Realtime publishers

Log Management:
Best Practices for Security and Compliance
The Essentials Series

Best Practices for Log File Management (Compliance, Security, Troubleshooting)

sponsored by

IPSWITCH WhatsUpGold
Network Management Software

Eric Schmidt

Introduction to Realtime Publishers

by Don Jones, Series Editor

For several years now, Realtime has produced dozens and dozens of high-quality books that just happen to be delivered in electronic format—at no cost to you, the reader. We've made this unique publishing model work through the generous support and cooperation of our sponsors, who agree to bear each book's production expenses for the benefit of our readers.

Although we've always offered our publications to you for free, don't think for a moment that quality is anything less than our top priority. My job is to make sure that our books are as good as—and in most cases better than—any printed book that would cost you $40 or more. Our electronic publishing model offers several advantages over printed books: You receive chapters literally as fast as our authors produce them (hence the "realtime" aspect of our model), and we can update chapters to reflect the latest changes in technology.

I want to point out that our books are by no means paid advertisements or white papers. We're an independent publishing company, and an important aspect of my job is to make sure that our authors are free to voice their expertise and opinions without reservation or restriction. We maintain complete editorial control of our publications, and I'm proud that we've produced so many quality books over the past years.

I want to extend an invitation to visit us at http://nexus.realtimepublishers.com, especially if you've received this publication from a friend or colleague. We have a wide variety of additional books on a range of topics, and you're sure to find something that's of interest to you—and it won't cost you a thing. We hope you'll continue to come to Realtime for your educational needs far into the future.

Until then, enjoy.

Don Jones

Table of Contents

Copyright Statement

Best Practices for Log File Management (Compliance, Security, Troubleshooting)

Introduction

The final article in this Essentials Series will move beyond the details of specific logs and scenarios in which they can be used to a discussion on best practices for implementing and leveraging centralized log management.

Architecting the Infrastructure

In order to develop an effective centralized log management strategy; the first task is the development of requirements for what will be collected, from what systems, and how long will logs be retained. To determine what systems logs will be collected from, the simplest thing to do is break systems into tiers based on the service they provide. For example, systems that hold critical business, financial, and authentication data would be required to have logs centralized whereas systems that perform less critical business tasks could be excluded. One of the best methods to use when breaking systems into tiers is to examine regulatory requirements as these may identify systems that are audited. Another valuable source to determine whether or not logs should be centralized is the company's disaster recovery(DR) plan. If a DR plan has already been developed it's very likely that all of the systems that comprise the infrastructure have already had their criticality assessed so in the event of a disaster, the most critical systems are restored first. Those that rank the highest on the DR plan should have their logs centralized. Figure 1 provides an example of a three-tier model with Tier 1 being the most important systems to centralize logging for.

Tier	Role
1	Network Infrastructure
1	Financial Systems
1	Personally Identifiable Information (PII)
1	Identity/Authentication Systems
1	DMZ systems (Internet Accessible)
2	Management Systems (Patch, Configuration, etc)
2	Non-Business Critical
3	Workstations
3	Development/Test Systems

Figure 1: Three Tier Role Categorization

Extending Centralization beyond Servers

Of course log centralization should not be limited to just systems. Both the network and security infrastructures should be required to have their logs centralized. These devices or appliances will not only contain valuable information regarding the overall health of the infrastructure, their logs will be some of the first to be examined during a security incident. Having the logs centralized will enable the incident response team to track the incident throughout the Enterprise.

One thing that's worth pointing out here is that the first thing being evaluated when gathering requirements is what systems will have their logs centralized. It was intentional to mention this before any other aspect because this is the required first step in determining what the ultimate centralization solution will look like and will drive all subsequent requirements. Under ideal circumstances the log centralization infrastructure will be architected to support what has been identified to be collected as opposed to limiting what can be collected based on predetermined tools or storage limitations.

Log File Retention

The next area for requirements gathering is a determination as to how long logs will be retained. In conjunction with that, one must also identify the type of access to the centralized logs is required throughout the retention period. For example, there may be a requirement to retain logs for seven years, but immediate access to log data may only be required for one year. In these situations, the centralization architecture can be designed to include an archival process where the most recent year's worth of data is readily available and data from years two through seven are archived and thus requiring slightly more effort to access. The benefit of implementing an archiving strategy is that the retention requirements are met, but the cost of meeting the requirement can be reduced. An example of an centralization strategy that leverages archiving would be one where the last year's worth of data is stored in a database that offers rapid access to the data and everything beyond the one year period is stored as flat files that can be compressed or stored using less expensive hardware.

Another way to gain the most efficiency with log storage is to determine retention periods per system based on the role that the system provides. Regulatory requirements may dictate extended retention periods for certain systems, while with other systems there may be no value in retaining the logs beyond a much shorter period of time. In the previous article, the value of centralizing workstation logs was mentioned with the potential benefit of detecting security incidents, however, this may be the first category of devices that are eliminated from log centralization due to the increased storage requirements that it would impose. This represents a prime example of establishing retention periods based on role because the usefulness of centralized workstation logs may only be something like ninety days or less. Given the benefit that centralizing workstation logs provide, it would be advantageous to have the ability to adjust the retention period as a means to control cost versus the total elimination of the collection itself.

Estimate Storage Requirements

The last requirement to be collected is an estimate of the log sizes themselves. This will likely be the most challenging part of the requirements gathering process because the size of the logs on a particular system are directly related to the services that it provides or the applications that are installed. To help with this determination, vendors will often have estimates of log sizes based on other conditions like how many users access the system. Beyond that, it's really just performing the leg work to gather data on existing log sizes and where possible, projecting growth. While this may be the most challenging task, it is also the most important because implementing an improperly sized centralized log infrastructure can render the collection useless, especially if there isn't sufficient storage space to collect and retain what has been identified.

One way to make log centralization as efficient as possible is to limit what is forwarded to specific events. Microsoft and many third party web sites provide detailed lists of events and explanations of what generates them. These resources can be utilized to identify the specific event IDs that need to be collected. A good example of this for Active Directory domain controllers would be to only centralize events that pertain to authentication attempts (both successful and unsuccessful), changes to group memberships, creation/deletion of user accounts. Taking this approach can dramatically reduce the number of events that are forwarded to the collector, thereby reducing network and storage requirements. There is however, a word of caution with this approach. A significant amount of analysis must be performed to ensure that a sufficient number of events are collected that will satisfy security and audit requirements. If great care is not taken it's likely that some will be missed and as such the collection of forwarded events will not be able to create a complete picture of system activity. If resources permit it is much more advantageous to forward all events and rely on the centralization tool to perform the required filtering. This will avoid a circumstance where a required event wasn't being forwarded, and therefore being overlooked during an analysis.

Optimizing bandwidth

Once the requirements have been collected the next step is to determine the mechanism that will perform the centralization and the impact that it will have on WAN bandwidth. Much of the design will be dictated by how dispersed the company is. Companies that have locations spread throughout the country or the globe will likely want to implement a tiered approach to centralization. A tiered architecture would have regional collectors which then consolidate everything to a central location. This approach provides the benefit of collection logs closer to the actual clients, thereby reducing WAN traffic while at the same time centralizing the collection of all logs. The number of tiers will be largely dependent on the network topology and the number of clients at each site. The figures below depict a flat architecture and a tiered model with regional collectors that forward to a central collector. The main take away from this is that the log centralization architecture can be designed to minimize WAN traffic while at the same time centralizing all the logs.

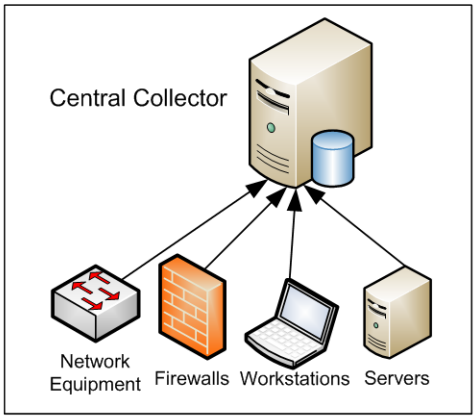

Figure 2: Single Tier Architecture

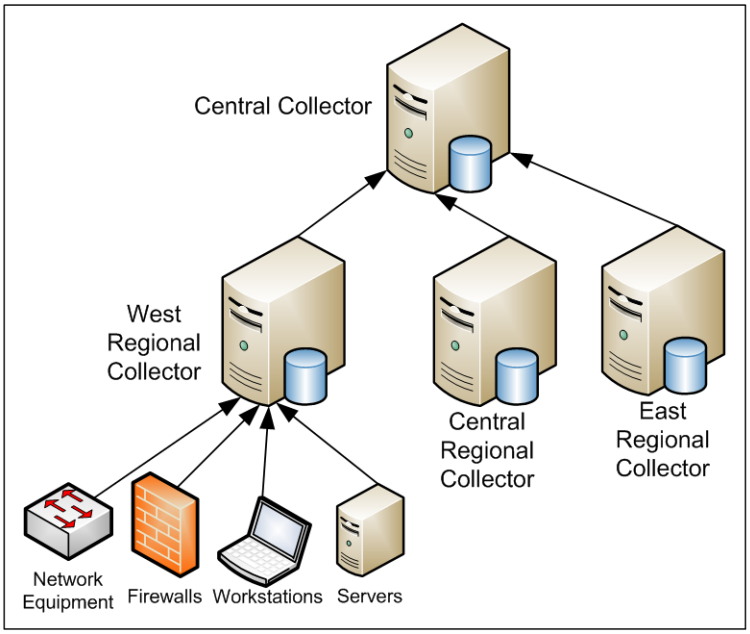

Figure 3: Two Tier Architecture

Leverage the Logs

Now that the architecture is in place and systems are forwarding all their events the next step is to leverage the information that is collected. There are multiple ways that this can be accomplished. The first is to establish criteria for automated alerts when specific events occur. The previous articles in this series provided examples of real time alerts, but those were limited in scope. Additional alerts to consider include multiple failed login attempts or accounts being locked multiple times over a short period of time. This type of activity may be something benign like someone forgetting to update a scheduled task or service account, but can also be an indication of malicious activity. Multiple failed login attempts also provides a good example of tailoring an alert based on a threshold of occurrences over a period of time because it's likely that a single failed login attempt does not warrant immediate notification, but if that same account has multiple failed login attempts over a short period of time then an alert is warranted. Having the ability to set thresholds on the number of events before an alert will prevent false positives and serve to create the overall effectiveness of the alerting process. Provisioning of accounts or accounts being added to highly sensitive groups is another thing that real time alerts can be used to make System Administrators aware of the activity. Alerting on infrastructure health should be configured for real time alerting. Events like systems that are low on available disk space or services that failed to start are good candidates. Ideally, each alert should be configured to notify the appropriate individuals and configured so that the alert they receive contains actionable information. Part of this may be the development of processes and procedures that are to be followed when the alerts are received. Real time alerts should also be setup for network infrastructure events like excessive firewall denies on a particular port or events that are captured by intrusion detection systems. As mentioned previously, these alerts, if implemented properly, will become the key to preventing widespread security incidents or compliance issues.

Tools that centralize log collection will often have reporting features that should also be leveraged. Reports that are capable of creating heat maps of events over a period of time can aid in the diagnosis of a problem in a particular region. An example of this would be a circumstance where a patch was deployed to a particular set of systems in a given region that breaks a critical service. A report that displays the failed start event for the broken service for that set of systems could aid in the correlation between the patch deployment the failed service start. Reporting could also be leveraged to collect metrics on uptime status for systems. For example, all Microsoft Windows systems write events to the log with respect to uptime and boot time. These collected events could be leveraged to generate uptime or last reboot reports for systems in the enterprise.

The Database

Some of the centralization products available also use a database to store all the forwarded events. This can be very advantageous because it allows System Administrators the flexibility to extend the usefulness of the collected data beyond the tools and interface the vendor has provided. There may be circumstances where the data needs to be analyzed in ways that aren't provided by the vendor. With some knowledge of the underlying database, queries can be written to very specific needs. Being able to query a database can also play a key role in optimizing the retention and archival process. It may be the case where only certain events need to be retained for an extended period of time and with the ability to interact directly with the database would allow the extraction and storage of those specific events on another external system. Not only will this reduce the storage requirements of the collector but it would also eliminate all the unnecessary events from the archival process. Direct database access would also allow custom dissemination of the data either via a web service like SQL Reporting Services or other custom means.

Conclusion

In this Essential Series we've examined the importance of the data stored in log files and the benefits of centralizing collection so that they can be leveraged for incident response, compliance management, and troubleshooting. Real-world examples were used to demonstrate the benefits that centralization provides. Finally, a strategy was outlined and best practices were identified for collecting, alerting, and reporting on the centralized events. All of which result in an Enterprise solution that enables System Administrators, Information Security and compliance personnel to operate more efficiently and effectively because data that was once highly distributed is now centralized and readily accessible.