



Introduction

This booklet describes how to build an infrastructure to collect, preserve, and extract useful information from your computer operating system and application logs. We will focus primarily on UNIX *syslog*, with some discussion of Windows logging and other sources of log data. Logfiles hold a wealth of information, from resource utilization diagnostics to problems with hardware and software, security problems, and forensic traces of intrusions. Our examples are heavily weighted toward security issues, but we provide some examples of resource and diagnostic monitoring.

Unfortunately, there's an awful lot of information in log files, and it's not well organized or codified. Formats of messages, even timestamps, vary between applications, and sometimes even between different versions of the same application; different operating system distributions will use different messages to record the same event; and the information you need may be spread out over several messages.

Many system administrators have been told to “go figure out those logs.” It's a daunting task—there's an awful lot of data, little of which seems to be useful or pertinent, at least at first glance. If you did persevere, you probably built a monitoring system based on the relatively random data that showed up early on, without recognizing that the project would get a lot easier if you thought about what you'd really like to know before you started putting the pieces in place. We're going to change all that.

This book assumes that you have little or no knowledge of logging. The experienced system administrator may find some—but far from all—of the information presented pedantic. But we'd rather be pedantic than leave some relative beginner out in the cold just because we assumed a level of familiarity with this stuff that a beginner doesn't have.

The goal of this book is not to teach you how to interpret log files from any particular system (how would we pick?), how to write Perl scripts, or how to rewrite *syslog*. It's to provide an overview of the sorts of information your logfiles can give you: how an archetypal UNIX log system (*syslog*) works, how to consolidate your UNIX and Windows XP/2000 logging, and how to monitor your network for intrusion detection, forensic analysis, and chaos reduction.

The strategy we present here consists of three basic parts:

- Generating good log information, because it's no good looking at your logs if they don't contain much useful data

2 / Introduction

Collecting and archiving log data in a central location, because it's easier to analyze information when it's all in one place

Performing analyses that extract useful information from the logs

That last point is our ultimate goal: teaching you how to find useful information in your logs.

In the first chapter, “Why Logs Are Important,” we’re going to talk about what’s good and bad about logs and give a few examples of what you can learn from them. The second chapter, “Getting Started,” will show you the basics of setting up logging on a single host, describe what logs look like, and suggest some things to start looking for in your logs. Then, in “Sources of Log Information,” we’ll get you thinking about what kind of information you want in your logs and how to collect it, along with some examples of configuring services for logging to *syslog*. Once you have your systems generating log information, “Centralized Logging” will show you some ways to have all your systems forward their *syslog* information to a central server and will discuss how to rotate and archive your logfiles. “The Gory Details” delves into the *syslog* protocol, *syslog* server configuration, and alternative *syslog* implementations. In “Log Reduction, Parsing, and Analysis” we’ll talk about tools and techniques for extracting useful information from your logs, including sample data and signatures of known attacks. The last chapter, “Windows Logging,” will describe the Windows analog of *syslog*, the Event Log, and how to integrate Event Log data into your *syslog*-based logging infrastructure. Finally, the Conclusion will leave you with a few words of advice and encouragement.