size: 400 bytes

Table 3: Encrypted log transfer — 100 connections

	Legacy syslog		IETF syslog	
	Not Encrypted	**TLS Encryption**	**Not Encrypted**	**TLS Encryption**
Messages per second	520,000	565,000	60,000	60,000
Average data rate (MB/sec)	195	210	30	30

Configuration: path – TCP, multithreaded, 100 connections, destination – multiple files (using macros), message size: 400 bytes

Type of storage:

The syslog-ng PE application can:

- Send syslog messages through the network to a syslog-ng PE server.
- Store log messages in a database.
- Store log messages in the Hadoop Distributed File System (HDFS).
- Send log messages to Elasticsearch.

Table 4: Type of storage — 10 connections

Type of Storage	Messages Per Second	Average Data Rate (MB/sec)
Plain text file	270,000	100
Multiple plain text files (using macros, with log messages divided by hostname)	640,000	240
Network destination — legacy syslog	250,000	95
Database destination — MongoDB	In the case of MongoDB destinations, performance is influenced by a number of criteria unrelated to syslog-ng. If you need information on MongoDB-related performance measurements, contact One Identity. If you are an existing customer, contact-support;. Otherwise, contact your Pre-Sales Engineer.	
Database destination — SQL	In the case of SQL destinations, performance is influenced by a number of criteria unrelated to syslog-ng. If you need information on SQL-related performance measurements, contact One Identity.	

Type of Storage	Messages Per Second	Average Data Rate (MB/sec)
	If you are an existing customer, contact-support;. Otherwise, contact your Pre-Sales Engineer.	

Configuration: path – TCP, multithreaded, 10 connections, message size: 400 bytes

Table 5: Type of storage — 100 connections

Type of Storage	Messages Per Second	Average Data Rate (MB/sec)
Plain text file	410,000	155
Multiple plain text files (using macros, with log messages divided by hostname)	505,000	190
Network destination — legacy syslog	245,000	90
Database destination — MongoDB	In the case of MongoDB destinations, performance is influenced by a number of criteria unrelated to syslog-ng. If you need information on MongoDB-related performance measurements, contact One Identity. If you are an existing customer, contact-support;. Otherwise, contact your Pre-Sales Engineer.	
Database destination — SQL	In the case of SQL destinations, performance is influenced by a number of criteria unrelated to syslog-ng. If you need information on SQL-related performance measurements, contact One Identity. If you are an existing customer, contact-support;. Otherwise, contact your Pre-Sales Engineer.	
HDFS	110,000	40
	Note: Processing speed is heavily influenced by the number of HDFS data nodes in use. When syslog-ng writes multiple files to HDFS, and Hadoop places these on different data nodes, then processing speed might increase in proportion to the number of data nodes used (not necessarily in a linear fashion). The data provided here shows performance in the case of a single data node.	
Elasticsearch	1,260 (with flush_limit(1))	1 (with flush_limit(1))

Type of Storage	Messages Per Second	Average Data Rate (MB/sec)
	9,700 (with `flush_limit(5000)`)	5 (with `flush_limit(5000)`)

Configuration: path – TCP, multithreaded, 100 connections, message size: 400 bytes

Number of files and directories when reading log messages from multiple plain text files:

When reading log messages from a set of files, the number of directories and the number of files per directory used have no significant impact on performance.

Table 6: Number of files and directories — using the `inotify` monitor method

Number of Directories	Number of Files Per Directory	Messages Per Second	Average Data Rate (MB/sec)
1	1	110,000	45
	10	175,000	70
	100	150,000	60
10	1	180,000	70
	10	150,000	60
	100	130,000	50
100	1	150,000	60
	10	130,000	50
	100	130,000	50

Configuration: path – TCP, multithreaded, `monitor-method(inotify)`, File source message size: 400 bytes

Table 7: Number of files and directories — using the `poll` monitor method

Number of Directories	Number of Files Per Directory	Messages Per Second	Average Data Rate (MB/sec)
1	1	110,000	45
	10	165,000	65
	100	150,000	60
10	1	175,000	70
	10	150,000	60
	100	130,000	50

Number of Directories	Number of Files Per Directory	Messages Per Second	Average Data Rate (MB/sec)
100	1	150,000	60
	10	130,000	50
	100	125,000	50

Configuration: path – TCP, multithreaded, `monitor-method(poll)`, File source message size: 400 bytes

Disk buffer:

The syslog-ng Premium Edition stores messages on the local hard disk if the central log server or the network connection to the server becomes unavailable.

Table 8: Disk buffer

	Without Disk Buffer	Reliable	Normal
Messages per second	345,000	40,000	60,000
Average data rate (MB/sec)	130	15	20

Configuration: path – TCP, multithreaded, disk buffer: 1000 MB, 100 connections, message size: 400 bytes

Log pre-processing:

Depending on the type of pre-processing, the rate at which syslog-ng PE collects messages can vary. Rewriting, using parsers, as well as pattern recognition processing through PatternDB have a significant impact on the message processing rate. Regular expressions have only a light impact, while facility filtering and tag filtering have virtually no impact at all.

Note that in a multithreaded environment, PatternDB has a particularly large impact on performance.

When combining multiple types of pre-processing, processing rate will drop below the processing rate of the slowest pre-processing method used.

Table 9: Log pre-processing — 10 connections

	Messages Per Second	Average Data Rate (MB/sec)
No pre-processing	680,000	255
Simple regexp (for example, matching a single string)	570,000	210

	Messages Per Second	Average Data Rate (MB/sec)
Facility filter	670,000	250
Tag filter	650,000	245
PatternDB (10 % of messages matched)	40,000	15
Simple rewrite (for example, rewrite hostname)	245,000	90
Python parser	35,000	15
JSON parser	40,000	25
kv parser	190,000	70
XML parser	15,000	20

Configuration: path – TCP, multithreaded, 10 connections, message size: 400 bytes

Table 10: Log pre-processing — 100 connections

	Messages Per Second	Average Data Rate (MB/sec)
No pre-processing	515,000	195
Simple regexp (for example, matching a single string)	510,000	190
Facility filter	500,000	185
Tag filter	530,000	200
PatternDB (10 % of messages matched)	35,000	15
Simple rewrite (for example, rewrite hostname)	360,000	135
Python parser	35,000	15
JSON parser	35,000	25
kv parser	140,000	50
XML parser	15,000	15

Configuration: path – TCP, multithreaded, 100 connections, message size: 400 bytes

The test environment

The test environment consisted of a single client and a server hardware, connected via a Gigabit switch. Note that in certain test runs, the client opened several separate connections to the servers to simulate real-life logging environments. The syslog-ng Premium Edition application was installed from the .run package.

Hardware parameters:

The client hardware had the following main parameters:

- 2x Intel® Xeon® Processor E5-2620 v3 (15M Cache, 2.40 GHz, 8 GT/s Intel® QPI, 6 cores)
- Hyperthreading disabled, turbo boost disabled
- 16 GB RAM
- 10 Gbps Ethernet
- HDD 500 GB
- Operating system: ubuntu-xenial amd64

The server hardware had the following main parameters:

- 2x Intel® Xeon® Processor E5-2620 v3 (15M Cache, 2.40 GHz, 8 GT/s Intel® QPI, 6 cores)
- Hyperthreading disabled, turbo boost disabled
- 16 GB RAM
- 10 Gbps Ethernet
- SSD 500 GB
- Operating system: ubuntu-xenial amd64

Performance improvement:

The following settings were used for performance improvement:

- Improving performance with lots of connections:

  max_connections = active_connections log_iw_size = number of active_connections * 1000 log_fetch_limit = 1000 flush_lines = 1000 log_fifo_size = log_iw_size * 2 use_dns = no keep_hostname = yes keep_timestamp = no

- Improving performance with a few connections but high amount of traffic:

  Source side:

  log_iw_size = number of active_connections * 100,000 log_fetch_limit = number of active_connections * 100,000

  Destination side:
  log_fifo_size = max_connections * (log_iw_size/number of active_connections)

flush_lines = 10,000 or greater

Resource usage:

The performance tests were carried out in multithreaded mode:

 threaded(yes)

One way to optimize the resource usage of syslog-ng PE is to limit the number of worker threads that syslog-ng uses. This helps prevent syslog-ng PE from using all available CPUs. You can limit the number of worker threads using the `--worker-threads` command-line option that sets the maximum total number of threads syslog-ng PE can use, including the main syslog-ng PE thread.

Note, however, that SQL sources and destinations, as well as Java destinations, such as Elasticsearch, HDFS, and Apache Kafka, always run in their own, separate threads. This means that the `--worker-threads` command-line option has no impact on them.

One Identity solutions eliminate the complexities and time-consuming processes often required to govern identities, manage privileged accounts and control access. Our solutions enhance business agility while addressing your IAM challenges with on-premises, cloud and hybrid environments.

Contacting us

For sales or other inquiries, visit https://www.oneidentity.com/company/contact-us.aspx or call +1-800-306-9329.

Technical support resources

Technical support is available to One Identity customers with a valid maintenance contract and customers who have trial versions. You can access the Support Portal at https://support.oneidentity.com/.

The Support Portal provides self-help tools you can use to solve problems quickly and independently, 24 hours a day, 365 days a year. The Support Portal enables you to:

- Submit and manage a Service Request
- View Knowledge Base articles
- Sign up for product notifications
- Download software and technical documentation
- View how-to-videos at www.YouTube.com/OneIdentity
- Engage in community discussions
- Chat with support engineers online
- View services to assist you with your product