
www.balabit.com

# FORWARDING LOG MESSAGES TO SPLUNK FROM SYSLOG-NG

## HOW TO CONFIGURE SYSLOG-NG PE TO COOPERATE WITH SPLUNK

March 08, 2017

# TABLE OF CONTENTS

# 1. PREFACE

Splunk is a popular search and analysis platform. Many users of Splunk also have syslog-ng deployed in their environments. This guideline describes some scenarios in which Splunk users can benefit from syslog-ng PE features and offers some technical guidance to optimize the syslog-ng configuration.

BALABIT

# TOP 5

WWW.BALABIT.COM

## SYSLOG-NG WITH SPLUNK USE CASES

### 1 COLLECTING LOGS FROM NETWORK DEVICES

Major router manufacturers transfer log messages using the syslog protocol, syslog-ng natively supports both versions RFC3164 and RFC5424. Using syslog-ng can improve the reliability log data collection from network devices.

### 2 FEEDING MULTIPLE ANALYSIS TOOLS

Feeding Multiple Analysis Tools - Organizations deploying Splunk usually have heterogeneous IT environments, often having different departments analyzing log data with different tools. syslog-ng can collect and flexibly route logs to multiple analysis tools.

### 3 LONG-TERM STORAGE OF LOGS

Organizations are required to archive data for compliance purposes, often for months or even years. syslog-ng's reduces storage costs and secures log files.

### 4 ADVANCED FILTERING ON CLIENTS

Many users use syslog-ng to filter log messages on clients to reduce network loads. sylsog-ng can reduce the data load on Splunk improving performance and reducing license costs.

### 5 VERY HIGH MESSAGE RATE LOG SOURCES

Network and security devices can generate large amounts of log messages. With its scalability, syslog-ng can meet the needs of the largest traffic environments. syslog-ng eliminates the need for complex design workarounds such as load balancers or forwarder instances.

# 2. USE CASE ONE

## Collecting Logs from Network Devices

Collecting and centralizing log messages from network devices such as routers is one of the most common deployments of syslog-ng with Splunk. Major router manufacturers like Cisco and Juniper use the syslog protocol to transfer log messages. The syslog-ng PE application natively supports the original syslog protocol RFC3164 (also known as legacy-syslog or BSD-syslog) and the new syslog protocol RFC5424 (also known as IETF-syslog). In addition, syslog-ng PE also supports variants of these protocols which are used by certain router manufacturers.

The most common configuration consists of a syslog-ng server which receives log messages via UDP or TCP. While Balabit strongly recommends that routers send log messages via TCP due to its greater reliability, many routers only transport logs via the inherently unreliable UDP or set to use UDP because of its lower resource requirements. Log messages on the syslog-ng server can be forwarded to Splunk over the more reliable TCP or stored into flat files to be read by Splunk. The benefits of this configuration are the following:

- Reliability: Using syslog-ng to collect and aggregate log messages from routers sending messages over UDP and then transfer messages over TCP will reduce message loss. It also provides a buffer for the data in the event that a network or server outage, or if Splunk is disabled for maintenance.
- Performance: Network devices can generate very high rates of events per second. Depending on its exact configuration, syslog-ng Premium Edition can collect more than 100,000 log messages per second from a single source.
- Flexibility: syslog-ng can filter, parse and rewrite log messages from network devices and turn them into more standard syslog messages.

To reduce message loss, particularly in the case of UDP transport, it is recommended, if feasible, that the syslog-ng server should be cabled directly to the network devices. With this combined solution, you can expect:

### Example 1. Receiving logs from UDP

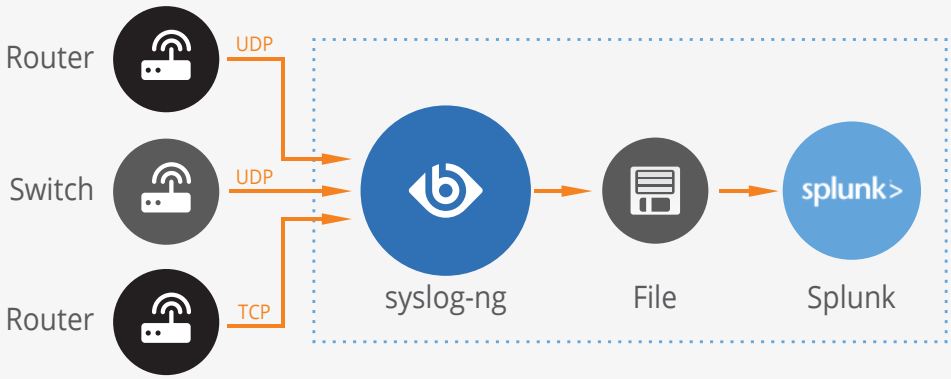


*Figure 1. Receiving logs from UDP*

The following example first adds an UDP network source, s_net. Next it adds a file destination called `d_splunk_files`, which saves logs in directories based on hostnames and filenames based on current month and day. Directories are automatically created. A log statement connects the source to the destination at the end of the example.

```
source s_net {
        udp();
};
destination d_files_splunk {
        file("/var/log/splunk/$HOST/$MONTH$DAY.log" create_dirs(yes));
};
log {
        source(s_net);
        destination(d_files_splunk);
};
```

## Example 2. Receiving logs from UDP using a relay

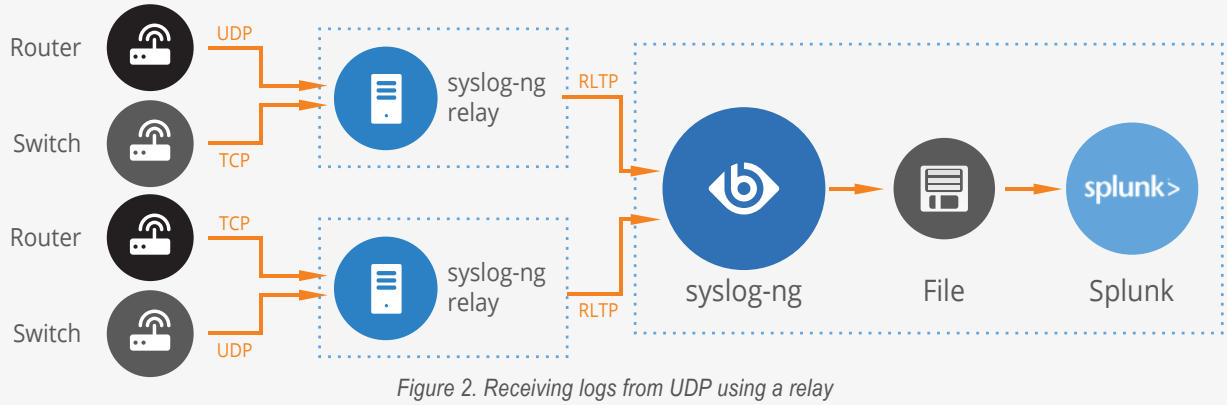


*Figure 2. Receiving logs from UDP using a relay*

The next example is from a larger network, where syslog-ng is installed at multiple locations receiving logs from local devices and forwarding to a central location. Here the local syslog-ng sends logs to a central syslog-ng server using RLTP™ (Reliable Log Transfer Protocol, a proprietary log transfer protocol for syslog-ng with additional reliability features), which forwards logs to Splunk using flat files.

The client side of the configuration is:

```
source s_net {
        udp();
};
destination d_rltp {
        syslog("192.168.1.1" port("4444")
                transport(rltp(tls_required(optional)))
                ip-protocol(4)
                disk-buffer( mem-buf-size(200000) disk-buf-size(2000000) reliable(yes) )
                tls(
                        peer-verify(required-trusted) ca_dir("/var/tmp/server/")
                        key_file("/var/tmp/client/client_priv.key")
                        cert_file("/var/tmp/client/client.crt")
                )
        );
};
log {
        source(s_net);
        destination(d_rltp);
};
```

The relevant part of the configuration at the central server is:

```
source s_rltp {
        syslog( ip("192.168.1.1") port("4444")
                transport(rltp(tls_required(optional)
                response_timeout(20) message_acknowledgement_timeout(10) )) ip-protocol(4)
                tls(
                        peer-verify(required-trusted)
                        ca_dir("/var/tmp/client/")
                        key_file("/var/tmp/server/server_priv.key")
                        cert_file("/var/tmp/server/server.crt")
                )
        );
};
destination d_files_splunk {
        file("/var/log/splunk/$HOST/$MONTH$DAY.log" create_dirs(yes));
};
log {
        source(s_rltp);
        destination(d_files_splunk);
};
```

Splunk recommends that network inputs, such as syslog messages, be persisted to files on disk. Splunk's file-input technology allows for reliable and resilient data collection that handles scenarios where Splunk may be disabled. The host name of the data source is an important piece of metadata that can be extracted by Splunk in multiple ways. One way is to extract it from the directory or file name, as seen above (the `$HOST` macro in the directory name). If you use this method, the parameters used (in Splunk) to set this are located within the `inputs.conf` file, specifically the `host_regex` and `host_segment` parameters. Another possibility is to have Splunk automatically extract the hostname directly from the log messages. In this case, the messages need to be classified as a `syslog` sourcetype or custom Splunk configurations will need to be applied.

# 3. USE CASE TWO

## Feeding Multiple Analysis Tools

Many organizations that deploy Splunk have existing log management and analysis tools. Some departments within the same company, such as the Network Operations group and the IT security group, may have use for the same data but prefer to use different analysis tools such as Security Information and Event Management (SIEM) solutions. In these environments syslog-ng is often used to collect and aggregate log messages and then forwarded to multiple destinations including a Splunk instance. syslog-ng supports more than 50 server platforms making it ideal in heterogeneous IT environments.

syslog-ng uses logpaths to define the sources and destination of log messages as well as the way in which they are transported. Logpaths can have one to many, many to one, or many to many relationships. For example, a user can specify that all log files generated by email servers are transported from clients to a central syslog-ng server via TCP with TLS encryption. Messages are then sent to both Splunk and a SIEM solution. Splunk has the ability to natively forward data to other systems in raw or syslog format using its own network output stream. Splunk can also forward data in custom formats such as CEF. One benefit of using syslog-ng for log routing is that it can forward events through an encrypted connection while also reformatting messages (using templates) to the format preferred by the given SIEM vendor.

### Example 3. Forwarding logs to a SIEM

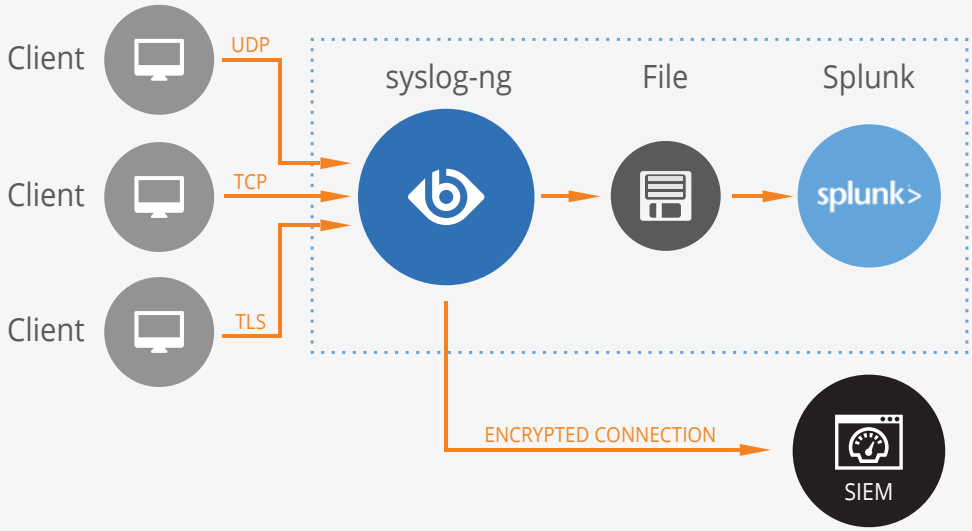


*Figure 3. Forwarding logs to a SIEM*

The following example extends the first one by adding multiple destinations to the configuration. It defines `d_siem` as an encrypted TCP destination. At the end, the log statement connects the source to both destinations, the file destination for Splunk and the network destination for the SIEM.

```
source s_net {
        udp();
};
destination d_files_splunk {
        file("/var/log/splunk/$HOST/$MONTH$DAY.log" create_dirs(yes));
};
destination d_siem {
        tcp("192.168.1.1" port("9000") tls(ca_dir("/opt/syslog-ng/etc/syslog-ng/ca.d"));
};
log {
        source(s_net);
        destination(d_files_splunk);
        destination(d_siem);
};
```

# 4. USE CASE THREE

## Long-term Storage of Logs

Depending on the type of log messages being collected, organizations are required to archive data for compliance purposes. Many data retention policies and regulations specify that log messages be stored in original format for several months or even years. If organizations do not need to analyze this data but simply must securely archive data, syslog-ng provides a cost-effective and convenient solution.

Users can specify the type of destination for archiving, text file, binary logstore, or SQL database. Output messages can be written to a specific file or set of files depending on certain criteria. The use of syslog-ng logstore facilitates both confidentiality and long term archiving. It uses compression for space saving and time stamping and encryption for tamper proof log storage.

### Example 4. Storing logs in compressed files

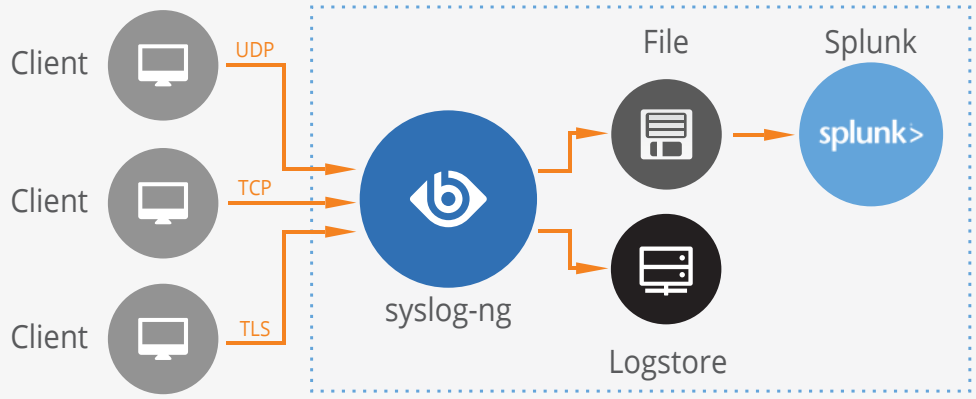


*Figure 4. Storing logs in compressed files*

The following configuration file example extends the first one by adding a long term destination to the configuration called `d_longterm`. One can delete flat files regularly once they are read by Splunk, while encrypted and compressed logstore files stay as long as needed by compliance regulations.

```
source s_net {
        udp();
};
destination d_files_splunk {
        file("/var/log/splunk/$HOST/$MONTH$DAY.log" create_dirs(yes));
};
destination d_longterm {
        logstore("/var/log/longterm/$MONTH$DAY.lgs" encrypt_certificate
        ("/opt/syslog-ng/etc/syslog-ng/keys/public-server-certificate.pem"));
};
log {
        source(s_net);
        destination(d_files_splunk);
        destination(d_longterm);
};
```

# 5. USE CASE FOUR

## syslog-ng PE Advanced Filtering on Clients to Reduce Data Load

Many users use syslog-ng to filter log messages on clients to reduce network loads. syslog-ng can filter out irrelevant data in the event that network capacity to remote clients is limited. When defining a log path, a user can insert a filter to route messages based on pre-defined criteria. Messages coming from the sources listed in the log statement and matching all the filters are sent to the listed destinations. To define a log path, add a log statement to the syslog-ng configuration file using the following syntax:

```
log {
        source(s1); source(s2); ...
        optional_element(filter1|parser1|rewrite1);
        optional_element(filter2|parser2|rewrite2);...
        destination(d1); destination(d2); ...
        flags(flag1[, flag2...]);
};
```

syslog-ng can handle embedded log statements (also called log pipes). Embedded log statements are useful for creating complex, multi-level log paths with several destinations and use filters, parsers, and rewrite rules. For example, if you want to filter your incoming messages based on the facility parameter, and then use further filters to send messages arriving from different hosts to different destinations, you would use embedded log statements. This advanced filtering enables users to finely tune the number and type messages sent to Splunk instances to be indexed lowering network capacity requirements.

### Example 5. Filtering log messages

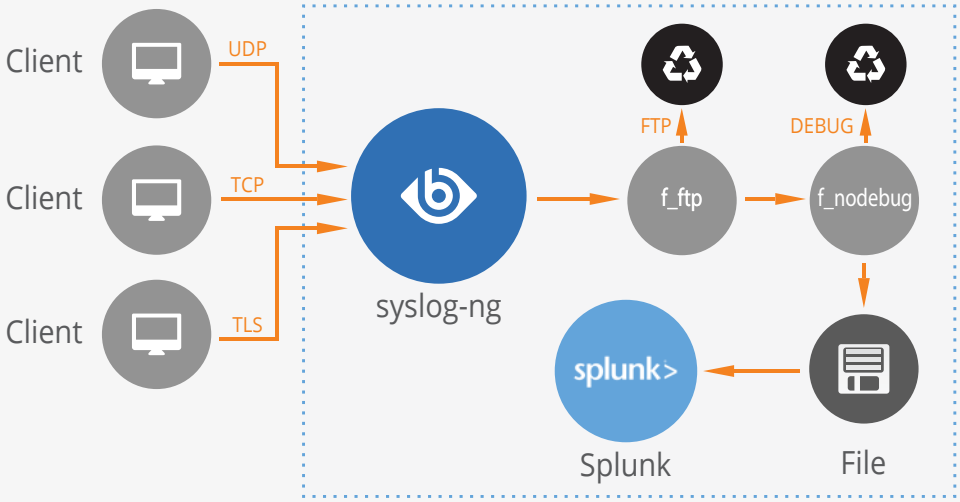


*Figure 5. Filtering log messages*

The following configuration file example extends the first one by adding filters to the configuration. The first one is called `f_proftpd` and discards any messages from the application called proftpd. The second one is called `f_nodebug`. This filter discards debug messages from the logs, which are only necessary under very special circumstances, but can increase log volume considerably.

```
source s_net {
        udp();
};
destination d_files_splunk {
        file("/var/log/splunk/$HOST/$MONTH$DAY.log" create_dirs(yes));
};
filter f_program { not program('proftpd');};
filter f_nodebug { level(info...emerg);};
log {
        source(s_net);
        filter(f_proftpd);
        filter(f_nodebug);
        destination(d_files_splunk);
};
```

# 6. USE CASE FIVE

## syslog-ng PE Very High Message Rate Log Sources

Network and security devices such as routers, switches, firewalls, and Intrusion Detection Systems can generate large amounts of log messages. Depending on its configuration, syslog-ng can collect more than 100,000 messages from a single source and can scale to more than 650,000 messages per second from multiple sources with multithread processing. With this scalability, syslog-ng can meet the needs of the largest traffic environments.

syslog-ng also offers several features that help manage high volumes of log messages. The `throttle` and `flow-control` features enable users to set the limit of messages being sent and received by syslog-ng. The syslog-ng application can stop reading messages from its sources if the destinations cannot process the sent messages. This feature is called flow-control. The throttle feature sets the maximum number of messages sent to the destination per second.

### Example 6. Receiving high-rate of messages

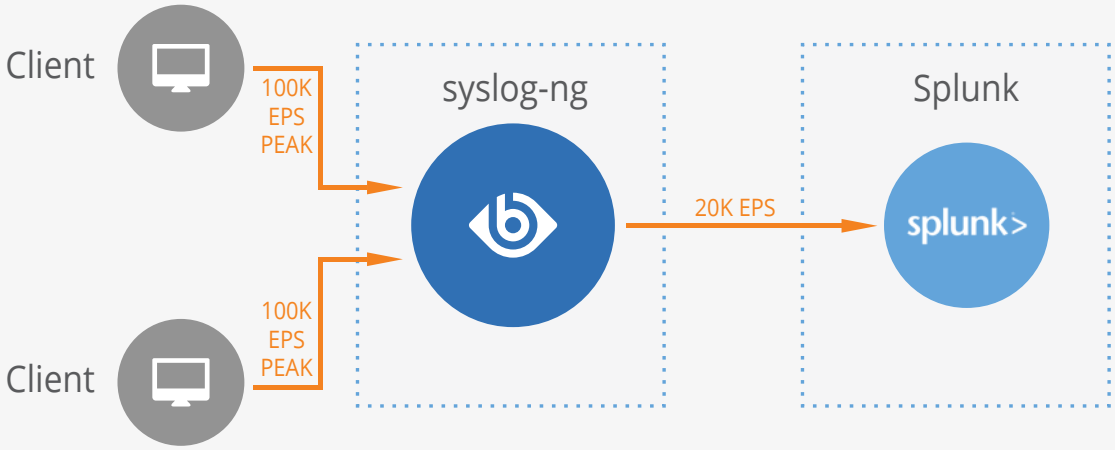


*Figure 6. Receiving high-rate of messages*

The following example modifies the first example by using a TCP destination instead of flat files. The destination is configured to use a diskbuffer to avoid losing logs and throttles outgoing message rate to 20,000 messages a second to avoid high peaks at the receiving end.

```
source s_net {
        udp();
};
destination d_net_splunk {
        tcp("192.168.1.1" throttle(20000) log_disk_fifo_size(4194304));
};
log {
        source(s_net);
        destination(d_net_splunk);
};
```

# 7. SUMMARY

This paper has shown you how syslog-ng handles network connections, what are the problems of having to handle many connections, and also how to overcome these problems to increase the performance and reliability of your logging infrastructure.

## 7.1. ABOUT BALABIT

Balabit's Contextual Security Intelligence (CSI) platform protects organizations from threats posed by the misuse of high risk and privileged accounts. Solutions include reliable system and application Log Management with context enriched data ingestion, Privileged User Monitoring and User Behavior Analytics. Together they can identify unusual user activities and provide deep visibility into potential threats. Working in conjunction with existing control-based strategies, Balabit enables a flexible and people-centric approach to improve security without adding additional barriers to business practices. Founded in 2000, Balabit has a proven track record, with 25 Fortune 100 customers and more than 1,000,000 corporate users worldwide.

To learn more about commercial and open source Balabit products, request an evaluation version, or find a reseller, visit the following links:

- The syslog-ng homepage https://www.balabit.com/network-security/syslog-ng
- Contact us and request an evaluation version https://www.balabit.com/network-security/syslog-ng/central-syslog-server/callback
- Find a reseller https://www.balabit.com/partnership/commercial/

All questions, comments or inquiries should be directed to <info@balabit.com> or by post to the following address: Balabit S.a.r.l. 1117 Budapest, Alíz Str. 2 Phone: +36 1 398 6700 Fax: +36 1 208 0875 Web: https://www.balabit.com/



The latest version is always available at the Balabit Documentation Page.