Chapter 53
Intelligent Log Analysis Using Machine and Deep Learning

Steven Yen
San Jose State University, USA

Melody Moh
https://orcid.org/0000-0002-8313-6645
San Jose State University, USA

ABSTRACT

Computers generate a large volume of logs recording various events of interest. These logs are a rich source of information and can be analyzed to extract various insights about the system. However, due to its overwhelmingly large volume, logs are often mismanaged and not utilized effectively. The goal of this chapter is to help researchers and industrial professionals make more informed decisions about their logging solutions. It first lays the foundation by describing log sources and format. Then it describes all the components involved in logging. The remainder of the chapter provides a survey of different log analysis techniques and their applications, consisting of conventional techniques using rules and event correlators that can detect known issues, plus more advanced techniques such as statistical, machine learning, and deep learning techniques that can also detect unknown issues. The chapter concludes describing the underlying concepts of the techniques, their application to log analysis, and their comparative effectiveness.

INTRODUCTION

Long before the advent of computers, logging has been used in various fields. Examples included physical logbooks, accounting transaction ledgers, car maintenance records, etc. They are used to record any events of interest based on the context. The information in the logs can then be used in the future for troubleshooting purposes, help improve operating procedures, act as an audit trail, and so on.

DOI: 10.4018/978-1-7998-7705-9.ch053

The practice of logging was adopted in computing systems from the very beginning. Developers used printf statements throughout their code to print relevant information to help them debug the code when issues arise. Some of the messages are only used during development and are removed before release, others were placed strategically to help with troubleshooting or monitoring purposes later on. These log messages can be shown directly to the user or be sent to specific outputs channels such as to a file. Due to its usefulness, logging became common practice, and nowadays almost every piece of software has logging capability. In modern computing systems, logs can come from operating systems, network devices, and various application software. They are meant to record interesting events that occurred when programs are ran.

These logs from various devices and processes proved to be extremely useful for the detection of security issues. Operating system logs (or *host logs*) can be analyzed to detect unauthorized access, such as that by an attacker using a *ssh-scanner* (Chuvakin, Schmidt, & Philips, 2013). Network logs can be analyzed to detect unusual traffic such as that between a malware and a remote attacker's device (Stamp, 2006). Web application logs can be analyzed to detect attacks such as cross-site scripting, SQL injection, and invalid resource access (Liang, Zhao, & Ye, 2017). Many, if not all, cyberattacks leave traces in logs somewhere, one just needs to know what to look for.

However, because of the automated nature of log generation in computing systems, the volume of logs generated became very large. An unfortunate consequence of this is that many users began to view logs as an annoyance rather than a helpful tool. Logs were seldom looked at and are often simply deleted when space runs out. To address these issues, log management systems were developed to facilitate the collection, storage, and analysis of logs.

Log analysis can be done manually by inspecting raw text files directly or using event viewers provided by log management systems. Such manual inspection is labor-intensive and often not timely enough for real-time incident response. To address these limitation, rule-based systems were developed that can evaluate log events based on a library of known issues (known as a rule-base). These tools proved to be quite effective and have helped organizations prevent many incidents in a timely fashion. The drawback is that they can only detect known issues for which there are exact rules in the rule-base, and misses unknown issues. To help detect new and unknown issues, anomaly detection approaches were introduced, which are based on identifying unusual or abnormal behavior. Statistical, machine learning, and deep learning techniques proved to be quite suitable for this application, because they can form their own detection criteria from training data rather than relying on human operators to specify rules. Over the years, more and more of these techniques have been applied to log analysis with impressive results.

Cognitive science played an important role in the development of intelligent log analysis tools. This is because the individuals using the computer systems, the analysts, and the attackers are all human. Understanding the thought process and motives of all these individuals is therefore crucial in trying to identify issues and malicious activity. As testament to this, there has been studies done to understand how human subjects analyze logs to detect issues (Layman, Diffo, & Zazworka, 2014) as well as studies to understand user web browsing behavior through logs (Kussul & Skakun, 2005). In fact, intelligent log analysis tools have always been designed to emulate human, with early rule-based systems aiming to capture the decision making process of security experts who wrote the rules. However, the attack methods are constantly evolving as the creators of these attacks are humans too, whose goals are to come up with new ways to avoid detection. It takes time for security experts to study and dissect a new attack,