



Technical White paper

www.balabit.com

Distributed syslog architectures with syslog-ng Premium Edition

Synopsis: The advantages of using syslog-ng Premium Edition to create distributed system logging architectures

Status: Released

Version: 1.0

Date: 07/30/2007



Table of contents

Preface.....	3
Introduction.....	4
What is system logging.....	4
What is distributed system logging.....	5
Why use syslog-ng as a solution.....	5
Deployment scenarios.....	7
Using syslog-ng on the end systems.....	7
Using relays.....	8
A hybrid approach.....	9
Other important features.....	10
Output data into various formats.....	10
Select important messages.....	10
Control the rate of messages.....	10
Works in both IPv4 and IPv6 environments.....	10
Collect logs from Microsoft Windows.....	10
Comparing syslogd and syslog-ng.....	11
What does syslog-ng PE offer over syslogd?.....	11
What does syslog-ng PE offer over syslog-ng OSE?.....	11



Preface

This paper discusses the advantages of using syslog-ng Premium Edition to implement distributed syslog architectures in heterogeneous environments. The document is recommended for technical experts and decision makers working on implementing centralized logging solutions, but anyone with basic networking knowledge can fully understand its contents. The procedures and concepts described here are applicable to version 2.1 of syslog-ng Premium Edition.

This paper is organized into the following sections:

- *Introduction* explains the basics of system logging and central logging, the problems of the standard syslog solution, and describes how you can use syslog-ng Premium Edition (syslog-ng PE) to solve these problems – including collecting event logs from Microsoft Windows operating systems.
- *Deployment scenarios* describes the most common network layouts used to implement system logging solutions, detailing the benefits syslog-ng PE provides in these situations.
- *Other important features* discusses further features of syslog-ng PE that can come handy for you when designing and implementing your system logging architecture.
- *Comparing syslogd and syslog-ng* gives you a summary of the differences between syslogd, the standard system logging solution used by most UNIX-like operating systems; syslog-ng OSE, the open source version of syslog-ng; and syslog-ng PE, the commercial version of syslog-ng.



Introduction

What is system logging

Operating systems, applications, and network devices generate text messages of various events that happen to them: a user logs in, a file is created, a network connection is opened to a remote host, etc. These messages, called log messages, are usually stored in a file on the local hard disk of the system. Storing the logs only on the host where the messages were created is problematic for several reasons:

- If the system is compromised, the attacker can access the logs and delete or manipulate them, erasing the tracks.
- You cannot easily access the logs of multiple hosts at once. This makes it difficult to see the big picture of what is happening on your network, making both maintenance and forensics difficult.
- Legal policies and regulations (e.g., SOX, Basel II, PCI) may require you to collect and archive the log messages. This task is much more difficult if the logs are located on many different devices.

The aim of central system logging is to collect the log messages to a single, central log server. The most straightforward method to transfer log messages is to use the legacy syslog protocol, which is supported by virtually every device and application: you can collect logs from servers, firewalls, network devices like routers or Wifi access points. Syslog has been implemented and is available on virtually every UNIX-like operating system, and has become the de-facto standard of remote logging. (The syslog protocol is described in RFC 3164, available at [ftp://ftp.rfc-editor.org/in-notes/rfc3164.txt](http://ftp.rfc-editor.org/in-notes/rfc3164.txt).) However, the syslog protocol has several deficiencies, including:

- It sends the messages over an insecure connection in unencrypted, plain text format.
- It uses the unreliable UDP transport protocol, which does not ensure that a message actually arrives to the destination.
- There is no way to know if a message is lost. Messages can get lost on the network, or if the central server or an intermediate networking device becomes overloaded and cannot process the incoming messages.
- There is no way to identify the sender of the message; it is easy to create fake messages and send them to the server.

Traditional syslog solutions can lose a tremendous amount of messages because of using the UDP transport protocol. Measurements have shown that when using UDP to transfer messages to a remote server, syslog can lose over ninety-nine percent* of the messages under high load. This ratio can get even worse if a single server has to collect the logs of a large number of clients – meaning that only a fraction of the messages arrive to the central server. It is obvious that UDP is not suitable to transfer important information, like log messages. If you want to take logging seriously, you have to use a solution based on the TCP protocol, such as syslog-ng.

Central logging is a big problem on Windows operating systems as well, because Windows does not have anything similar to syslog – remote logging is not part of the Windows operating systems. A typical solution on Windows is to share the folder that stores the log files, with the central server periodically downloading the files. However, this solution has important security aspects, because the messages are not transferred instantly to the server, leaving time for an attacker to manipulate them.

* Marcus J. Ranum: System Logging and Log Analysis, http://www.ranum.com/security/computer_security/archives/logging-notes.pdf



What is distributed system logging

Distributed system logging is essentially central logging on a global scale: where central logging solutions typically collect the logs of a local network (e.g., a site, an office, or a particular facility), distributed logging collects the logs from several different facilities of an organization. These facilities may be spread around the world, meaning that log messages must be transferred over the Internet – so reliability and encryption become a must.

You have to create a distributed syslog architecture if you want to collect your syslog messages to a single server from several different locations. A typical example is a company or organization that has offices in different cities, but wants to store the logs of every facility at the headquarters. It is not unusual for the facilities to connect to the Internet via Wide Area Network (WAN) links offering only limited bandwidth. Implementing a distributed syslog solution simplifies log management, analysis, and archiving, often required for policy compliance.

When implementing a distributed system logging infrastructure, you must ensure that the following requirements are fulfilled:

- The messages sent by the end systems arrive to the server (reliable transfer).
- No messages are lost when the network or the server is temporarily down (disk buffer).
- Communication to the central server is encrypted, so third parties cannot gain access to sensitive data (SSL/TLS support).
- The identity of the end systems is verified, so it is not possible to inject fake log messages into the central logs.

The syslog-ng Premium Edition application fulfills all the above requirements, as detailed in the following sections.

Why use syslog-ng as a solution

The syslog-ng Premium Edition application allows you to collect log messages from your devices to a central syslog server in a reliable, secure way. As syslog-ng PE supports a wide variety of operating systems (including Solaris, AIX, HP-UX, and Windows), it is especially suited for organizations having a widely distributed, heterogeneous network. The syslog-ng PE application is suitable for every organization, ranging from small companies with a few offices to multinational enterprises or governmental institutes.

The syslog-ng PE application can run on your end systems in client mode, replacing the original syslog implementation of the host, and transfer the log messages generated on the system to the central server. When used on the end systems, syslog-ng PE provides the following benefits:

- Your log messages are transferred using the reliable TCP protocol in an authenticated, SSL-encrypted channel.
- You do not lose messages during network or system outages, because syslog-ng PE can store the unsent messages on the local hard disk until the server becomes available again.
- You can create multiple, independent logging centers to store copies of your logs, because syslog-ng can send the log messages to multiple destinations.
- You can install syslog-ng PE on a wide variety of platforms, because it supports several operating systems and hardware architectures. For the latest list of supported platforms, visit <http://www.balabit.com/network-security/syslog-ng/central-syslog-server/>

Your central log collector server can also run syslog-ng PE to accept, sort, and store the incoming log messages. Using syslog-ng PE as a central syslog server has the following benefits:



- The syslog-ng PE application can work in concert with your System Integrity Monitoring (SIM) and log analyzer solutions by forwarding the messages to the SIM, or providing a backend that stores the incoming messages that the SIM can index.
- You can even limit the number of messages sent to the SIM per second. This prevents the SIM from being overloaded, and when used together with the disk buffering capability of syslog-ng PE, it can level the load on the SIM – a useful feature to have, as many SIM products may drop messages arriving over a certain rate.
- The syslog-ng PE server can authenticate the clients sending the logs.
- The syslog-ng PE server can accept log messages from many different channels, including legacy (UDP-based) syslog, TCP-based connections, and secure, SSL encrypted connections.
- You can store your log messages in customizable output format in SQL databases (Oracle, PostgreSQL, MySQL, SQLite) and plain text files.
- You can easily add site-specific customizations to the log analysis pipeline, including custom scripts, filtering, archival, etc.

You can also create relays to collect the logs from platforms that run only the legacy syslog implementation, and enjoy the benefits of syslog-ng PE between the relay and the central server: reliable and secure message transfer, disk buffering to avoid server and network outages, and more.



Deployment scenarios

The following sections describe the most common ways to deploy syslog-ng into your network infrastructure.

Using syslog-ng on the end systems

The most straightforward scenario is to install syslog-ng PE on the devices that create the logs you want to collect: on the servers and other devices. That way the devices can send their logs directly to the central syslog server.

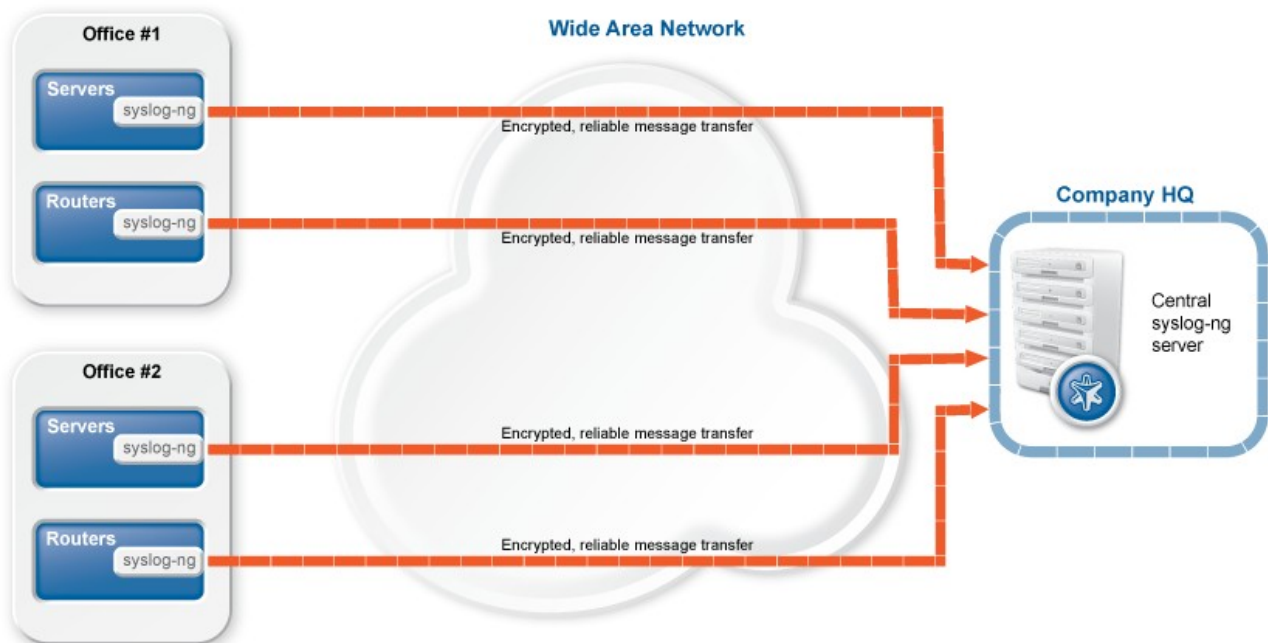


Figure 1. Logging directly to the central syslog-ng server

Use syslog-ng PE on the end systems in the following situations:

- If end-to-end encryption is required between the end system and the central syslog-ng server. That secures the entire remote logging communication, while relays provide only encryption between the relay and the central syslog-ng server.
- If you want to authenticate every individual end system that sends logs to the central syslog-ng server, or if you want to authenticate the server to ensure that your log messages are sent only to their proper destinations. The syslog-ng PE application supports the authentication of both the central server and the client hosts using X.509 certificates.
- If your end systems have hard disk and you want to use disk buffer to ensure that no messages are lost. If the central syslog server or the network becomes unavailable, the end systems buffer the log messages to the local hard disk.
- If you want to transfer your logs using the reliable TCP protocol, and the native syslog solution of the device supports only the UDP protocol.
- If you want to collect logs from Microsoft Windows systems. The Windows operating systems cannot send their logs to a remote server by default, you have to use a third-party application like syslog-ng Agent for Windows.
- If you want to preprocess the log messages on the end system, and send only the important messages to the central syslog-ng server.

Note that most networking devices like routers or switches do not have hard disks, and do not allow you to install third-party applications on them. In these cases you have to use a local relay to securely transfer the



logs of these devices.

To enable you to install syslog-ng PE on every end system you may need, among others, the following architectures and operating systems are supported : x86, x86_64, and SUN UltraSPARC; Linux, BSD, Solaris, AIX, HP-UX, and Microsoft Windows.

For the latest list of supported platforms, visit the syslog-ng Premium Edition webpage at <http://www.balabit.com/network-security/syslog-ng/central-syslog-server/>

Installing syslog-ng PE on the end systems has lots of advantages, but installing and configuring it on every host of a large network may require significant administrative and maintenance work. Therefore, using relays may be preferred for large sites.

The main points of using syslog-ng PE on the end systems are the following:

- The communication between the end systems and the central server is completely encrypted and based on the reliable TCP protocol.
- The server can verify the identity of every end system.
- The end systems have their own disk-buffer, resulting in higher fault tolerance.
- You must install the syslog-ng Agent for Windows to collect logs from Windows-based systems.

Using relays

Relay devices collect the logs of the local network, for example, a relay can collect the logs of a site or a subnet. Configure the log devices to send the logs to these local relays; the relays forward the log messages to the central syslog server. Since the relays and the log devices are located close to each other – possibly on the same local network – there is less chance for losing messages even if the log devices can use only the unreliable legacy syslog protocol. The relays run syslog-ng PE, and can send the logs messages to the central server in a reliable, encrypted channel. If the server or the network connection is unavailable, the relays save the log messages to the hard disk, ensuring that no messages are lost.

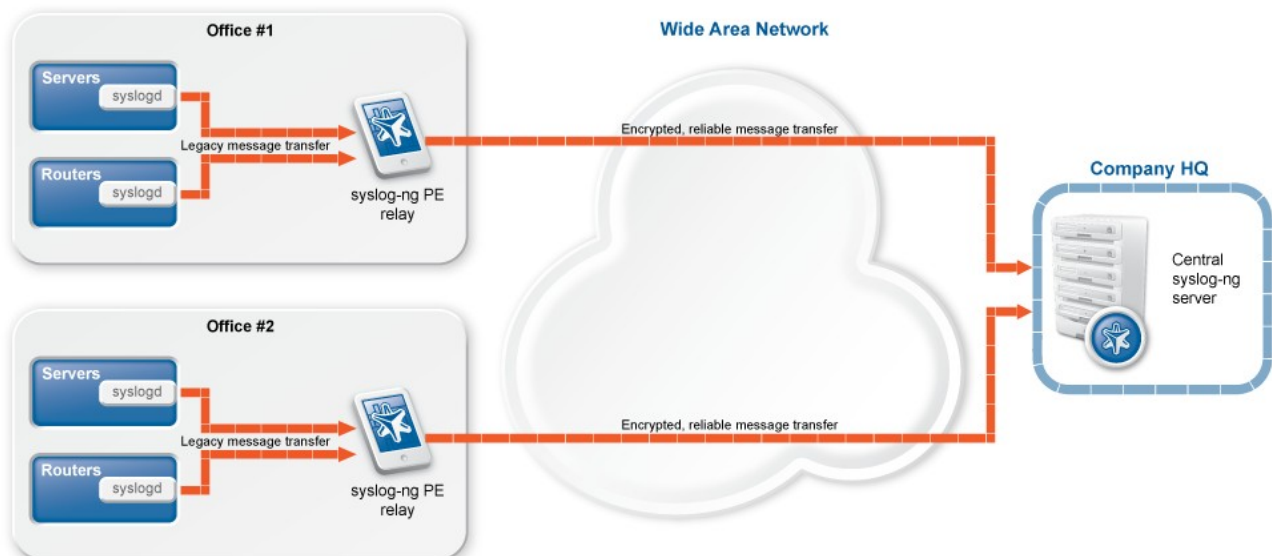


Figure 2. Using relays to transfer the messages

Use syslog-ng PE relays in the following situations:

- If your end systems do not support installing third-party applications. This is the case with most routers, switches, and other networking devices.
- Your local network is reliable and trustworthy, and you want to simplify the management of the logging infrastructure.



Make sure to size your relay according to its expected load: it should be able to buffer to disk the log messages of every relayed end system even during the longest downtime of the central server or the network connection. For the relay, use a hardware that is equipped with a redundant power supply and an uninterruptible power supply (UPS) to minimize the chance of relay outages.

The main points of using syslog-ng PE relays are the following:

- The end systems and the relay may communicate using the legacy syslog protocol – deploying the relay close to the end systems lessens the disadvantages of this protocol.
- The communication between the relay and the central syslog server is encrypted and based on the TCP protocol.
- If the central syslog server or the network becomes unavailable, the relay buffers the log messages to the local disk without affecting the end systems.

A hybrid approach

To minimize the chance of losing log messages, you can combine the two scenarios discussed above:

- Where possible, install syslog-ng PE directly on the end system. These end systems can either send their logs directly to the central syslog-ng server, or to the local relay.
- Deploy relays to every site or subnet that contains end systems that cannot run syslog-ng PE.

This approach takes the best of both worlds:

- You can encrypt and authenticate most of the log traffic, only the short path between the problematic end systems and the local relay is unencrypted.
- Both the relay and the end systems running syslog-ng PE can store the messages locally in the disk-buffer to avoid network and server outages. That minimizes the impact of any problem with relay, the server, and the network.

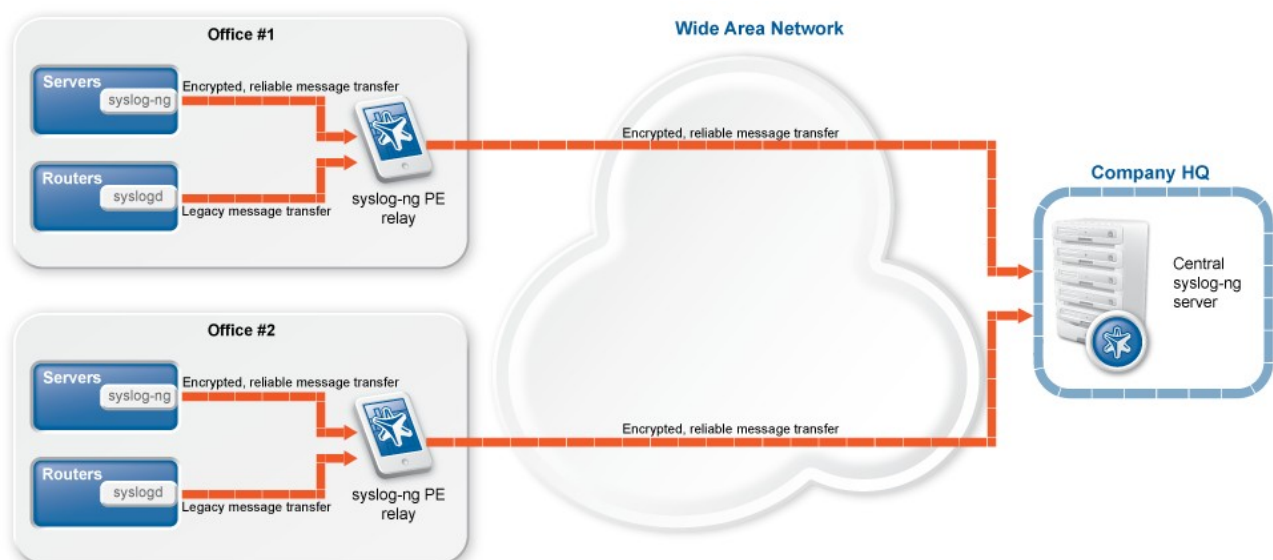


Figure 3. Using the hybrid approach



Other important features

This section highlights some of the features of syslog-ng PE that were not discussed in detail so far, but are useful to know about.

Output data into various formats

The syslog-ng application does not limit what you can do with your log messages: it is meant to provide you with the most effective way to collect them. You can store your logs in files, databases, or pass them to a log analyzing application: syslog-ng PE can customize the messages into the format you want. You can even reorganize the contents of the log messages if you are not content with the original message format – or if it makes your log analyzing application more effective.

Select important messages

You can use various filters – ranging from very simple to really complex ones – to select messages based on their content, source, or other parameters. This is useful if you do not want to send every message to the central server, or you have to process messages differently based on their content. The syslog-ng application can dynamically create directories, files, and database tables using macros.

Control the rate of messages

You can control the number of messages syslog-ng PE sends to the central server to ensure that sudden message bursts do not consume the bandwidth of other important applications, or to flatten the load of the server. Controlling the number of sent messages is useful also if you have a database or a log analyzing application on the server that can process only a limited number of messages. Using disk-based buffering together with the rate-limiting feature of syslog-ng PE prevents the loss of messages, and helps to use the resources effectively without overloading backend systems.

Works in both IPv4 and IPv6 environments

You can deploy syslog-ng in both types of networks, and use the same system logging tool across your entire network infrastructure.

Collect logs from Microsoft Windows

Using syslog-ng Agent for Windows, you can collect messages from logfiles and eventlog groups, and transfer all log messages to the central syslog server using encrypted, reliable TCP connections. That way you can integrate your Windows-based and UNIX-based devices into the same logging infrastructure.



Comparing syslogd and syslog-ng

This section gives you a summary of the differences between syslogd, the standard system logging solution used by most UNIX-like operating systems, and the open source and commercial editions of syslog-ng.

What does syslog-ng PE offer over syslogd?

The syslogd application is the standard system logging application used by network devices like switches and routers, as well as servers running operating systems based on Unix, including Linux, HP-UX, BSD, Solaris, and AIX, but excluding Microsoft Windows (Windows does not have a built-in remote logging solution). The implementations of syslogd on the different operating systems are in part system-specific, while syslog-ng has higher portability, using the same codebase on every platform. Regarding reliability, syslogd does nothing to ensure that the sent messages really arrive to the server. It uses the unreliable UDP network protocol, meaning that messages can get lost on the network without the sender or the server ever noticing it. Additionally, syslogd simply drops messages when the server is unavailable or overloaded. It does not have the ability to encrypt the messages, and the server can output the logs only into text files.

What does syslog-ng PE offer over syslog-ng OSE?

The syslog-ng Open Source Edition (syslog-ng OSE) application is the most popular and widespread alternative system logging application used in the world, having replaced syslogd on tens of thousands of systems. It has several features surpassing syslogd, including reliable message transferring using the TCP protocol and controlling the flow of messages to handle minor server outages. But only syslog-ng PE has the more advanced features of buffering the messages on the hard disk, natively supporting message encryption (SSL/TLS), logging directly into a database, and support for Microsoft Windows operating systems.

The following table summarizes the main differences between the syslogd, syslog-ng Open Source Edition (OSE), and syslog-ng Premium Edition (PE).

	syslogd	syslog-ng OSE	syslog-ng PE
Reliable message transfer using TCP	✗	✓	✓
Content-based message filtering	✗	✓	✓
Use macros to dynamically create target files, directories, and database tables	✗	✓	✓
IPv6 support	OS dependent	✓	✓
Encrypted message transfer (TLS support)	✗	✗	✓
Disk-based buffering	✗	✗	✓
Direct output to database	✗	✗	✓
Message-rate control	✗	✗	✓
Windows support	✗	✗	✓

For more information, see <http://www.balabit.com/network-security/syslog-ng/central-syslog-server/>



All questions, comments or enquiries should be directed to info@balabit.com or by post to the following address: BalaBit IT Security
1115 Budapest, Bárfai str. 54 Phone: +36 1 371-0540 Fax: +36 1 208-0875 Web: <http://www.balabit.com/>

Copyright © 2007 BalaBit IT Security Ltd. All rights reserved.

For more information about the legal status of this document please read:
http://www.balabit.com/products/zorp/docs/legal_notice.bbq