



Global Information Assurance Certification Paper

Copyright SANS Institute
Author Retains Full Rights

This paper is taken from the GIAC directory of certified professionals. Reposting is not permitted without express written permission.

Interested in learning more?

Check out the list of upcoming events offering
"Hacker Tools, Techniques, and Incident Handling (Security 504)"
at <http://www.giac.org/registration/gcih>

Creating a Logging Infrastructure

GIAC (GCIH) Gold Certification

Author: Brian Todd, a.brian.todd@gmail.com

Advisor: Johannes Ullrich

Accepted: March 31, 2017

Abstract

Logs are an essential aspect of understanding what is occurring in a company's network infrastructure and a company's applications. Log events help analysts to understand the health of the network and give insight into many types of issues. This paper explains how to set up a logging infrastructure by covering log formats and data sources. Then the discussion includes different ways to collect logs and transmit them. This paper then goes over how to pick relevant log sources and events to enable for collection. A company-wide architecture describes the process of collecting logs from offices across the world. Once the company-wide architecture is set up, the paper goes over some correlations using data from a real production network. The paper finishes by reviewing tools that are used to process, index, and correlate all the events that are received.

1. Introduction

In a world without logs, a company would have a limited idea what is happening in the network. It would be as if the engineers were all walking around with blinders covering their eyes and earplugs in their ears. The engineers would be bumping into objects and not aware of what is going on around them. Without logs, security professionals would have a more difficult time to see nefarious actors causing problems in an organization's network.

Investigating an incident is one significant benefit of having a developed logging infrastructure. This next example uses a network architecture where the Firewall performs NATing, and there is no proxy. A company's security team receives an email communication from their ISP that informs the security team that a copyright holder has sent the ISP a DMCA violation which includes an IP address assigned to the security team's company. The email from the ISP contains the timestamp, the source IP address, and source port along with some other information. From this information, a search would be conducted through the firewall logs looking for the given outbound source IP address and outbound source port that occurred around the timestamp. Once an analyst has found the entry that matches the criteria, the analyst will look at the other fields in the firewall event for the internal IP address.

Once the investigator has obtained the internal IP address, the DHCP logs or static IP address assignments will be searched to find out which host was assigned that IP address during the time in question. In most cases, a single host corresponds to an individual, so the analyst would be able to figure out who owns that host using an asset tracking database or by checking who is logged into that host and talking to them as necessary. The above scenario shows one of many situations where an efficient logging infrastructure can support incident response management.

Most of a company's security solutions rely on attaining information about the status, health, and actions of the network infrastructure and applications. Information security engineers have tools, such as a SIEM, which can tie events together and correlate events from across the company's network. A SIEM is invaluable due to the large size of

the networks. The large size of the networks refers to not only the quantities of systems and applications but also to the geographic locations across the globe.

A SIEM can process the logs and can provide insight into what is happening within the organization's network infrastructure and applications. This insight includes authentication issues, privilege escalation, and vulnerability information which is very useful for network management, intrusion detection, forensics investigations, and for meeting legal requirements.

For the reasons mentioned above, it is important to configure and maintain a proper logging infrastructure. The following sections of this research will discuss the various ways to set up and configure an effective logging infrastructure. This paper details a few different architectures, including architectures for single offices and multiple offices around the world. This article also describes ways to collect the logs, process the logs, and store the logs.

The paper concludes by looking at correlations. Correlations are useful in logging as they allow a security team to create relationships between different logging events. These correlations can help reduce alert fatigue and also allow for more refined alerts. The paper looks at some correlations and uses real production data to check the correlations effectiveness.

2. General Logging

Logging refers to the process of collecting information from various sources. During this collection process, the logging infrastructure is storing this information, which is composed of data known as 'events,' into a particular format. The events are the discrete pieces of information that tell analysts what is happening with the network, on a host, or with specific applications. "An event is a single occurrence within an environment, usually involving an attempted state change. An event typically includes a notion of time, the occurrence, and any details that explicitly pertain to the event or environment that may help explain or understand the event's causes or effects." (The CEE Editorial Board, 2010). Events need to have a minimum set of information so that they are considered useful. For example, an event that states 'Transaction failed' does

Author Name, email@addressa.brian.todd@gmail.com

not give an investigating analyst enough useful information. Programmers can improve the value of an event by adding additional fields, such as the date, time, time zone, and more information about the event, for example, ‘2017-02-06 14:06:23 UTC Transaction failed – credit card rejected’. These new fields convey information which allows for a more informed action.

There are some standard fields that are important for events to have. The field information can vary depending on the event, but for the most part, they should have a date, time, time zone, event source hostname, event severity level, and then the event description. In contrast to fields that are important to have, some fields are important not to store in the logs. Examples of these types of fields would include personally identifiable information and passwords.

It is important to use NTP and have a reliable time server that will keep the time synchronized across the logging infrastructure. Using a common time zone, such as UTC, for the logging infrastructure will avoid the need to convert timestamps from different geographic locations regularly. Log sources configured with local time become normalized to UTC by the logging infrastructure.

3. Log Formats and Event Fields

3.1. Log Formats

There are two main logging formats, formatted text logs, and binary formatted logs. For formatted text logs it may be possible for a human to read and interpret them. However, in some cases, text-based logs can be comprised of Unicode or a long format like XML which makes it more difficult for a human to read. Below in Figure 1 is a screenshot of what a text format event entry looks like when opened with WordPad.

```
2017-02-26T12:29:09-08:00 fw,fwmon src=10.10.10.100 dst=10.10.20.26  
ipprot=6 sport=55216 dport=23 Unknown inbound session stopped  
2017-02-26T12:29:43-08:00 fw,fwmon src=10.10.10.101 dst=10.10.20.26  
ipprot=6 sport=59975 dport=4000 Unknown inbound session stopped
```

Figure 1: Sample firewall log

An advantage of text-based logs is that it is usually straightforward for a human to read them and get an idea of what is happening. Text formatted events also lend themselves to creating personalized parsing tools for additional processing. Some of the downsides of formatted text events are that they take more processing power to process than binary formatted logs. They are larger which affects both the storage and bandwidth necessary to handle the formatted text events, although they can be compressed with good results.

Binary formatted logs require some method to view the data. A typical example of a binary log is the Windows Event logs. Below in Figure 2 is a screenshot of what the binary log formatted event entry looks like when opened with WordPad.

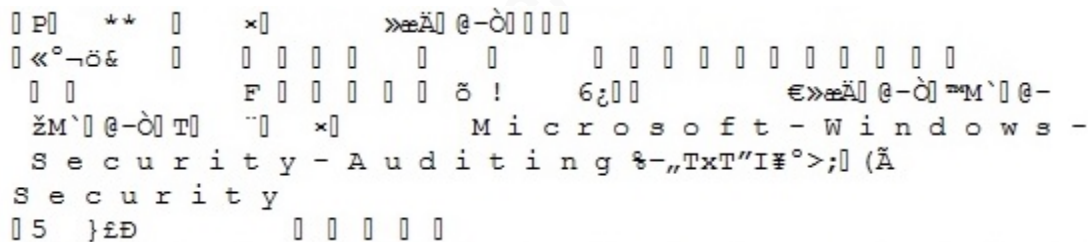


Figure 2: Sample Windows Event log

One of the main disadvantages of binary formatted logs is that they require some post-processing to view their content. However, they are usually more compact than formatted text logs. Thus this gives the advantage that the transmission and storage of logs to use less bandwidth and space. Binary formatted logs also tend to require less processing power, which is critical in some implementations as a company should avoid its logging functionality to affect the performance of the system.

3.2. Event Fields

Events require essential fields in order to provide useful information. These fields can vary depending on the event, but for the most part, they should have the following fields at a minimum.

<date> <time> <time zone> <event source identifier> <event severity level>
<event information>

The above fields will provide a good basis for understanding what is occurring in a network. As noted above, there is a variation in the event fields depending on the source of the event. For example, in the <event information> field, an event from a router will most likely contain the source IP address, source port, destination IP address, destination port, and some informational text about the event, whereas an application might have the application name, user, and description of the event.

4. Log Sources

Nearly anything that is running within a network can be considered a log source. Most systems have health events which report CPU utilization, memory utilization, network bandwidth, and temperatures of the system and its components. On top of these health events, the systems can add operating system events and application events. Given that there are so many different devices running on a network which are potential log sources, it is helpful to decide what to log efficiently.

One of the more critical log sources to collect from is any device that is performing authentication or authorization. Authentication logs are important for detecting attacks throughout a company's network as well as for knowing which users were logged in to systems during any time period. Authorization logs can let a security engineer know if users are trying to access information that they are not authorized to view.

Another group of important log sources to collect and process are from the Firewalls, IDS, and IPS. Companies place these systems on the border of a company's internal network and the public Internet. Thus, these systems log events are important to review to understand the security of a network and to perform investigations. The IPS and IDS systems will log intrusion events that were detected or prevented. The Firewalls will log all the inbound and outbound connection information as well as a wealth of information, such as connections per second, concurrent connections, bandwidth used, and the duration of connections.

The next group of critical log sources are the endpoint security log sources. These solutions include anti-malware, application control, file integrity monitoring, and forensic

information. These logs provide information about detected malware, installed programs/patches, and historical forensics information.

Companies also want to log their network infrastructure hardware and services. For the network infrastructure hardware logs this includes routers and switches. These logs are used to monitor and troubleshoot network issues as well as to detect some network attacks. For the network infrastructure services, this includes DHCP and DNS. Both of these services allow for a smooth and easy operation of a network.

DHCP logs are used to correlate which client had a particular IP address at a certain time. DHCP logs can be useful for identifying if any new systems appear on a company's network. An analyst can also use DHCP logs to check for anomalous behavior, such as a sudden increase in the DHCP server assigning IP addresses, clients wanting to renew their IP address lease very frequently, or if all of the IP addresses are used up.

DNS logs are used to see what domains are being looked up, and a company can check for strange domain names that don't make sense. If the DNS logs show that someone made a lookup for `ufdhjj453fg.ff84hdfaskjf489.com` that would certainly stick out as unusual. This strange domain lookup request could be an indicator of compromise, so an analyst would check the logs and see which host made this request and investigate this further.

The last type of log sources to collect are application log sources. These sources include databases, on-premises applications, and cloud-based applications. The information from these logs shows who access them and what commands they run on the applications. Applications with sensitive or confidential information need to have a higher priority or risk rating than applications that don't process that type of information.

5. Collecting Logs

Many modern systems natively support some method of logging transmission. Among Linux based systems and network infrastructure (Firewalls, Routers, Switches) the most common is syslog. For systems that have this capability, it is relatively straightforward to configure the information for where to send the logs. The

Author Name, email@addressa.brian.todd@gmail.com

configuration parameters are usually a FQDN or IP address, a choice between UDP or TCP, and port (usually 514). There also may be alternative parameters such as which format to send the logs in, rate limiting, and truncation.

Companies use agents for systems that do not natively support the transmission of logs in a particular format needed by the log collectors or other tools. Agents are a program that runs on the system and will collect the logs and then transmit them to the log collection server or other processing systems in a format that is understood. Some examples of agents are Snare Enterprise (for Windows events), Snare Epilog (for flat text files), Syslog-NG, and Rsyslog. For example, Windows Server 2012 events, which can include domain controller events, are stored in the Windows event log format which is a binary file. There is no native support to send them off the system in the syslog format. So, in this case, an agent will be installed that will be able to read the Windows event log and send them to the log collectors and other log processing systems via syslog.

6. Transmitting Logs

When transmitting the logs, there are a few choices. The main option is to choose between UDP or TCP for the transport layer. UDP has less messaging overhead but does not guarantee delivery. TCP has more messaging overhead but does guarantee delivery.

The system or agent that is transmitting the logs should send the log in a format that can be parsed by the receiving system. The most common formats are Syslog defined in RFC3164 or RFC 5424. Other formats include CEF (HP's ArcSight SIEM) and LEEF(IBM's QRadar SIEM).

When transmitting logs from one physical location to the site of the log processing tools an organization needs to protect the information in transit. The protection of information involves encrypting the logs with TLS for the best protection. This will encrypt the logs from the source to the destination. Another alternative is to use a VPN over the public Internet. However, using a VPN to send the logs results in the logs being sent unencrypted through the company's internal network which is less secure.

7. Choosing Relevant Events for Logging

An organization wants to select log sources and events that are relevant and that provide information that is of practical and forensic use. A company does not want to be sending every log or event that a system produces. In configurations where there is too much logging data, it can place a burden on the infrastructure as well as anyone that is reviewing the logs. The excessive logging will use up more network bandwidth, take up unnecessary CPU cycles, and use up storage space.

Most events have a field that contains the severity or importance of the particular event. These categories are used to help remove logs that are not useful to be sent for processing or storage. Having between five and eight categories is common. The syslog entry on Wikipedia shows eight categories as shown in Figure 3 below. (Wikipedia.org 2017)

Severity level [\[edit \]](#)

The list of severities is also defined by RFC 5424 [\[6 \]](#):

Value	Severity	Keyword	Description	Examples
0	Emergency	emerg	System is unusable	This level should not be used by applications.
1	Alert	alert	Should be corrected immediately	Loss of the primary ISP connection. Ski Haus Delta has not reported status within status_timeout (120)
2	Critical	crit	Critical conditions	A failure in the system's primary application. Ski Haus Delta reports temperature < low_critical (30)
3	Error	err	Error conditions	An application has exceeded its file storage limit and attempts to write are failing. Ski Haus Delta reports temperature < low_error (32)
4	Warning	warn	May indicate that an error will occur if action is not taken.	A non-root file system has only 2GB remaining. Ski Haus Delta reports temperature < low_warning (36)
5	Notice	notice	Events that are unusual, but not error conditions.	Ski Haus Delta reports temperature < low_notice(50)
6	Informational	info	Normal operational messages that require no action.	An application has started, paused or ended successfully. Ski Haus Delta reports temperature 60
7	Debug	debug	Information useful to developers for debugging the application.	

Figure 3: Severity level of log events

The severity levels that are of the most interest are Emergency, Alert, Critical, and Error. The severity levels of Warning and Notice generate a greater volume of messages which may not be useful to an organization. There may be cases where there are sub-categories within the Warning and Notice events that will allow an analyst to refine further the events being sent for those particular severities so that there is limited

processing of events that are of little or no value. Achieving the right balance for Warning and Notice events will take some fine tuning and will be worthwhile.

For Informational and Debug logs an analyst can get those logs on the appliances themselves if needed or configure them to be sent for a short period to the logging infrastructure as required to investigate some issue. These severity categories create a lot of events, and when not configured correctly it is possible to overwhelm parts of the logging infrastructure.

8. Truncating Events

In some cases, events will have reoccurring or unneeded information in them. Although the main event information itself is needed and useful, the extraneous information will be wasting resources. Similar to logging unneeded events, the irrelevant information in events will use up more network bandwidth, take up unnecessary CPU cycles, and use up storage space. For cases like this, it is helpful to truncate event information so that only the necessary information is present.

Two examples of events that have reoccurring and unneeded information in them are from Microsoft Windows Active Directory. Active directory has event identification 4624 which is an event that covers a successful login. The successful log on event includes Interactive, Network, Batch, Service, and Remote Interactive types. The number of 4624 event identifications can vary quite widely on a company and department basis. The example below uses five hundred of these events per day per user as an average. The standard 4624 event identification shown below in Figure 4 and is approximately 2269 bytes for Windows Server 2012.

```
Feb 6 01:16:24 hostname01.domain.com host01L-PDC01 - - Feb 05 17:16:23 2017 4624 Microsoft-Windows-Security-Auditing
domain\host01-test715672-D$ N/A Success Audit hostname01.domain.com Logon An account was successfully logged
on. Subject: Security ID: S-1-0-0 Account Name: - Account Domain: - Logon ID: 0x0 Logon Type: 3 Impersonation
Level: Delegation New Logon: Security ID: S-1-5-21-3075943545-4156245207-31223452600-51221 Account Name: host01-test715672
-D$ Account Domain: domain Logon ID: 0x2C1DAF3A Logon GUID: {ECC52B9D-354E-1391-8724-49F8CF65B816} Process Information:
Process ID: 0x0 Process Name: - Network Information: Workstation Name: Source Network Address: 10.200.200.100 Source
Port: 62078 Detailed Authentication Information: Logon Process: Kerberos Authentication Package: Kerberos Transited
Services: - Package Name (NTLM only): - Key Length: 0 This event is generated when a logon session is created. It is
generated on the computer that was accessed. The subject fields indicate the account on the local system which requested the
logon. This is most commonly a service such as the Server service, or a local process such as Winlogon.exe or Services.exe. The
logon type field indicates the kind of logon that occurred. The most common types are 2 (interactive) and 3 (network). The New
Logon fields indicate the account for whom the new logon was created, i.e. the account that was logged on. The network fields
indicate where a remote logon request originated. Workstation name is not always available and may be left blank in some cases.
The impersonation level field indicates the extent to which a process in the logon session can impersonate. The authentication
information fields provide detailed information about this specific logon request. - Logon GUID is a unique identifier that can be
used to correlate this event with a KDC event. - Transited services indicate which intermediate services have participated in this
logon request. - Package name indicates which sub-protocol was used among the NTLM protocols. - Key length indicates the length
of the generated session key. This will be 0 if no session key was requested. 378
```

Figure 4: Windows Security Event 4624 full text

Starting with the text, ‘This event is generated when a logon’ is a description of the event that is in every event occurrence. Each event does not need to send this information, so the truncation result is in Figure 5 below. The truncation functionality and configuration is a part of the collecting agent's configuration parameters.

```
Feb 6 01:16:20 hostname01.domain.com MSWinEventLog - - - Feb 06 06:46:19 2017 4624 Microsoft-Windows-Security-Auditing
domain\testusername N/A Success Audit hostname01.domain.com Logon An account was successfully logged on.
Subject: Security ID: S-1-0-0 Account Name: - Account Domain: - Logon ID: 0x0 Logon Type: 3 Impersonation Level:
Impersonation New Logon: Security ID: S-1-5-21-3085385345-4159003607-313546834400-30359 Account Name: testusername Account
Domain: domain Logon ID: 0x2C793D31 Logon GUID: {B7645763F1-BCB3-E22F-444E-FA81635623263} Process Information: Process
ID: 0x0 Process Name: - Network Information: Workstation Name: Source Network Address: 10.200.200.200 Source Port:
61513 Detailed Authentication Information: Logon Process: Kerberos Authentication Package: Kerberos Transited Services: -
Package Name (NTLM only): - Key Length: 0 <truncated 1304 bytes> 5140892
```

Figure 5: Windows Security Event 4624 truncated

There is a reduction of the event to 43% of its original size and the removal of 1304 bytes for each one of these event occurrences.

The next example is Windows event 4634 which covers an account that is logged off. These events will be numerically close in amount to the login events, so the calculations use the same quantity of 500 events per day per user. The standard 4634 event id shown below in Figure 6 is approximately 596 bytes for Windows Server 2012.

```
Feb 6 01:16:22 hostname01.domain.com host01L-PDC01 - - - Feb 05 17:16:21 2017 4634 Microsoft-Windows-Security-Auditing
domain\host01L-PDC01$ N/A Success Audit hostname01.domain.com Logoff An account was logged off. Subject:
Security ID: S-1-5-18 Account Name: host01L-PDC01$ Account Domain: domain Logon ID: 0x2C1DAE7F Logon Type: 3 This
event is generated when a logon session is destroyed. It may be positively correlated with a logon event using the Logon ID value.
Logon IDs are only unique between reboots on the same computer. 377
```

Figure 6: Windows Security Event 4634 full text

For this event, the truncation is going to start with the text ‘This event is generated.’ This text is in every event and not needed for investigation or forensic purposes. The truncated event appears below in Figure 7.

```
Feb 6 01:16:19 hostname01.domain.com MSWinEventLog - - - Feb 06 06:46:18 2017 4634 Microsoft-Windows-Security-Auditing
domain\host02-PDC01$ N/A Success Audit hostname01.domain.com Logoff An account was logged off. Subject:
Security ID: S-1-5-18 Account Name: host02-PDC01$ Account Domain: domain Logon ID: 0x2C793CB5 Logon Type: 3
<truncated 190 bytes> 5140891
```

Figure 7: Windows Security Event 4634 truncated

There is a reduction of the event to 68% of its original size and the removal of 190 bytes for each one of these event occurrences. Below in Figure 8 is a summary of information for the two events truncated.

Author Name, email@addressa.brian.todd@gmail.com

Win Event ID	Message	Original Event size in Bytes	Truncated Bytes per event	Events per day (1000 users)	MB saved per day
4624	Success Audit – An account was successfully logged in	2269	1304	500,000	652.00
4634	Success Audit – An account was logged off	596	190	500,000	95.00

Figure 8: Event truncation calculations

Truncating events and removing information that provides no value for investigations is beneficial and should be performed. One straightforward method for seeing which events are candidates for truncation is to go through the logs and see which events have long repeated strings that are not beneficial. The benefits of truncation include the savings for bandwidth and processing. There can also be some advantages with storage; however, that may depend on the compression algorithms used.

One excellent example of truncation benefits is for companies that use Splunk. Splunk's licensing is based on the amount of data indexed during one day. Thus, removing extraneous data from events reduces a company's license usage and costs.

9. Logging Architecture

To help illustrate log collection architectures the paper will start at a small scale and then expand into a worldwide logging infrastructure.

1. One physical location
 - a. One host
 - i. One log source
 - ii. Multiple log sources
 - b. Multiple hosts
 - i. Each host has one or more log sources
2. Multiple physical locations
 - a. Multiple hosts
 - i. Each host has one or more log sources

When a startup has a single log source, then it is not necessarily cost-effective or advantageous to setup a logging infrastructure. Even when a startup has multiple log sources being generated from the single host a robust logging infrastructure may not be cost-effective.

Once a startup gets into the realm of multiple hosts with each host generating multiple log sources within a single physical location, a startup should have a logging architecture in place. An architectural solution is needed to provide a plan to scale for future growth. The company uses a system at that physical location to collect the logs. They may also use that system to do some processing of the logs or to send specific logs to other tools.

Another function that a log collector performs is rotating the log files so that they do not get too large and unwieldy. The options to rotate the logs are based on both a time threshold and a size threshold. Whichever limit is reached first will trigger a new log file to be created, and a compression of the previous data for later transmission to a storage server. A third function performed is the sending of the compressed logs to a secure storage location. The transmission of the compressed logs can be done once a day or even a few times per day as needed.

In more advanced log storage scenarios an organization may have various retention times for the different types of logs that are received. For example, a company may only want to keep firewall connection logs for sixty days and DNS logs for ten days. In some cases, the DNS events are only processed and not stored. There may be compliance and regulation issues that require that storing logs for longer periods of time.

Below in Figure 9 is a basic diagram of an effective logging infrastructure that a startup will use for each office.

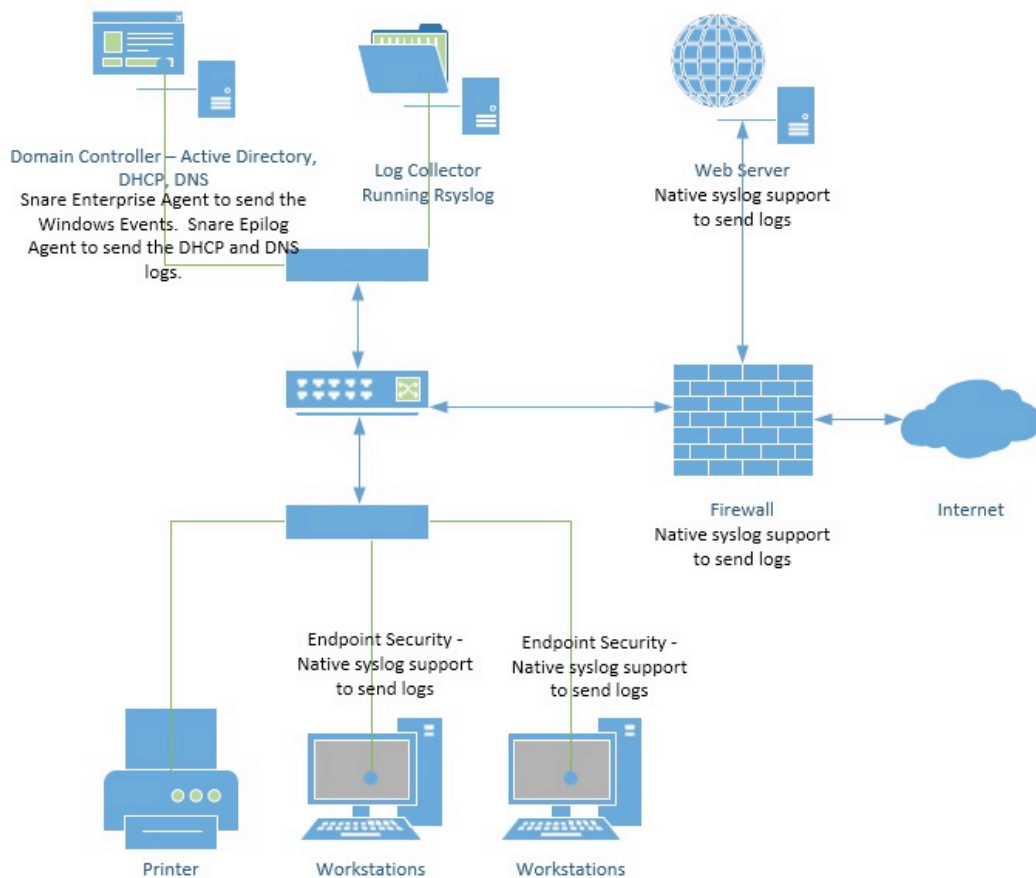


Figure 9: Example logging architecture for a single location

When it comes to a new physical location, a startup can simply add a log collector at the new site to collect the logs for that site. This log collector can also do some processing as well as sending logs to other tools. When sending logs from one physical location to another, they should be encrypted, with SSL/TLS being the best option. In the case of multiple log collector systems, a company can designate a single location to store the logs. This can be at one of their physical locations or using a cloud service. Figure 10 below depicts one option of what that communication can look like:

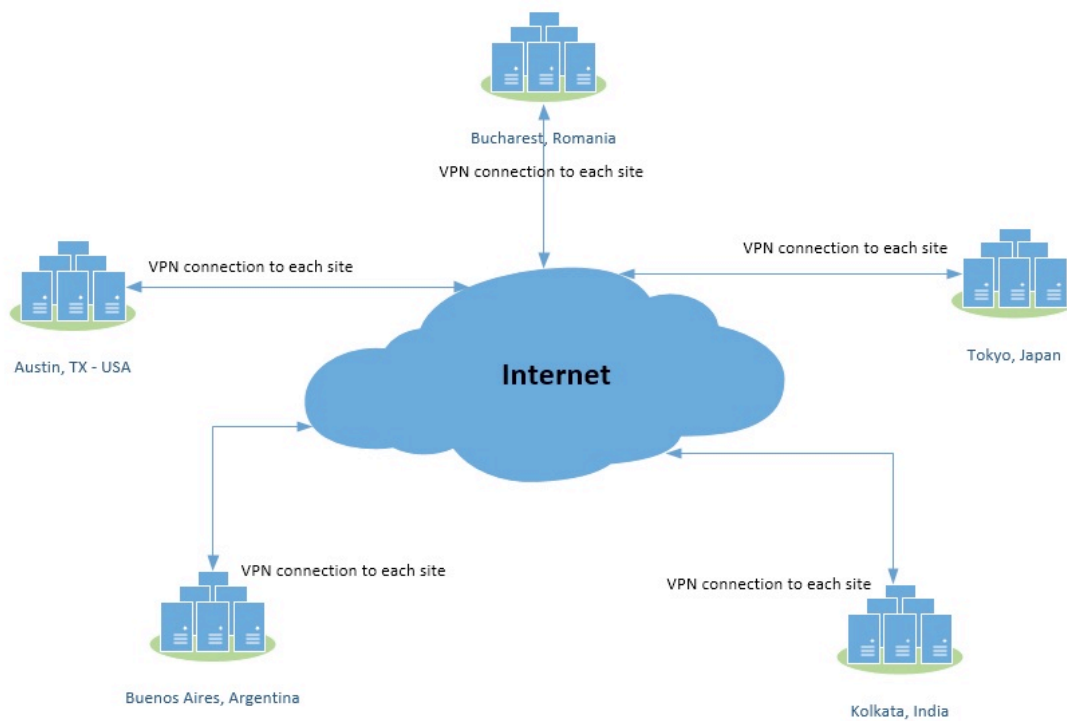


Figure 10: Example logging architecture for multiple locations

10. Correlations

Now that a logging infrastructure is in place a company can take advantage of the events they are receiving. If a company has a SIEM, then the SIEM solution will usually come with common correlations for alerting and investigation. This section will cover some correlations that are not common in SIEMs and that can be used and optimized for different environments.

Correlations use a combination of events from the same log source type and events across different log source types. A benefit of correlations is that when an event is viewed by itself, it may not be suspicious, but when multiple events are viewed together and correlated with each other, they can lead to something actionable. The paper will cover some correlations and look at real production log data to determine the value of the correlations.

10.1. Correlation 1

For the first correlation, logs are chosen from two different log source types. These log source types are DNS events and events from a network security device that blocks connections to/from public IP addresses that have negative reputations. The first consideration in the correlation is when a DNS lookup request for a domain name string is equal to or longer than 35 characters. The reason 35 characters was originally chosen was to avoid getting a lot of noise for normal size domains names and the thought that malware would want to use longer domain names to avoid collisions with pre-existing domain names. When this first consideration is met, the correlation then checks to see if the same system that made the DNS lookup request then tries to connect to an IP address that has a negative reputation within the next 30 minutes.

An information security engineer will want to check the benefits and effectiveness of any created correlations. To check the usefulness of this correlation, I downloaded five pieces of malware from www.totalvirus.com and ran them in a Windows 10 virtual machine using VirtualBox. I created a snapshot of the virtual machine before running the malware so that I could quickly return to my starting state for the next malware sample. I used Wireshark to capture the network traffic. I started the network packet capture, then surfed some websites for a few minutes, then I ran the malware and continued surfing the web for about ten more minutes to make sure the malware had time to complete. Figure 11 shows the results.

Malware Sample	DNS Request	DNS >= 35 characters?	IP addresses	IP address has negative reputation?
1	centromiosalud.es cfigueras.com	No	178.255.225.215 51.254.83.173	No
2	mjpgldtrdnzv.attlocal.net giiwmwbkqwj.attlocal.net firmubspthdmj.attlocal.net	No	Lookup was refused	NA
3	uploads.shantan.moe www.shantan.moe www.shantan.moe	No	107.180.26.160 107.180.26.160 107.180.26.160	Yes

Author Name, email@addressa.brian.todd@gmail.com

4	inoveinternet.com.br 74jhdrommdtyis.net	No	192.99.175.130 119.28.100.249	Yes
5	zabandan.com 74jhdrommdtyis.net	No	130.185.72.116 119.28.100.249	Yes

Figure 11: Malware results for DNS lookup and IP address reputation

If we look at each condition individually at first, we see that the results show that none of the malware samples had a DNS request that was greater than or equal to 35 characters. Thus, we would not have received any alerts based solely on our first condition. For our second condition, we see that 3 of the malware samples would have triggered an alert if it was an isolated rule (one malware sample didn't resolve an IP address). When correlated together no alerts would have been generated.

This correlation gives us false negatives for the four malware samples that were successfully run. It appears that this correlation may not be useful in its present form. Further investigation can be done with more malware samples to get an idea about domain names and lengths. Another avenue of research would be to get a statistical table of DNS lookups and their character lengths. We can tweak the first condition to reduce the number of characters or decide that it is not worthwhile to use this correlation.

10.2. Correlation 2

For this correlation, we are using events from two different log sources. The first log source is from an IDP system. The IDP system will send log events when an IDP signature is matched, and the connection is blocked. One of the event fields is the source IP address which will be our main point of interest. The second log source is a network security device that blocks connections to/from public IP addresses that have negative reputations.

For this correlation, we will take the previous seven days of history for the IDP events and sort them in descending order based on how many events were received per internal IP address. Then we will do the same for our events from the IP address reputation based network security system. Now that we have our lists for these two different log sources we run the correlation that compares the two lists, and any internal

IP address that appears in both lists will trigger an alert. Figure 12 shows the results for the IDP events (left side) and the IP address reputation events (right side), which are limited to twenty-five entries for brevity.

1	source_address	percent		1	LOCAL_IP	percent	
2	1	.74	17.75027	2	1	.45	25.13328
3	1	.88	17.67052	3	1	.17	6.321401
4	1	.125	9.247893	4	1	.11	5.962354
5	1	.210	8.471168	5	1	.76.50	5.418344
6	1	.243	4.802524	6	1	.15	5.135459
7	1	.31	4.545927	7	1	.43	4.950495
8	1	.111	2.559035	8	1	.8	3.895115
9	1	.146	1.206699	9	1	.10	3.818953
10	1	.23	1.047193	10	1	.28	3.590469
11	1	.19	0.998648	11	2	.66	3.187901
12	1	.63	0.901557	12	1	.112	3.035578
13	1	.15	0.859947	13	6	.250	2.937656
14	1	.25	0.842609	14	1	.63	2.633011
15	1	.96	0.797531	15	1	.2	2.567729
16	1	.124	0.766323	16	1	.229	2.317484
17	1	.18	0.766323	17	1	.32	2.001958
18	1	.123	0.752453	18	1	.43	1.305625
19	1	.6	0.721246	19	1	.20	1.088021
20	1	.129	0.710843	20	1	.10	0.772495
21	1	.65	0.68657	21	1	.125	0.739854
22	1	.66	0.669233	22	1	.23	0.718094
23	1	.031	0.60335	23	1	.125	0.707214
24	1	.37	0.60335	24	1	.34	0.696333
25	1	.16	0.596415	25	1	.14	0.696333
26	1	.163	0.575609	26	1	.11	0.685453

Figure 12: IDP events (left) and IP address reputation events (right)

To get an idea about how the time window can affect this correlation I also ran the correlation for durations of three days and one day. For three days the correlation also found the same IP address as showing up in both lists. For one day the correlation did not come up with any matches. The time window is something that an information security engineer can tune for their particular network.

If we had to investigate all the hosts that are in these lists it would take a lot of time, so that is why we are using this correlation to direct us to systems of concern and

where we can concentrate our investigation. The IP address reputation events can have a lot of false positives because an IP address can host multiple sites, thus if one site is on the negative reputation list then visiting any of those sites that use that same IP address will cause an event to be generated. I looked into this a bit further, and this can be exacerbated for content delivery networks which have hundreds to thousands of sites using a single IP address. This shows us why this correlation can help narrow down our focus.

Getting back to the results above we see one internal IP address that is present on both lists. I investigated the host and saw that the anti-malware program was disabled. The anti-malware definitions were four months out of date, and the last scan that was run was also four months ago. So, I enabled the anti-malware scanner, updated the definitions and then ran a full system scan. The results turned up two items. One was a potentially unwanted program that was a web browser plugin, and the other was malware. The anti-malware program was able to remove both items. So, this correlation was a true positive.

To gather more data about the usefulness of this correlation we look at the top five hosts for the IDP events and the IP address reputation to see what the results are for them separately. For the top five systems reported by the IDP events, only one of them had malware on it. This is the one that was identified by the correlation. The other four systems are engineering development systems, and two of them were running services that we found were no longer needed, so we disabled those services. All of these systems were located in the US.

For the top five systems reported by our IP address reputation system, we found that three of them were only on the guest wireless network and they were not issued by our company. It turns out they were visitors to that site. Interestingly enough, given their DNS names, all three are smart devices running iOS or Android. The last two IP addresses are for the same windows system that was on the guest wireless and later on the wired production network. When I investigated the system, I found that it did not have the standard company image on it. It did have an anti-malware program, but it was one that no one on the security team had heard of before, and it was also not an approved

application. The anti-malware program is called 360 safeguards, and it was not reporting any malware on the system. Since the system was not in compliance with the company's policies the IT team will reimage the system.

An interesting topic that I discovered when looking at the results for the IP address reputation events had to do with the composition of the devices. Eight out of the top ten systems that were reported trying to connect to IP addresses with negative reputations were smart devices (mobile phones) running iOS or Android. I think that people, in general, tend to install many different apps on their smart devices and there is not much awareness of what some of these apps are doing behind the scenes. Just looking at the data, smart devices and their security seems like an area that can have more focus on in the future.

The other item that we discovered was a system that was not using our companies image. So, it did not have our approved security software on it and was not subject to the other controls we had in place. So, for this case I am going to call this a false negative for our correlation since there was something that needed to be handled with this system that our correlation did not trigger an alert for.

Given the results I think that this correlation will provide valuable alerts and a good way to focus priorities and time. A security engineer can review the alerts that this correlation provides with a higher priority than just investigating individual IDP events or individual IP address reputation events.

10.3. Correlation 3

For the next correlation, we will look at two events which can come from the same log source type or different log source types. The first step in the correlation is when an event that a user gets added to an administrator group is received. When this event occurs the correlation then checks for any events received within 30 minutes that are for log deletion events. This correlation is targeted to detect an attacker in a company's network that is trying to cover their tracks by erasing log files.

To get an idea of the effectiveness of this correlation I used real production log data. In the past 12 months, there are 36 instances of accounts getting added to

administrator groups. So, if we were only alerting on events where a user gets added to an administrators group, we would have had 36 alerts. I then checked for events that indicated that log files were cleared or erased. There were no events within the last 12 months for log files being cleared or erased. So, when we add in our second consideration, then we would have 0 alerts to investigate.

These results are from a smaller production network, so a larger network or a network that is more of a target for malicious behavior might find this correlation more useful than the results shown in this paper. An engineer can further optimize this correlation by altering the time duration from when a user gets added to an administrator group and how long to look for a correlating log clearing or deleting event.

10.4. Correlation Summary

This section discussed the importance and usefulness of correlations. It is a good practice to review all the correlations that are configured in a SIEM to decide which are relevant to the business and to optimize them for the company's situation. As shown above, there may be correlations that need to be altered to have a better desired effect or correlations that are not practical or useful in certain production environments.

11. Conclusion

Logging is an essential part of any IT infrastructure. A company needs to pay attention to their logs to better understand what is going on in the network. When an analyst looks through the logs, they can discover misconfigured settings, systems that are starting to show issues, and security violations or breaches.

This paper presented the different types of file formats and the protocols used for logging. It reviewed a few different options for collecting the logs from various hosts. The option for collecting logs covered the native support within a system and using an agent on the host to gather the logs and send them to the configured destination. Once those essential items are in place, an analyst can look through the logs and then start to prune the events that do not provide any value. While reviewing the logs to prune events, an analyst will also want to truncate any extraneous information for the events but do not provide any value.

Author Name, email@addressa.brian.todd@gmail.com

This paper then went through an example of a startup and how that startup implemented their logging infrastructure as the company grew. The architecture initially went over logging for only a small number of hosts in a single location. Then the business developed, and the discussion went through the logging architecture for a company that has multiple physical locations worldwide and with many hosts that need logging at each physical location.

After the logging architecture discussion, the paper talked about tools that will perform indexing, correlation, and other data mining activities on the logs. These tools are invaluable due to the sheer number of logs generated. There is no way that a human would be able to go through all the logs by hand. So, it is important to have an application such as a SIEM or indexing application to process these logs and provide overviews as well as alerts. The paper covered correlations and their uses in event monitoring and response. The correlations included situations where events can come from different log source types and reasons why they should get reviewed for their usefulness in each deployment.

Finally, the discussion covered the retention of logs. There may be different requirements for the duration of time legally required for storing logs. For industries such as the payment card industry, there may be longer durations needed for storing logs that are legally required.

All of these aspects of logging need to come together to form a robust logging infrastructure. Once a company has an effective logging infrastructure in place, an analyst will have a wealth of information that will enable them to make more informed decisions. The abundance of information that logging reveals allows an analyst to be able to be more proactive with investigations and problem-solving. With the logging tools and applications that perform data mining, indexing, and correlation alerts will be generated to notify an analyst about critical events in the network. A robust logging infrastructure can solve issues faster and make users happier and analysts jobs easier.

References

- Chuvakin, Dr. Anton A., Schmidt, Kevin J., Phillips, Christopher. (2013). *Logging and Log Management*. Waltham, MA: Syngress
- Ranum, Marcus J. System Logging and Log Analysis. Retrieved 6 February 2017, from http://ranum.com/security/computer_security/archives/logging-notes.pdf
- Kent, Karen., Souppaya, Murugiah., (2006). NIST Special Publication 800-92 Guide to Computer Security Log Management. Gaithersburg, MD: U.S Department of Commerce
- The CEE Editorial Board, (2010). Common Event Expression Architecture Overview. [Online]. Retrieved 8 February 2017, from http://cee.mitre.org/docs/CEE_Architecture_Overview-v0.5.pdf
- Smith, Randy F. (n.d.). Randy Franklin Smith's Ultimate Windows Security. [Online]. Retrieved 10 February 2017, from <https://www.ultimatewindowssecurity.com/default.aspx>
- Gerhards, R. (2009). The Syslog Protocol. Retrieved 10 February 2017, from <https://tools.ietf.org/html/rfc5424>
- Davis, David. (2013). Configuring a Syslog Agent in Windows Server 2012. Retrieved 9 February 2017, from <http://techgenix.com/configuring-syslog-agent-windows-server-2012/>
- syslog. (n.d.) In Wikipedia. Retrieved 10 February 2017, from <https://en.wikipedia.org/wiki/Syslog>
- Logging Cheat Sheet. (2016). Logging Cheat Sheet. Retrieved 13 February 2017, from https://www.owasp.org/index.php/Logging_Cheat_Sheet

Appendix A – Tools

1. Log Collection

1.1. Agents for endpoint collection

1.1.1. Syslog

1.1.1.1. rsyslog agents

1.1.1.2. <http://www.rsyslog.com/windows-agent/about-rsyslog-windows-agent/>

1.1.1.3. Snare agents

1.1.1.3.1. <https://www.intersectalliance.com/our-product/snare-agent/>

1.1.2. Windows events

1.1.2.1. Snare enterprise agent

1.1.2.1.1. <https://www.intersectalliance.com/our-product/snare-agent/operating-system-agents/snare-agent-for-windows/>

1.1.2.2. Datagram syslog agent

1.1.2.2.1. <http://www.syslogserver.com/syslogagent.html>

1.2. Servers for collection

1.2.1. Syslog

1.2.1.1. rsyslog

1.2.1.1.1. <http://www.rsyslog.com/>

1.2.1.2. syslog-ng

1.2.1.2.1. <https://www.balabit.com/network-security/syslog-ng/central-syslog-server>

1.2.1.3. Snare Server

1.2.1.3.1. <https://www.intersectalliance.com/our-product/snare-central/>

1.2.1.4. Datagram Syslog Server

1.2.1.4.1. <http://www.syslogserver.com/index.html>

2. Log Processing

2.1. SIEM

2.1.1. LogRhythm

2.1.1.1. <https://logrhythm.com/>

- 2.1.2. QRadar
 - 2.1.2.1. <http://www-03.ibm.com/software/products/en/qradar>
- 2.1.3. ArcSight
 - 2.1.3.1. <https://saas.hpe.com/en-us/software/siem-data-collection-log-management-platform>
- 2.2. Indexers
 - 2.2.1. Splunk
 - 2.2.1.1. <https://www.splunk.com/>
 - 2.2.2. loggly
 - 2.2.2.1. <https://www.loggly.com/product/>
- 3. Log Storage
 - 3.1. rsyslog
 - 3.1.1. <http://www.rsyslog.com/>
 - 3.2. syslog-ng
 - 3.2.1. <https://www.balabit.com/network-security/syslog-ng/central-syslog-server>
 - 3.3. store in AWS
 - 3.3.1. Use any of the above applications and Amazons S3 (Simple Storage Service)
 - 3.3.1.1. https://aws.amazon.com/?nc2=h_lg