



Collecting syslog messages into an SQL database with syslog-ng Premium Edition

Synopsis: The advantages of using syslog-ng Premium Edition to collect system log (syslog) messages in an SQL database, such as Oracle, MSSQL or MySQL

Status: Released

Version: 1.1

Date: 04/10/2008



Table of contents

Preface.....	3
Introduction.....	4
What is system logging.....	4
Why is it useful to store the logs in a database.....	4
How to log into a database using syslog-ng.....	5
Using syslog-ng to log into an SQL database.....	5
Advantages of using syslog-ng Premium Edition	5
Other solutions	6
Creating the initial database.....	6
Visualizing messages stored in a database.....	7
Other important features.....	9
Secure logging using SSL/TLS.....	9
Disk-based message buffering	9
Output data into various formats.....	9
Select important messages.....	9
Control the rate of messages.....	9
Flow-control	9
Support for IPv4 and IPv6 environments.....	10
Heterogeneous environments.....	10
Collect logs from Microsoft Windows.....	10
Collect logs from IBM System i.....	10



Preface

This paper discusses the advantages of using syslog-ng Premium Edition to collect system log (syslog) messages into an SQL database, such as Oracle or MySQL. 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* briefly describes what system logging is, and why it is useful to store log messages in a database.
- *How to log into a database using syslog-ng* explains how you can send your messages into the database, and what are the advantages of using syslog-ng Premium Edition.
- *Visualizing messages stored in a database* gives you a brief overview of the tools you can use to browse and manage the log messages.
- *Other important features* discusses further features of syslog-ng PE that can come handy for you when designing and implementing your system-logging architecture.



How to log into a database using syslog-ng

The following sections describe how to log into an SQL database using syslog-ng, what are the advantages of using syslog-ng Premium Edition, and what are the disadvantages of other solutions.

Using syslog-ng to log into an SQL database

The syslog-ng application natively supports logging into databases, you only have to set the destination database. Currently the following databases are supported:

- MySQL
- Microsoft SQL (MSSQL)
- Oracle
- PostgreSQL
- SQLite

Native database support means that provided the database is already installed, you just simply configure syslog-ng to send the log messages into this database – no additional scripting or tweaking is required. (For detailed instructions and configuration examples, see the [documentation of syslog-ng](#))

The syslog-ng application offers superior performance compared to other solutions for the following reasons:

- The syslog-ng application generates the INSERT operations feeding the messages to the database natively in a C program, which is much faster than the custom scripts used in other solutions. Scripting languages (e.g., Python, Perl) are interpreted languages and inherently slower than optimized solutions created using compiled languages, such as the C language.
- The syslog-ng application uses the native interface (API) of the supported databases, which allows for much faster communication with the database than any other tool that loads data into the database from a file – that way syslog-ng can send several thousands of log messages per second into the database.
- To increase the processing speed further, syslog-ng performs all processing and operations in the memory of the system, without writing any messages to the hard disk. Messages are written to the hard disk only to avoid losing messages if the destination database becomes unavailable, or cannot process messages fast enough during message bursts.

Advantages of using syslog-ng Premium Edition

The syslog-ng PE application ensures that you do not lose messages during network or system outages, because syslog-ng PE can store unsent messages on the local hard disk until the database server becomes available again. The syslog-ng PE application can also apply flow-control on the messages. Flow-control means that if the destination server or database becomes overloaded, syslog-ng PE can stop accepting messages from the sending hosts or applications. That way the senders are notified that there is a problem in the logging infrastructure and can act accordingly: for example, in an environment where policy compliance mandates all events to be logged, the applications may temporarily halt until the logging can be resumed, so there are no actions that are not logged.

You can also 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 to the database is useful if the number of messages that the database can process is limited, or if you have 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.

The syslog-ng PE application is ideal for heterogeneous environments, allowing you to send logs into the database from several different operating systems, including Linux, Sun Solaris, AIX, HP-UX, Microsoft Windows, and IBM System i.

Other solutions

Other solutions that use syslog applications (e.g., syslogd) that do not have direct database support use the following method to write log messages into the database.

- The syslog application creates INSERT commands from the log messages using templates or scripts, and writes them into a file.
- A script executes the INSERT commands using the client application of the database, which sends the messages to the database server.

Alternatively, the following method can be also used:

- The syslog application creates comma-separated-value (CSV) files from the log messages using templates or scripts.
- The CSV file is fed into the database using the loader application of the database.

Both methods have the serious drawback that they require intensive scripting, and – depending on the amount of log traffic – may have serious performance limitations. Every message must be written to the hard disk first which can severely degrade performance, depending on the skills of the programmer who writes the scripts. But even a perfect script cannot be as effective as syslog-ng, which uses the native API of the database, without having to use hard disk operations.

Also, there is no way for the script to give feedback to the syslog application, e.g., it cannot request the syslog application to 'slow down' if it cannot process the messages – in contrast to the flow-control capabilities of syslog-ng.

Creating the initial database

When implementing a central syslog solution that collects the log messages into a database, it is important to create the respective database tables that store the data.

The syslog-ng application simplifies this problem by automatically creating and modifying the tables and columns if they do not exist. Indexes to the selected columns are also created automatically, so searching the database for important data is fast. The names of the tables can contain macros (e.g., logs_\${YEAR}\${MONTH}\${DAY}) that can include the host name or the IP address of the machine sending the messages. That way you do not have to manually modify the database if you add a new machine to your syslog infrastructure – syslog-ng handles this all for you automatically.

The values of the columns entered into the database can contain any macro supported by syslog-ng (e.g., the hostname, name of the application sending the message, date, etc.), or even data extracted from the text part of the message using regular expressions.

Other solutions usually use static database structure that must be created and maintained manually, although some features of syslog-ng can be reproduced with complex scripting.



Visualizing messages stored in a database

Once the messages are in the database, there are a number of tools that allow you to browse or search the data. Every database provides command-line tools to access the database (e.g., sqlplus for Oracle), but not everyone is comfortable with the command line. The following list provides a brief overview of the most common tools, including both general database-access tools and applications specialized for viewing syslog messages. Note that the list is not exhaustive, as some databases have a large number of management utilities available from different vendors.

MySQL

- [MySQL Administrator](#): a graphical utility for managing and developing MySQL databases (available Microsoft Windows, Linux, and Mac OS X)
- [MySQL Query Browser](#): a graphical utility for searching MySQL databases (available Microsoft Windows, Linux, and Mac OS X)
- [Toad for MySQL](#): a graphical utility for managing and developing MySQL databases (available only for Microsoft Windows)
- [phpMyAdmin](#): a web-based application for managing and accessing MySQL databases

Specialized tools

- [php-syslog-ng](#): a web application for viewing and searching syslog-ng logs
- [php syslog viewer](#): a web application for viewing and searching system logs
- [phpLogCon](#): a web application for viewing and searching system and event logs

Oracle Database

- [Oracle SQL Developer](#): a graphical utility written in Java for managing and developing Oracle databases (available for Microsoft Windows, Linux, Mac OS X, and other platforms)
- [PL/SQL Developer](#): a graphical utility for managing and developing Oracle databases (available only for Microsoft Windows)
- [Toad for Oracle](#): a graphical utility for managing and developing Oracle databases (available only for Microsoft Windows)

PostgreSQL

- [pgAdmin III](#): a graphical utility for managing and developing PostgreSQL databases (available for Microsoft Windows, Linux, Mac OS X, and other platforms)
- [phpPgAdmin](#): a web-based application for managing and accessing PostgreSQL databases

SQLite

- [SQLiteManager](#): a web-based application for managing and accessing SQLite databases
- [SQLite Explorer](#): a graphical utility for accessing and managing SQLite databases (available only for Microsoft Windows)

The main website of SQLite maintains a long list of management tools at <http://www.sqlite.org/cvstrac/wiki?p=ManagementTools>.



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.

Secure logging using SSL/TLS

Log messages may contain sensitive information that should not be accessed by third parties. Therefore, syslog-ng Premium Edition 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.

Disk-based message buffering

The Premium Edition of syslog-ng stores messages on the local hard disk if the central log server or the network connection becomes unavailable. The syslog-ng application automatically sends the stored messages to the server when the connection is reestablished, in the same order the messages were received. The disk buffer is persistent – no messages are lost even if syslog-ng is restarted.

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.

Flow-control

Flow-control uses a control window to determine if there is free space in the output buffer of syslog-ng for new messages. If the output buffer is full, then the destination cannot accept new messages for some reason: for example, it is overloaded, or the network connection became unavailable. In such cases, syslog-ng stops reading messages from the sending applications or hosts until some messages have been successfully sent to the destination.



Support for 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.

Heterogeneous environments

The syslog-ng application is the ideal choice to collect logs in massively heterogeneous environments using several different operating systems and hardware platforms, including Linux, Windows, Unix, BSD, Sun Solaris, HP-UX, and AIX. For a complete list of platforms supported by syslog-ng Premium Edition, visit <http://www.balabit.com/network-security/syslog-ng/central-syslog-server/>.

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.

Collect logs from IBM System i

Using syslog-ng Agent for IBM System i, you can collect application and system messages, as well as messages from the System i security audit journal. The collected messages are forwarded to the central syslog server using encrypted, reliable TCP connections. That way you can integrate your IBM System i devices into the same logging infrastructure as your UNIX and other devices.



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

Copyright © 2008 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