

The Syslog Standard: Facilities, Severities, and Which Variant to Run

Sun, Aug 16, 10:00 AM EDT · <https://scottslab.io/posts/the-syslog-standard>



TL;DR

Syslog is the lingua franca of Unix logging, and each message has four parts: a header (timestamp, host, process ID), a facility code (0–23, where it came from), a severity (0 emergency → 7 debug, lower is worse), and the free-form message. A common alert rule fires on severity 2 or lower. The original syslog is deprecated; syslog-ng and rsyslog are what actually run on Linux, journalctl reads systemd's binary journal, and NXLog centralizes across platforms including Windows.

If logs are the raw material, syslog is the format most of them arrive in, and knowing its anatomy makes filtering and alerting far less mysterious.

A syslog message has four components. The header carries the basics: a timestamp, the sending host (a hostname, or an IP when no name is available), and the originating process ID. The facility is a code from 0 to 23 that says where the message came from (kernel, mail, auth, and so on), which is how you route or filter by subsystem. The severity is the one worth memorizing, because it's backwards from intuition: it runs 0 to 7 where 0 is emergency and 7 is debug, so a *lower* number is *more* severe. And the message is the free-form content the process wanted to record. A sane default alert rule is to page on severity 2 (critical) or lower. Here, "lower" means "worse."



The implementation has drifted over the years, which trips people up. The original syslog daemon is largely deprecated now. Syslog-NG came along in 1998 and added the things the original lacked: encryption and reliable delivery. Rsyslog arrived in 2004 with further enhancements, and both syslog-ng and rsyslog are what you'll actually find running on Linux today. Separately, systemd brought journalctl, which reads a binary journal format instead of syslog's plain text. It's more structured and queryable, but not a text file you can grep, which surprises people the first time. And when you need to pull it all together across operating systems, NXLog centralizes both syslog and Windows sources.

The reason any of this matters for security is that filtering and forwarding rules live on these fields. You alert on severity, you route by facility, and you ship it all off the host. A log that only exists on the box that generated it is a log an attacker can delete. Which is exactly what a SIEM is for.