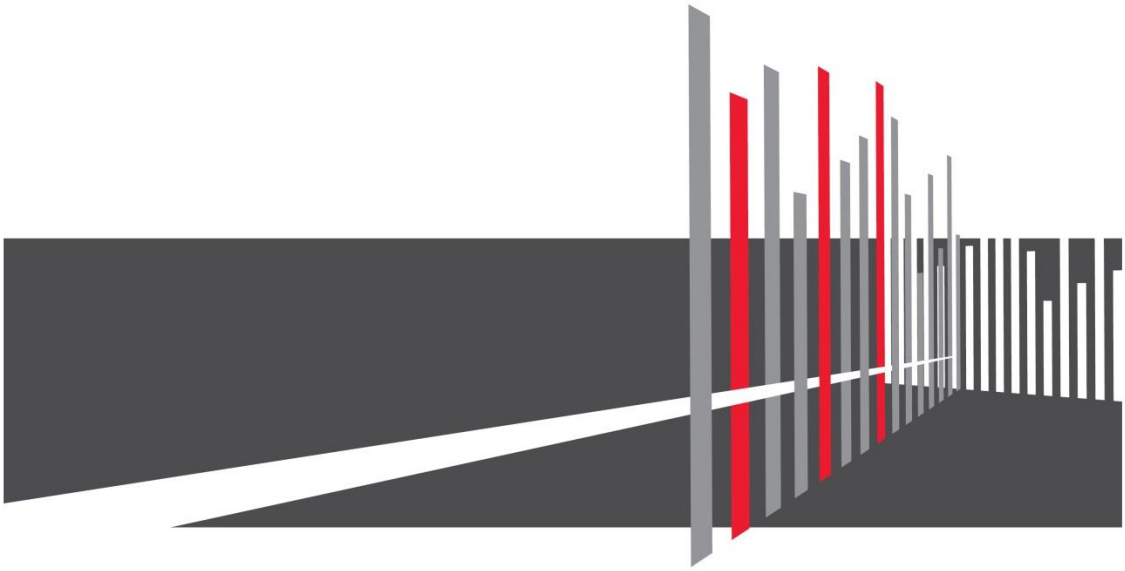


Centre for the Protection of National Infrastructure

Effective Log Management

Author: Tom Goldsmith

Context Ref: 20140402 - CPNI-CTX - Effective Log Management

Date: 2nd April 2014

Tel: +44 (0) 20 7537 7515

Email: response@contextis.co.uk

Contents

List of Tables

List of Figures

1 Executive Summary

Log files are historical records of the running state of hardware and software, storing information on how they are used, errors that occur and application-specific events which detail how users interact with them. Routine review of this information can provide system administrators and computer security teams with insight into how effectively the business is operating and where configuration errors may be causing issues on the network so that they can be remediated before they have wider impact.

Log records are also an immensely valuable source of information for computer security purposes, but their value as part of a corporate intrusion detection and incident response process is largely misunderstood by many organisations; logs are either not collected at all or are collected without consideration for how they might be used should an incident occur. This paper has been commissioned by the UK Centre for the Protection of National Infrastructure (CPNI) to inform the reader about the value of this data, discuss the key sources of log records and demonstrate how they can be used to support an efficient response to a computer intrusion event. Case studies of actual incident response events will be presented to demonstrate how log files can form a critical part of the analysis process and how failure to fully take advantage of sources of log data can limit the effectiveness of response.

Developing a log management strategy to enhance an organisation's computer security posture is not easy and there are a number of hurdles to overcome. The number of devices that generate logs on a network can be overwhelming and storage strategies can be both expensive and complex to implement. What logs should be collected? For how long should they be stored? How should they be stored to best ensure the security and integrity of the data? How can log files be used proactively to look for the malicious "needles" in the "haystack" of data available?

The lesson for organisations is simple: logs are an evidence source of potentially vital importance, but effort is required to exploit them for maximum value. They are a dataset which requires little effort to start collecting at some level, are greatly helpful during the investigation of security breaches. They can also be used by forward facing organisations to identify, track and mitigate attacks before any damage is caused. However, in all cases a clear log management strategy is essential to ensure the data is of sufficient quality and is organised appropriately. Organisations should also consider testing their ability to recover and utilise log data from appropriate devices in preparation for the situation where they need to respond to a real incident.

2 About Context

Context has a client base including some of the world's most high profile blue chip companies and government organisations. Our strong track record is based above all on the technical expertise, professionalism, independence and the integrity of our consultants. Context's comprehensive portfolio of technical services sets the standard for the security industry. In the ever-changing world of security, many of our clients choose to retain our services year after year.

Context has a dedicated incident response and investigation team, which works with clients to detect, investigate, understand, respond to, and protect against nefarious activity on their networks. This team does not focus on generic malware, but offers services designed to counter the most sophisticated attacks targeted against our clients. Such attacks are likely to cause substantial damage unless met with a decisive, informed response. Our team comprises experts who can provide substantial experience and expertise on a business as well as a technical level.

Context is proud to be part of the Cyber Incident Response scheme run by CPNI and CESG, the Information Assurance arm of GCHQ. Context exceeded the rigorous requirements set for entry and we continue to evolve our capabilities and expand our offerings to keep one step ahead of the attackers.

The advice provided in this document is based on our experience of working with organisations who have suffered significant network intrusions. Due to the endless combinations of different infrastructure and operating systems this document is intended to be a practical, informative starting point, rather than a complete solution for organisations wishing to improve their log management capabilities.

This material is provided for general information purposes only. You should make your own judgement as regards use of this material and seek independent professional advice on your particular circumstances. Neither the publisher, nor the author, nor any contributors assume any liability to anyone for any loss or damage caused by any error or omission in the work, whether such error or omission is the result of negligence or any other cause.

3 Introduction

Log files are historical records of the running state of hardware and software, storing information on how they are used, errors that occur and application-specific events which detail how users interact with them. Routine review of this information can provide system administrators and computer security teams with insight into how effectively the business is operating and where configuration errors may be causing issues on the network so that they can be remediated before they have wider impact.

In modern computing environments, almost every type of hardware device and software product provides some functionality to log events relating to interactions between users and devices or other machine to machine communication. While logs were originally designed to troubleshoot technical issues, log sources have been augmented to provide the capability to inform administrators of network performance issues and to assist in the investigation of computer security incidents.

Although the wide availability of logging functionality provides a rich source of data that can be used to detect computer intrusions as they occur or to determine the actions of an attacker after the event, it also presents a number of strategic challenges to an organisation in the form of log aggregation, retention and analysis policies. On top of this, the security and integrity of log records has also become increasingly important as they represent a potential target for intruders who seek to cover their tracks from security teams by removing evidence of their activity.

While the concept of log file generation is nothing new to system and network administrators, this paper seeks to inform both technical and managerial audiences of the importance of having an established log management and analysis strategy within the corporate environment. Additionally, it will demonstrate how a comprehensive approach to logging can assist in recovery after a computer security incident and how organisations can start using these sources as part of a proactive approach towards ongoing security monitoring.

The advice within this document is intended to be used in conjunction with the guidance provided by the 20 Critical Security Controls for Effective Cyber Defence coordinated by the Council on CyberSecurity, specifically Critical Control 14: Maintenance, Monitoring and Analysis of Audit Logs[1].

3.1 The Challenges of Effective Log Management

When defining a log management strategy within an organisation, a balance will likely have to be struck between breadth and depth of data collection and how it will be stored and analysed. While the ideal policy may be to fully log all possible data sources and retain this information for extended periods whilst simultaneously analysing them for anomalies, this is rarely practical for many organisations. Therefore, it is critical that the inclusion of each data source is blended into an overall incident response process to create a layered approach that ensures that the best possible view of an intrusion can be determined as quickly as possible. This strategy should be extended to store

[1] http://www.cpni.gov.uk/advice/cyber/Critical-controls/in-depth/critical-control14/

data for as long as possible under organisational constraints such as policy-based considerations or technical limitations like available data storage volumes.

Common challenges associated with effective log creation, analysis and management include:

- **Numerous data sources.** Logs are available from many devices within a network including workstations, servers and network infrastructure such as routers and wireless access points. Additionally, many of these sources often generate several kinds of log records. For each log source, a process to obtain, de-duplicate and store the data must be established.

- **Data volumes and retention.** The volumes of data generated by these sources will naturally increase with the number of nodes and amount of user activity on a network. The storage of log records for some sources may become problematic on busy networks, particularly those relating to external communication such as proxy and Domain Name Service (DNS) resolution logs. This issue is compounded by log retention restrictions, where keeping months or years of data to ensure that extended intrusion activities can be uncovered may involve considerable storage requirements.

- **Effective analysis of log data.** While comprehensive logging across a range of assets is desirable, effective intrusion detection relies on this data being analysed on a regular basis. Therefore, it is critical that trained staff and resources are made available for ongoing log review, and that the records are stored in an organised manner so that they may be retrieved by incident response professionals in a timely manner. Log file analysis must also be integrated into an organisation's wider network security monitoring process to provide analysts with additional data points for correlation. For example, Intrusion Detection System (IDS) alerts should be correlated with host and network log data to indicate time ranges of intrusion rather than relying on discovery processes across millions or even billions of log events.

- **Data formats and content.** Depending on the hardware and software deployed within an organisation, log file formats may be generated in a range of plain-text, mark-up language and binary formats. While some records may be easily reviewed or parsed using scripts or log viewing tools, some may only exist in proprietary formats that require specific applications to ingest and analyse them.

- **Inconsistent time references.** Log analysis for the purposes of intrusion detection and incident response relies heavily on accurate timestamp registration to allow correlation of events across a range of data sources in order to create a full timeline of intruder activity. Each device must have access to a reliable time source to enable this correlation, preferably using a combination of multiple synchronised internal and external time servers to ensure accuracy. The location of network assets in multiple time zones can also present issues, highlighting the importance of using a common time zone (e.g.

UTC) for all log sources.

- **Security and integrity of log data.** Log files themselves may be a target for intruders who seek to cover their tracks by modifying or deleting events during and after their attack. To ensure that the data is available for analysis, log files should be ideally be stored in a centralised location so that their integrity can be more easily managed to reliably establish a picture of the intruder's activities.

- **Third-party data sources.** Where third-party providers are used to supply network services, agreements to obtain log records pertaining to network intrusions will need to be established prior to any incident occurring in order to maximise the speed and efficiency of response. The use of cloud services may be particularly problematic in this regard, as log files may not be accessible to the end customer.

Although these challenges exist, they can generally be mitigated by establishing policies and dedicating resources to store events using a layered approach to logging. While there may be limitations on the amount of data that can be stored within the organisation, it should be noted that comprehensive logging over the longest timescale possible will allow incident responders to generate a detailed picture of an intruder's activities. Many compromises are only discovered weeks, months or in extreme cases, years after the initial breach and require historical data sets to properly determine their attack lifecycle.

3.2 The Role of Log Management in Incident Response

The correlation of time-synchronised log events across a range of computer assets is a critical part of any incident response activity as it assists an organisation in assessing the extent and impact of a network compromise as well as informing what steps may be necessary for mitigation. Various security devices may provide elements of the picture during an attack, but may not have full visibility over a network or possess the correlation capabilities to fully describe the attacker's activities. For example, IDS alerts may identify when an intruder performs an SQL injection attack against web application, but may not have the alert rules to identify subsequent uploads of tools into temporary storage within the database, tool use to elevate privileges on the host and subsequent lateral propagation across a network through exploitation of trust relationships between servers. By correlating the IDS alert with log files from the web, database and authentication servers as well as events generated on the workstations, greater visibility into the extent of the compromise can be established.

3.3 The Role of Log Management in Intrusion Detection

While log files are typically used during incident response to determine the 'how, when and what' of an intruder's activities, they can also be used proactively to monitor the security posture of a network. By analysing the data generated by devices on a regular basis, security teams may be able to detect and respond to the initial stages of a security incident as they occur. For example, the identification of out-of-hours creation of privileged user accounts may indicate that an attacker has gained a foothold within a network and is establishing further access.

Although intrusion detection requires greater up-front resource commitment both in staffing and time allocation, proactive analysis can detect events that may not be highlighted by automated systems and can be used to limit the wider impact and cost of a security incident.

4 Recommended Log Sources and Fields

While log data can be generated by large numbers of devices, they generally relate to one of four source classes:

- Services that provide functionality to users
- Infrastructure supporting the network
- Host devices
- Remote connection services

When establishing a log management policy, it is important to consider what constitutes the minimum level of data which should be stored to enable intrusion detection and response procedures. Although storing all data for every possible source may be the ideal for full response purposes, it is acknowledged that this might not be practical in all situations. This section provides a product-agnostic view of the primary log sources and the recommended minimum fields that should be stored if they are available, describing how they might be used during analysis to contribute to the overall picture of a security incident. Although some log sources are designed to generate records that can be easily ingested into analysis frameworks, certain products may store information in a human-readable format that is more complicated to process in bulk or a proprietary format that requires a specific reader to analyse. The format and suitability of each log source should be evaluated prior to being incorporated into a log management strategy to ensure that it delivers the appropriate information in a usable format.

For all log sources, both “failures” and “successes” should be logged wherever possible, as some attacker activities may not be considered anomalous and therefore would not be stored as exceptions. Without logging successful actions, it will be extremely difficult to compile an accurate picture of how an attack was executed and what follow-on activities were conducted.

This document focuses primarily on logging that can be enabled to facilitate intrusion detection and incident response on hardware and services run within the organisation. An increase in the availability and popularity of cloud and managed services means that logging for these data sources may not be readily controlled by local network administrators. The following recommendations will still be relevant in situations in which a third-party provides a service, although the availability and accessibility of particular records may vary. Therefore, it is important that the availability of log data from these services is discussed and agreed with the provider in advance of potential security incidents.

4.1 Network Services

4.1.1 Proxies

Proxy servers are intermediate devices through which remote hosts such as web servers are accessed from within a network. If employed on a network, web proxies can provide a source of log data for outbound network traffic as they store information on requests

to web servers including the Uniform Resource Locator (URL) requested and the status of the response. This information can provide an indication of outbound requests sent from malware to obtain tasking and to send data back to an attacker, along with the IP address of the affected host on the network.

Recommended field	Usage
Time and date stamp	Used for time line creation, filtering and correlation with other log data. It is important to store the time zone offset of this field if it is available to enable correlation with other log sources.
Client IP address	Storing the IP address of the internal client issuing outbound requests can assist with attribution to physical machines on the network when associated with other log sources such as DHCP lease log records.
Requested URL	The address for the remote resource (typically the domain name, directory and file on a web server) can be used to build up a picture of host activity during an incident. They can be used for direct matching of known malicious requests or to determine how data is sent and received between remote servers.
Referrer	Used to determine the origin of outbound requests if they have been triggered by the user clicking on a hypertext link. This is particularly useful for determining the origin of malware downloaded from an external resource.
User agent	Analysis of user agent strings can be used to detect anomalies in corporate environments running on standard builds, particularly when malware utilises custom user agent declarations.
Session size	Used to determine how much data is being transferred across network boundaries and how it might relate to malicious activity. For example, large HTTP POST requests may suggest data exfiltration activity is occurring.

Recommended field	Usage
Server response code	Used to determine whether the requested resource was actually delivered back to the client. This is particularly useful for determining whether requests issued by malware for additional tasking and upgrades are successful.
Content type	The reported content type of the request can be used to determine the type of data downloaded by the client, such as requests for executable files. This field can also be used in conjunction with other data sources to detect mismatched content transfer types, which may indicate obfuscated malicious activity.

Table 1: Recommended fields for proxy request and response logging

The figure included below provides an example of some proxy log entries recording information such as the time stamp, client IP address, the requested URL, server response code and user agent. In this case, the logs are tracking outbound requests for tasking performed by a piece of malware.

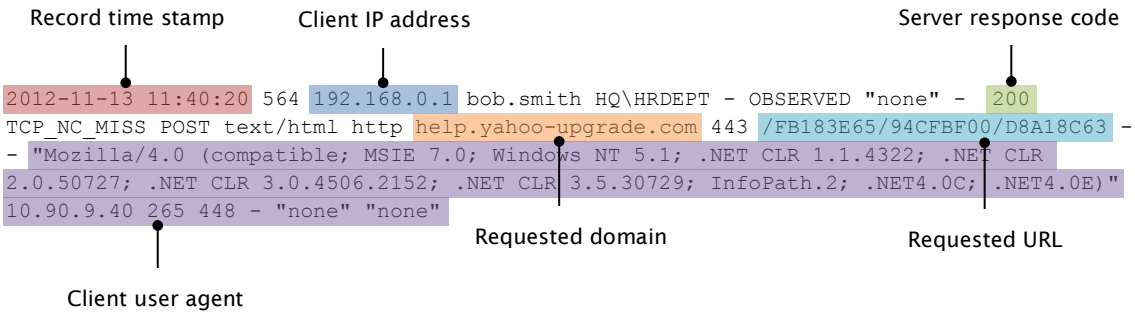


Figure 1: Proxy server log entry example

4.1.2 Web Servers

Web servers represent a potential attack vector either through the exploitation of vulnerabilities in the applications they are serving or through the web server software itself. Servers that are connected directly to an external network such as the internet can provide an initial foothold into a network, whereas internal web servers may be exploited for lateral propagation purposes. Web servers are also commonly compromised to act as “watering holes”, where an attacker will use their access to serve exploit code to visitors of corporate websites.

As most web servers are connected to the internet, they also represent an attractive target for “staging” purposes, where data obtained by an attacker is temporarily stored

before onwards transmission to an external host. In addition to the logging of the requests to the server itself, it is important to ensure that host logging is activated to detect this activity. For more information on host logging, see section 4.3 of this document.

During incident response analysis, the log files of requests to a web server can be examined for information pertaining to initial reconnaissance or remote exploitation activity such as attempts to compromise associated servers such as SQL injection attacks. Web server logs may also provide insight into outbound requests that may show data exfiltration activity from servers being used as data staging host.

Recommended field	Usage
Time and date stamp	Used for time line creation, filtering and correlation with other log data. It is important to store the time zone offset of this field if it is available to enable correlation with other log sources.
User agent	Analysis of user agent strings can be used to detect anomalies in corporate environments running on standard builds, particularly when malware utilises custom user agent declarations.
Referrer	Used to determine the origin of inbound requests if they have been triggered by a visitor clicking on a hypertext link from another resource. While there are many legitimate reasons why referrer fields may be blank, malicious tools will often incorrectly set headers including the referrer header and may provide opportunities for anomaly detection.
Client IP address	Storing the IP address of the remote client issuing inbound requests may provide an indication of the source of an attack, although intruders will often obfuscate their true origin.
Server response code	Used to determine whether the requested resource was actually delivered back to the client. This is particularly useful for determining whether attacker requests to web servers successfully return data (for example, if they are able to authenticate and access secure areas within web applications).

Recommended field	Usage
Requested URL	The URL requested by the client can be used to track an attacker's interactions with the web server. In some cases, encoded commands created by automated tools can be recovered from these records.
Size of response	Used to determine the actual amount of data transferred from or to the client IP address, useful in corroborating other information from the log files, such as the server response code.
HTTP request type	The HTTP request type can be useful to provide a reference of directionality on traffic to and from web servers. For example, a GET request type will generally represent a client request for a resource from the web server, whereas a POST request type generally indicates that a client has sent some data to the server.

Table 2: Recommended fields for web server request logging

The figure included below provides an example of some web server log entries recording information such as the time stamp, requested URL, client IP address and request type of an external user. In this case, the logs are tracking inbound requests attempting to exploit a web application vulnerability using brute force techniques.

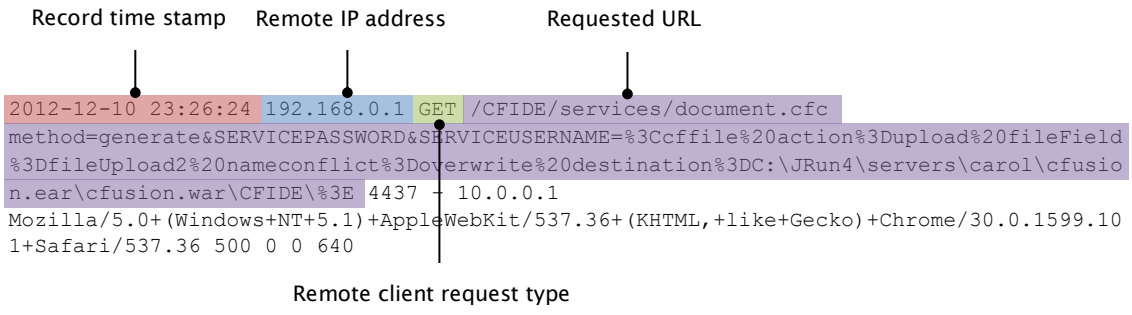


Figure 2: Web server log entry example

4.1.3 Mail Transfer Agents

A common vector for host compromise is spear phishing or "content delivery" attacks, where an attacker sends emails to a target containing malicious software as an attachment. Logging of metadata and content pertaining to inbound email traffic can help establish which users have received emails with malicious content, when the

attacks occurred and other information that may be useful for ongoing detection purposes such as email subject lines that may be reused or provide insight into the methodologies used to entice users into opening the attachments.

In some cases, malicious software may have the capability to use Simple Mail Transfer Protocol (SMTP) as a command and control or data exfiltration channel. When investigating incidents involving this type of malware, logs of outbound email sessions can also be useful to determine the frequency and type of data sent from the network.

Recommended field	Usage
Time and date stamp	Used for time line creation, filtering and correlation with other log data. It is important to store the time zone offset of this field if it is available to enable correlation with other log sources.
Client IP address	Useful to store to discover client misuse of the mail server or issues with access control lists.
Server IP address	Used to provide a point of reference in the event that data is aggregated from more than one mail server.
Recipient email address	Used to track the end recipient, allowing for local analysis if the email is discovered to contain malicious content. This field is also useful to monitor the recipients of content delivery attacks over multiple campaigns.
Source email address	Used as a primary query point for mail log analysis. Although the source address of an email can be spoofed, this information can be useful to develop signatures for malicious mail senders and to track attack campaigns.
Email subject	The subject line of an email can be used as a form of identification for malicious emails. Additionally, if a piece of malware is using email as an data exfiltration mechanism, the subject line can provide an additional indicator of activity.

Table 3: Recommended fields for Mail Transfer Agent logging

The figure included below provides an example of some mail server log entries recording information such as the time stamp, sender email address, email subject line and recipient email address.

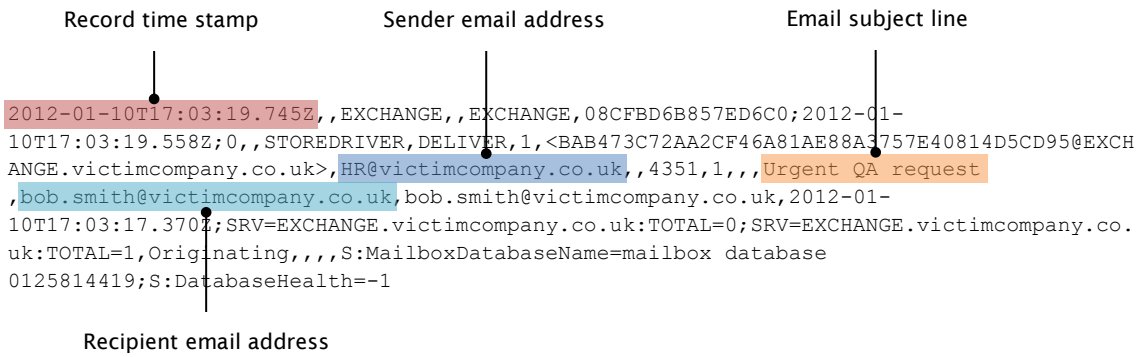


Figure 3: Mail server log entry example

4.1.4 Database Servers

SQL injection (SQLi) attacks are a common method for exploiting weaknesses in web application software to obtain information from, or control over, a database server. SQLi attacks can be potentially used to display the contents of protected database tables, add new user accounts to enable direct remote access, provide the ability to upload and execute attack tools and can even provide full remote desktop access to the server.

Logged database transactions can enable an incident responder to determine what commands were sent to the database and whether or not they were successful. For attacks where the primary objective is to obtain the information stored within the databases themselves, log data can provide insight into what access the attacker has obtained and the impact of the activity. If an intruder attempts to upload additional tools to the server via SQLi, database logs can potentially be used in conjunction with web server log records to extract and analyse the malicious payloads even if the original files have been deleted from the host.

Database servers may also be targeted directly for the information they hold, depending on how they are used within the enterprise and the objectives of the intruder. Although access to the databases may be achieved remotely with the appropriate credentials, it may be preferable for an attacker to obtain local access to the hardware on which the server software runs to improve the persistence of their access. In addition to the logging of database transactions on the server, it is important to ensure that host logging is activated to detect attacker activity on the servers themselves. For more information on host logging, see section 4.3 of this document.

Recommended field	Usage
Time and date stamp	Used for time line creation, filtering and correlation with other log data. It is important to store the time zone offset of this field if it is available to enable correlation with other log sources.

Recommended field	Usage
Query	The SQL content of the query sent to be executed by the server can be used to determine the actor's activity. In cases where automated tools are used to upload code to databases for ongoing exploitation, this field may be useful to reconstruct and analyse the attacker's toolset.
Username	The name of the authenticated account performing the query can be used to provide a pivot point for determining all attacker activity if the username is discovered to have been compromised or created during an intrusion.
Transaction message	The description of the event as logged by the database server. This can provide useful information for determining the success status of the executed query.

Table 4: Recommended fields for database server logging

The figure included below provides an example of database server log entries recording information such as the query executed and transaction message recorded in response. In this case, warning messages are generated noting that queries are taking longer to run than a pre-set threshold. Although in this case this message is unlikely to be indicative of malicious activity, databases often restrict query times to prevent long-running queries from impacting the performance of the server, and exceptions to this would be logged as errors.

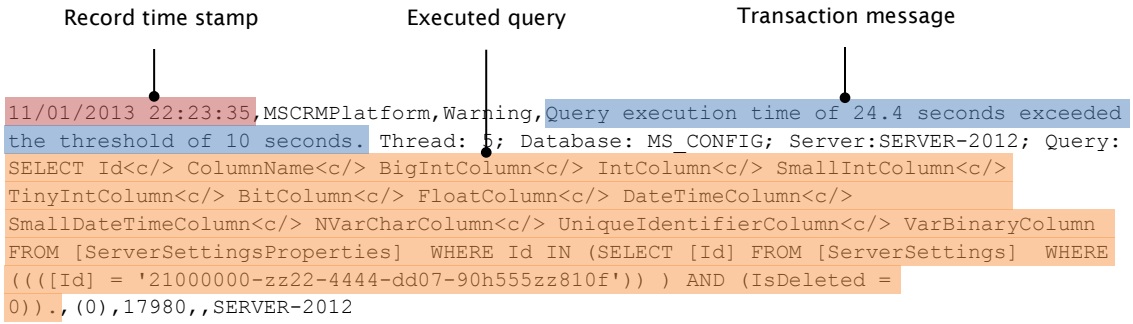


Figure 4: Database server log entry example

4.1.5 Other Services

Although not as critical as the aforementioned services, several other log sources may be available on a network that can produce logs to assist during incident response and analysis. For example, networks utilising Dynamic Host Configuration Protocol (DHCP) servers that allocate client IP addresses with short lease times may require additional information to enable correlation between IP addresses and activity stored in other log files. The ability to quickly identify a compromised host using dynamic addressing when an organisation is informed weeks after a security event is vital and can save significant analysis resource. Additionally, the logging of client requests to local Domain Name Service (DNS) servers may assist in identifying malware attempting to communicate to known malicious domains, or may locate domain names that malware is attempting to contact that do not currently resolve to an IP address as they will not be included in proxy logs because the connections will not be successful. In all cases, an assessment of each data source's role within the organisation should be evaluated and, if appropriate, incorporated into the wider logging policy prior to an incident occurring.

4.2 Network Infrastructure

4.2.1 Authentication Servers

Authentication servers are used to authorise clients to join networks and access resources held on them. Although the implementations and the protocols used for authentication vary depending on the network environment, they typically log each user authentication attempt along with a set of metadata about the connection. Log data from authentication servers can be used during incident response to determine how an intruder is connecting to the network and to track their movements if they are propagating between hosts.

Control of an authentication server may represent an attractive target to an intruder as they can potentially leverage this to have unrestricted access to a number of resources on the network. In addition to the logging of authentication attempts to the server, it is important to ensure that host logging is activated to detect direct compromise of the authentication server. For more information on host logging, see section 4.3 of this document.

Recommended field	Usage
Time and date stamp	Used for time line creation, filtering and correlation with other log data. It is important to store the time zone offset of this field if it is available to enable correlation with other log sources.

Recommended field	Usage
Remote IP address	The IP address of the client requesting authentication can be useful to determine where the request is originating from and as a general indicator to query other data sources.
Hostname	The hostname of the machine requesting authentication can be used to locate and analyse compromised hosts on the physical network.
Username	Used to determine which user account may be compromised, or if the attacker has created additional accounts to use on the network.
Authentication status	Used to determine whether the user was able to successfully authenticate to the server. Brute force attacks attempts may be identified with repeated authentication failure messages.
Action	A message relating to the type of account event (such as user login or logout) can be used to determine the pattern of attacker activity over time.

Table 5: Recommended fields for authentication server logging

4.2.2 Firewalls

Firewalls can provide a rich source of data for network traffic that contravene company policies based upon the configuration of their rule sets. Most devices can also track the state of traffic flows passing through them in order to perform content inspection of the sessions to detect and alert on malicious activity. While firewalls are able to generate detailed logs, they often don't provide much context to each event and are therefore more suited to complement other sources of information. The value of firewall log records will vary depending on where it is deployed within a network and the type of attack activity undertaken; firewalls on the border of a network may be able to detect incoming attacks and data exfiltration events, but will have no visibility of lateral movement between hosts.

Recommended field	Usage
Time and date stamp	Used for time line creation, filtering and correlation with other log data. It is important to store the time zone offset of this field if it is available to enable correlation with other log sources.
Event type	Used to determine which rule triggered the log event for each traffic flow and how the firewall interacted with it. For example, it will show whether the activity was permitted or denied based on the running rule set.
Source and destination IP address	Used to determine the origin and destination of the logged traffic flow and to identify the target of an attack within the network.
Destination port	Provides an indication of the protocol being used, based upon common port assignments. For example, large numbers of requests for port 80 across a range of IP addresses may indicate scanning for web servers on the network.
Source Port	Storing the source port of a connection is not vital but is recommended to help identify sessions that have been logged by other devices by correlating them with both the client and server port along and associated session time stamp.

Table 6: Recommended fields for network security device logging

4.2.3 Network Intrusion Detection Systems

Network Intrusion Detection Systems (NIDS) can record detailed information on sessions that contain suspicious traffic indicative of an attack based upon installed signature detection routines. Logging rates for NIDS will vary depending on the type and number of signatures deployed on the system and therefore the effectiveness of detection capability can be reviewed over time by examining log records to determine if the alerts are relevant and actionable.

Recommended field	Usage
Time and date stamp	Used for time line creation, filtering and correlation with other log data. It is important to store the time zone offset of this field if it is available to enable correlation with other log sources.
Signature	Used to determine which detection rule triggered the log event. This log record must provide enough detail so that the analyst can understand why the network traffic triggered the event and what other indicators they should look for.
Source and destination IP address	Used to determine the origin and destination of the logged traffic flow and to identify the target of an attack within the network.
Source and destination port	Used to help identify sessions that have been logged by other devices by correlating the client and server port along with the associated session timestamp.

Table 7: Recommended fields for NIDS logging

4.2.4 Routers and Switches

Routers may be configured to permit or deny particular network traffic types and to log information about traffic that falls outside of these policies. In addition to this, many device models will log information regarding their system state and attempts to authenticate to their administrative interfaces. Although router logs are primarily used in combination with network security device logs to develop a wider picture of an intruder's activity on the network, access logs should also be reviewed for indications of unauthorised users attempting to gain administrative access to network infrastructure. Some routers and switches have the ability to log additional information about protocols that are in use on the network. This information may be used in an intrusion detection capacity or as part of a proactive approach to improving network security by detecting implementation issues. For example, monitoring Simple Network Management Protocol (SNMP) traffic logs may provide indications of configuration issues that may manifest themselves as security issues, or may detect an intruder attempting to use SNMP to perform reconnaissance of the network.

Recommended field	Usage
Time and date stamp	Used for time line creation, filtering and correlation with other log data. It is important to store the time zone offset of this field if it is available to enable correlation with other log sources.
Source and destination IP address	The source and destination IP address of the connection can be used as pivot points to other data sources.
Source and destination port numbers	Used to help identify sessions that have been logged by other devices by correlating the client and server port along with the associated session timestamp as well as giving an indication of the protocol in use (such as HTTP) based on common port assignments.
Protocol	Typically either TCP or UDP, the protocol can be used as supplementary data relating to the network traffic and can indicate the kind of malicious activity being observed. For example, high levels of UDP traffic to random ports on a single host might indicate that a UDP flood attack is being performed.
Access Group	Used to reference which Access Control List blocked traffic is violating.
Application	Some routers allow for more intelligent logging of certain common applications such as HTTP. This is useful as it allows more detail of the connection to be identified from this log without having to correlate with other sources.

Table 8: Recommended fields for router logging

To obtain a broader view of host activity on a network without relying on full packet capture, devices that support network flow export such as NetFlow[2] can be used to log a variety of information about traffic flows. This information can be used to determine evidence of intruder activity over ports or protocols that are not specifically logged by another device.

[2] http://www.cisco.com/c/en/us/products/ios-nx-os-software/ios-netflow/index.html

Recommended field	Usage
Time and date stamp	Used for time line creation, filtering and correlation with other log data. It is important to store the time zone offset of this field if it is available to enable correlation with other log sources.
Source and destination IP address	Used to determine the client and server side of each logged connection for intruder source and host identification purposes.
Source and destination port numbers	Used to help identify sessions that have been logged by other devices by correlating the client and server port along with the associated session timestamp as well as giving an indication of the protocol in use (such as HTTP) based on common port assignments.
Number of bytes and packets in flow	The amount of data in the flow is useful for statistical analysis and to determine possible exfiltration events based upon abnormal transfer volumes.

Table 9: Recommended fields for NetFlow logging

4.3 Host Infrastructure

Host-based logging can provide a rich source of data relating to intrusion activities, although routinely analysing the event records can be complex due to the potentially high volumes of data that can be generated. It is best used in conjunction with less verbose log sources that can be used to identify specific hosts for further analysis and to validate analytical conclusions. For example, if proxy logs show that a piece of malware was downloaded from a web server, host log sources can be used to verify whether it was actually run.

4.3.1 Program Execution Auditing

Most operating systems provide administrator-level logging of program execution, such as Audit Process Tracking within the Microsoft Windows environment. When enabled, the log records can allow analysts to track when applications start and terminate as the host is running. Other logging options can provide opportunities to detect malware persistence mechanisms as they can track services being started and scheduled jobs being registered, which are both techniques that malware can use to maintain persistence on a victim. These log sources are best used in conjunction with an

analyst's understanding of the corporate computing environment established on the network so that anomalous behaviour can be detected.

Recommended field	Usage
Time and date stamp	Used for time line creation, filtering and correlation with other log data. It is important to store the time zone offset of this field if it is available to enable correlation with other log sources.
Hostname	Used to correlate logged activity with the physical hosts on the network.
Username	Used to track the username the under which the process was executed, which can indicate which user on a host is affected and whether the process is running with administrative privileges.
Process name	The process name indicates the actual executable that was run, which is useful for providing initial investigation points during host forensics. Logging of the executable path is recommended to help discriminate between legitimate executables and malicious programs using the same process names. In most managed environments, a known list of services should be available, enabling comparisons against the process name value to determine suspicious service installation events.
Event action	Used to track what type of event was being logged e.g. did the process start, stop or restart.

Table 10: Recommended areas for program execution logging

4.3.2 File Access Auditing

The creation and analysis of log records of user access to sensitive files can provide an indication of potentially what data may have been obtained during a network intrusion event. Within the Microsoft Windows environment, this is achieved through Windows Object Access Auditing, which logs accesses to files in administrator-defined folders and network shares.

In addition to file access logging within the operating system itself, Document Management Systems may also provide log data to track which documents are accessed by which users.

Recommended field	Usage
Time and date stamp	Used for time line creation, filtering and correlation with other log data. It is important to store the time zone offset of this field if it is available to enable correlation with other log sources.
Hostname	Used to correlate logged activity with the physical host.
Username	Used to track the username which was used to access the file, allowing for correlation with user logon events and process execution logs to determine whether it was likely a user directly accessing the file or whether malware was used.
Process name	The process name can provide context as to how the file was accessed. For example, this may represent access through an application, via malware or through direct file system access.
File name	The file name is used to determine what data was accessed on the host.
Result	The result field can indicate the current user or processes access privileges and whether the data was successfully accessed.

Table 11: Recommended areas for file access logging

4.3.3 Host Security Software

Security products such as anti-virus software are often deployed to provide a local level of protection to the host. Log records generated by this software can be useful for identifying intruder tools, although their value will strongly depend on the quality of signature sets and heuristic rules available to the application and whether the malware has been specifically designed to evade detection. Even if an attacker's malware is not detected by anti-virus applications, additional tools such as password dumpers may be identified through heuristic detection routines. In addition to the analysis of anti-virus

detection logs, log files of the software itself can be useful as attackers may attempt to disable scanning routines, leaving evidence as log events.

In addition to the analysis of anti-virus detection logs, reviewing the log files of the software itself can be useful as attackers may attempt to disable scanning routines, leaving evidence as log events. In order to ensure the integrity of these files, all anti-virus log records should be stored on a centralised log server wherever possible.

Recommended field	Usage
Time and date stamp	Used for time line creation, filtering and correlation with other log data. It is important to store the time zone offset of this field if it is available to enable correlation with other log sources.
Hostname	Used to correlate logged activity with the physical host.
Username	Used to track the user account in use when the anti-virus event was logged.
Signature	The signature name associated with the detected threat can be used to obtain information about the behaviour of the malware and how it may be remediated.
File name	The file name that triggered the alert can be used to obtain a sample of the virus from the host for further analysis.
Action	Useful for determining whether the anti-virus software had automatically taken action against the detected threat, including quarantining or deleting it.

Table 12: Recommended fields for host security software logging

The figure included below provides an example of a plain-text anti-virus log entry including information on the infected file, the name of the detected malware and the username associated with the scanning event. In this case, an on-access scan was executed by the operating system itself, with the process running as the NT AUTHORITY\SYSTEM user.

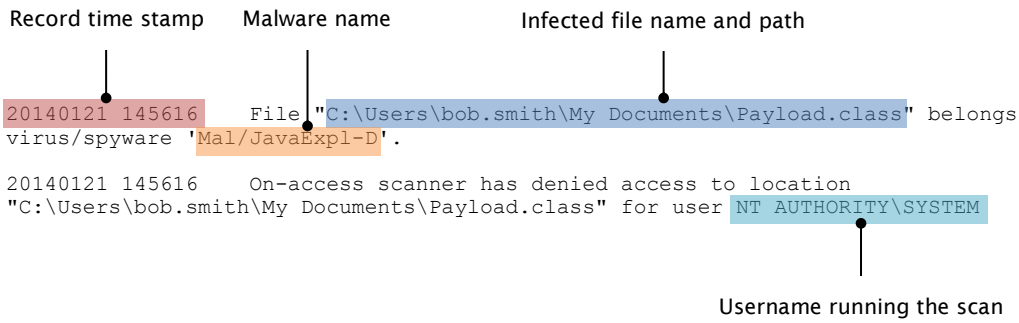


Figure 5: Anti-virus log entry example

4.4 Remote Connections

Although remote connection facilities are provided through network servers that may be monitored through other mechanisms, they represent an access mechanism that may exist outside of normal network policies (such as “Bring Your Own Device”) and are therefore worth considering for specific attention when defining a log management strategy.

4.4.1 Virtual Private Networks

Virtual Private Networks (VPNs) provide a mechanism to remotely connect devices to a network and use associated resources as if they were connected to the corporate environment. While VPN technology is advantageous for connecting disparate computing resources and enabling remote working, the compromise of VPN access credentials can represent significant risk to the security of the network as a remote attacker can interact with any host on the network that the user account has access to.

VPN connection logging should primarily include details of user connections and should be as comprehensive as possible to reduce the amount of time required to correlate this data with other log sources in response to an intrusion.

VPN logs can provide a good data set for behavioural analysis purposes and can be used to identify sessions that are markedly different from typical user interactions with the network. For example, out of hours login events for users who do not work during those hours, logins from valid user accounts that do not normally use VPN connections, and logins from geographic regions far from a user’s typical working location may all be indicative of unauthorised access to the network.

Recommended field	Usage
Time and date stamp	Used for time line creation, filtering and correlation with other log data. It is important to store the time zone offset of this field if it is available to enable correlation with other log sources.

Recommended field	Usage
Session ID	The VPN session ID can be used to associate activity between other log sources from that particular VPN session. This field is particularly useful when security appliances automatically map the session ID over multiple logs.
Username	The username used for VPN authentication is essential for identifying whether legitimate user account has been compromised or a new account has been added by the intruder, and to determine what access they might have to other parts of the network by correlating the sign-on event with other data sources during incident response. It can also be used to baseline client activity on a per-user basis to identify anomalous connection events.
Client IP Address	The IP address of the VPN client can be used with the username field to associate anomalous connections in conjunction with other data sources.
Action	The action associated with the event (such as login and logout) can be used with the session time to establish a time line of user activity.
Session time	If the session is successfully authenticated, a log record of the connection time can be used to establish a time line of user activity which can then be used to correlate with other log sources.

Table 13: Recommended fields for VPN access logging

4.4.2 Remote Desktop Services

Remote desktop services allow users to connect remotely to machines within a network, either by establishing a new session on the host or taking control of an existing one depending on the software used. While the remote host is not logically connected to the network as is the case with VPN connections, remote desktop services still provide many of the capabilities available to a local user such as access to network shares and applications installed on the machine. They also allow data to be sent between the remote host and the client through interactions between clipboards (copy and paste) and, depending on the software and configuration, “drag and drop” file transfer.

Remote desktop connections are often favoured by intruders as a mechanism for controlling a compromised host after valid user credentials have been obtained by malware or key loggers. Once connected via a remote desktop connection, attackers have the opportunity to use the graphical user interface of the host to install new tools, disable host-based security software and perform further reconnaissance of the network. Analysis of log records pertaining to remote desktop connections can indicate which hosts within a network are controlled by an attacker based off known information about an intrusion, or can be used in a proactive manner to detect abnormal activity such as access to hosts outside of business hours.

Recommended field	Usage
Time and date stamp	Used for time line creation, filtering and correlation with other log data. It is important to store the time zone offset of this field if it is available to enable correlation with other log sources.
Username	Used for identifying whether a legitimate user account has been compromised or if a new account has been added by the intruder. This information can then be used for correlation purposes with other log data sources to determine the user's activity on the network. It can also be used to baseline client activity on a per-user basis to identify anomalous connection events.
Client IP address	The IP address of the remote client can be used with the username field to associate anomalous connections in conjunction with other data sources.
Client hostname	Although it is unlikely that the hostname of the remote client will be represented in other sources of data, it can provide a useful information source for determining whether the connection has been made from another machine within the network.
Action	The action associated with the event (such as session connection and disconnection, screen saver unlock and the origin of the connection) can be used with the session time to establish a time line of user activity.

Table 14: Recommended fields for remote desktop access logging

5 Analysis of Log Data

One of the most important aspects of any log management strategy is to employ an effective mechanism for the analysis of the generated log data. Although the methods and workflows for analysis will vary between organisations and the threats they face, it is critical that an ongoing process of review is put in place to ensure that the best value is obtained from investment in creating and storing log records.

5.1 Mapping Log Data Sources to the Intrusion Kill Chain

The Kill Chain is a conceptual model used to define the stages through which an adversary has to progress to achieve an objective. This model has been extended into the context of information security by Hutchins, Cloppert and Amin[3] in 2010 to describe the "Intrusion Kill Chain", which discusses how an iterative process of analysis and defence can be used to prevent persistent adversaries from gaining access to networks. In the case of computer intrusions, this will reflect how an attacker locates a target, prepares a method for exploiting the network and establishing a presence in order to have control over one or more hosts.

When considering how log sources should be collected and analysed, the stages of the Intrusion Kill Chain can be a useful model for determining how to maximise the opportunities to detect an intruder through proactive analysis of a range of data sources. The principle of Kill Chain analysis involves the identification and remediation of attacks at one stage of the chain and using information about how the intruder conducts their operations to identify additional characteristics for detection at earlier stages of future attacks. Although this iterative process demonstrates how persistent actors can be detected over time, the speed and effectiveness of incident response can be improved by properly establishing an effective log management strategy from the outset.

Figure 1Figure 6 provides an example mapping of log sources to stages of the Kill Chain to demonstrate how a layered log management strategy can be used to detect a range of attacker activities.

[3] http://papers.rohanamin.com/?p=15

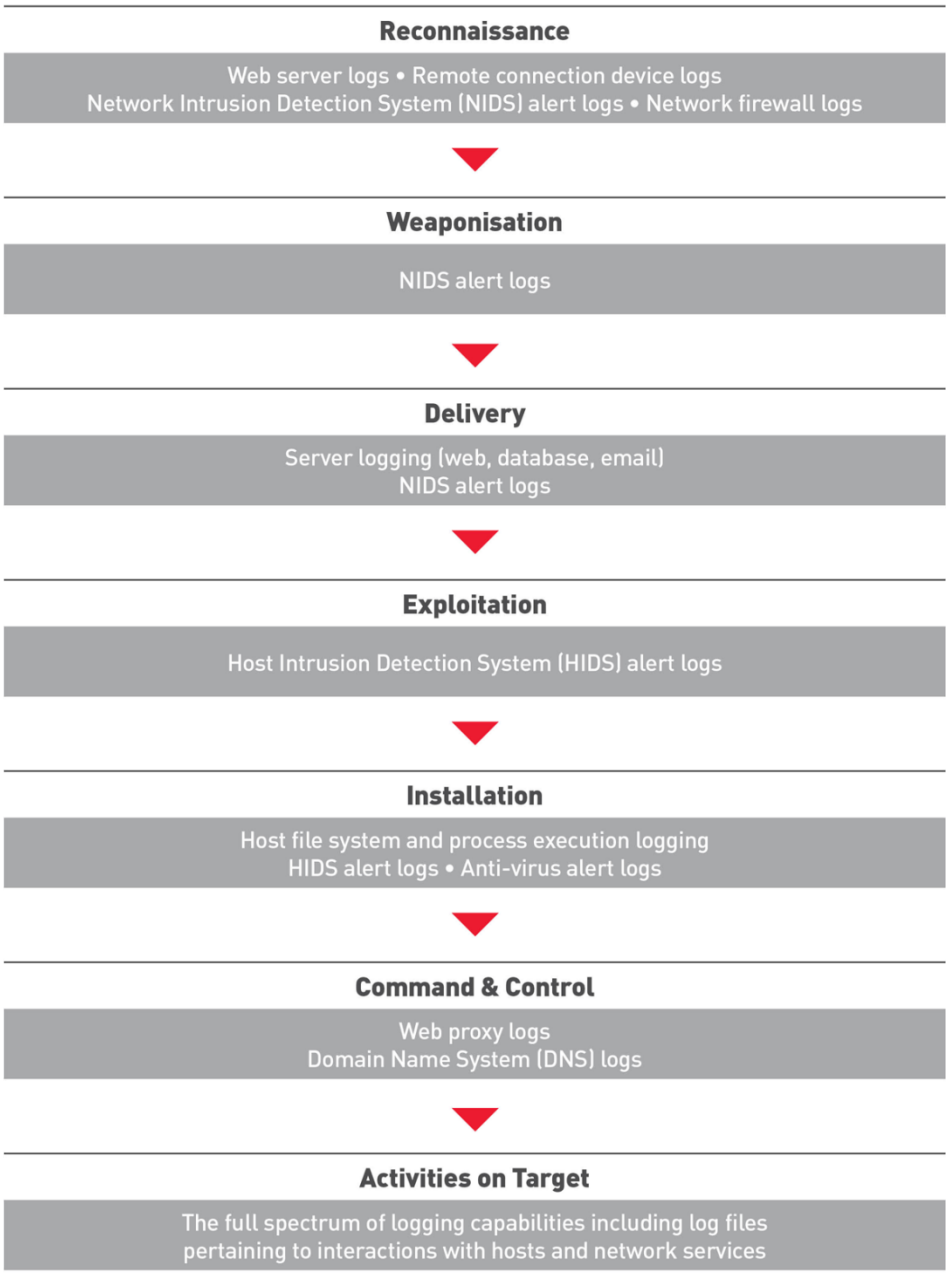


Figure 6: Example mapping of log data sources to phases of the Intrusion Kill Chain

5.2 Responding to Incidents

In broad terms, the role of log analysis during incident response involves the extraction and retrospective analysis of log data in response to a security incident, employing post-event filtering to the data set by using indicators of compromise to develop a picture of an intruder's activity. This understanding can then be used to remediate the attack by identifying and cleaning the affected systems, determining and eliminating the intrusion vector used by the attacker and ensuring that no backdoors have been

installed to maintain access to the network. As part of this work, log analysis can be used to gain an understanding of the impact of the attack, including clarifying the origin and quantity of any data taken from the network.

The log analysis process will vary with the nature and extent of the attack, involving the correlation of data from multiple sources using common fields. A company will frequently become aware of an incident through the detection of malicious activity by security software, a notification from an external organisation (private sector or a government department) with wider knowledge of an attack campaign, or through proactive anomaly detection conducted by internal security teams. When this occurs, one or more indicators of an attack, such as an email address, domain name or IP address will be provided to incident handlers that can be used as a 'seed' for further queries to pivot through other data sources to map attacker activity. For example, an anti-virus alert on a host machine may reveal an initial infection that can be tracked back to an email received by a user. Information about this email such as the sender address and subject line can then be used to search mail server logs to locate other users on the network who might be targeted by the same attack.

The role of the affected user or machine may influence the response. For example, an infection on a typical user's workstation may be treated differently to the same infection on a board member's or system administrator's machine, or if the infection is detected on key infrastructure such as a server running a process control system. This approach is entirely appropriate and should be informed by risk management decisions within your organisation.

In some situations, the scope and complexity of a security incident may warrant more in-depth analysis of log records that is beyond the resources of in-house network security teams. If this occurs, the expertise of an external incident response company may be required to assist in the handling of an intrusion event, during which time it is advantageous to have readily available and well organised log stores so that data can be quickly passed to incident handlers to enable rapid analysis and response. This task is not straightforward for incident response companies either; it is worth confirming that your incident response company of choice has the ability and expertise to handle and analyse the logs on your behalf.

5.3 Proactive Analysis for Intrusion Detection

Although log management is most commonly associated with incident response procedures where log files are queried retrospectively based upon one or more indicators of compromise, where possible log data should be used as part of an ongoing process of review and analysis to identify attacks as they are in progress. This is achieved by having a solid understanding of typical traffic patterns on your network so that anomalous events can be more readily identified and investigated. The most effective way to begin this process is to dedicate regular time, perhaps even a full time resource, to routine log review after which a baseline of activity can be established and additional log sources can be introduced to provide a richer understanding of network activity. It is critical that a deep understanding of the role of users and machines on a network is developed over time to enable this style of analysis.

The three primary stages of log analysis for intrusion detection are:

- Filtering of legitimate traffic
- Detection of known malicious traffic
- Identification of anomalous traffic

To reduce the potentially enormous volume of available data to a level which can be analysed effectively, some degree of filtering is required. All "known legitimate" traffic can be filtered out straight away and further filters will evolve over time based upon feedback from the analytic process. "Whitelisting" certain traffic interactions will reduce the dataset further. For example, web requests to domain names highly ranked in the Alexa Top Sites[4] list are unlikely to be representative of malware tasking requests (unless a major web service is compromised) and therefore can usually be discounted from initial analysis. Note that this process simply filters data from queries rather than removes the data entirely; findings of further analysis may require whole data sets to be re-queried to fully understand the extent of an attack.

Detecting known malicious traffic does not need to be problematic; data feeds (both paid and free) are available to incorporate into intrusion detection workflows for the purpose of detecting "known malicious" behaviour. These can include indicators of compromise such as domain blacklists, file hash databases and full threat intelligence feeds containing end-to-end descriptions of specific attack group's modus operandi, tools and known infrastructure. Although these feeds may not provide information on customised or emerging threats, they can provide an efficient way of detecting prolific intruder activity, or activity which represents a high threat to a particular sector.

After known legitimate and known malicious filters are applied to log data sets, analytical techniques can be applied to the remaining data to identify anomalies and emerging attacks. The techniques used to determine anomalous activity will vary depending on the network environment and will develop over time as security personnel become familiar with the data and how intruders attack their network. However, some typical techniques for anomaly detection are presented below as an example of how log data can be used in this way.

5.3.1 Detection of Unusual Network Traffic Requests

Outbound requests from a network can be indicative of beaconing or data exfiltration events, where malware on a host contacts an external web server for command and control. When constructing these requests, malware authors will often encode information pertaining to the infected host into a URL string of an HTTP GET request such as the hostname of the client or some other unique identifier, as demonstrated in Figure 7 below.

```
GET /VzJoPTY0/VTpYPAY1a0s12323/13152/27VzMMPzBjaA/ZTRRDDUwYUxpYW8292/
```

Figure 7: Example of an HTTP GET request as generated by the Backdoor:Win32/Comfoo malware

[4] http://www.alexa.com/topsites

In some cases, these requests are extremely distinctive and may be revealed through statistical analysis of web proxy logs, especially if a single host is periodically but repeatedly requesting a specific resource that is unique from any other machine on the network. It should be noted that care must be taken when investigating possibly suspicious requests as many legitimate services also use frequent and unique requests to support dynamic web services such as user messaging; some of these requests can be filtered by combining proxy logs with known whitelists as discussed previously.

Most applications that access web resources specify a "user agent" string within their requests as recommended in RFC 2616 to identify themselves to remote servers. Although the RFC specifies the general format of these strings, there is no enforced standard and therefore they vary immensely. While many user agents accurately describe a web browser with associated metadata about the operating system running on the host, some malware authors specify custom user agents that appear abnormal when compared to those commonly in use across the network. Figure 8 depicts the user agent used by a variant of the Win32.Agent.byhm malware [A] compared to that of a typical web browser [B].

```
[A] User-Agent: EMSCBVDFRT
[B] User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:26.0) Gecko/20100101 Firefox/26.0
```

Figure 8: Comparison of Win32.Agent.byhm malware and Firefox web browser user agent strings

It should be noted that full user agent strings will vary greatly even for legitimate software and is dependent on how the application is configured and the patch level of the host it is running on. Therefore, it is critical that an understanding of common user agents within your network environment is established, so that unusual variants can be detected during log analysis. Creating baselines of typical user agents in use on your network is beyond the scope of this document, although documentation of several techniques for capturing and filtering this data is freely available[5].

5.3.2 Temporal Analysis of Network Events

If it is possible to develop an understanding of the typical user interactions within your company's network, temporal analysis of network events can be a powerful tool for determining abnormal behaviour that may be indicative of compromise by an attacker located in a different time zone. For example, if a particular user works remotely but always logs on to a company VPN at 9am every morning and logs off at 5pm, their access to corporate resources at 1am might be suspicious and worthy of further investigation. Similarly, it is likely that most user administration activity is likely to be conducted during office hours. If log files show a privileged user account being created early in the morning, it could indicate that administrator credentials have been

[5] https://www.sans.org/reading-room/whitepapers/hackers/user-agent-field-analyzing-detecting-abnormal-malicious-organization-33874

compromised and an intruder is creating a backdoor account to maintain network access.

The concept of “impossible travel” can also be used to detect if otherwise valid user credentials have been co-opted by an intruder. For example, each client IP address of remote connections per user account can be stored and geo-located using online tools to determine the approximate location of the user. If multiple login events occur within a short period of time from geographically disparate locations, their origins should be reviewed to determine if each authentication event is legitimate. If a particular user logs in to the network from an IP address located in the UK at 9am and a subsequent connection is observed at 10am from a US IP address, it is reasonable to assume that the individual could not have conceivably travelled between those points in one hour. Although there are technological reasons while this may occur such as the use of VPN technology with gateways in other countries, your company policy should dictate whether access of this type should be allowed and whether the activity merits further analysis. A short phone call to the user in this case could clarify where they are and whether the activity is legitimate.

While temporal and geographical analysis can enable anomaly detection, it may be impossible to achieve without prior and ongoing knowledge of how and where all users will authenticate with the network. If they are spread out over many countries, analysing log data using the correct time zone offsets will be vital for this approach.

5.3.3 Base-lining of Network Traffic Volumes

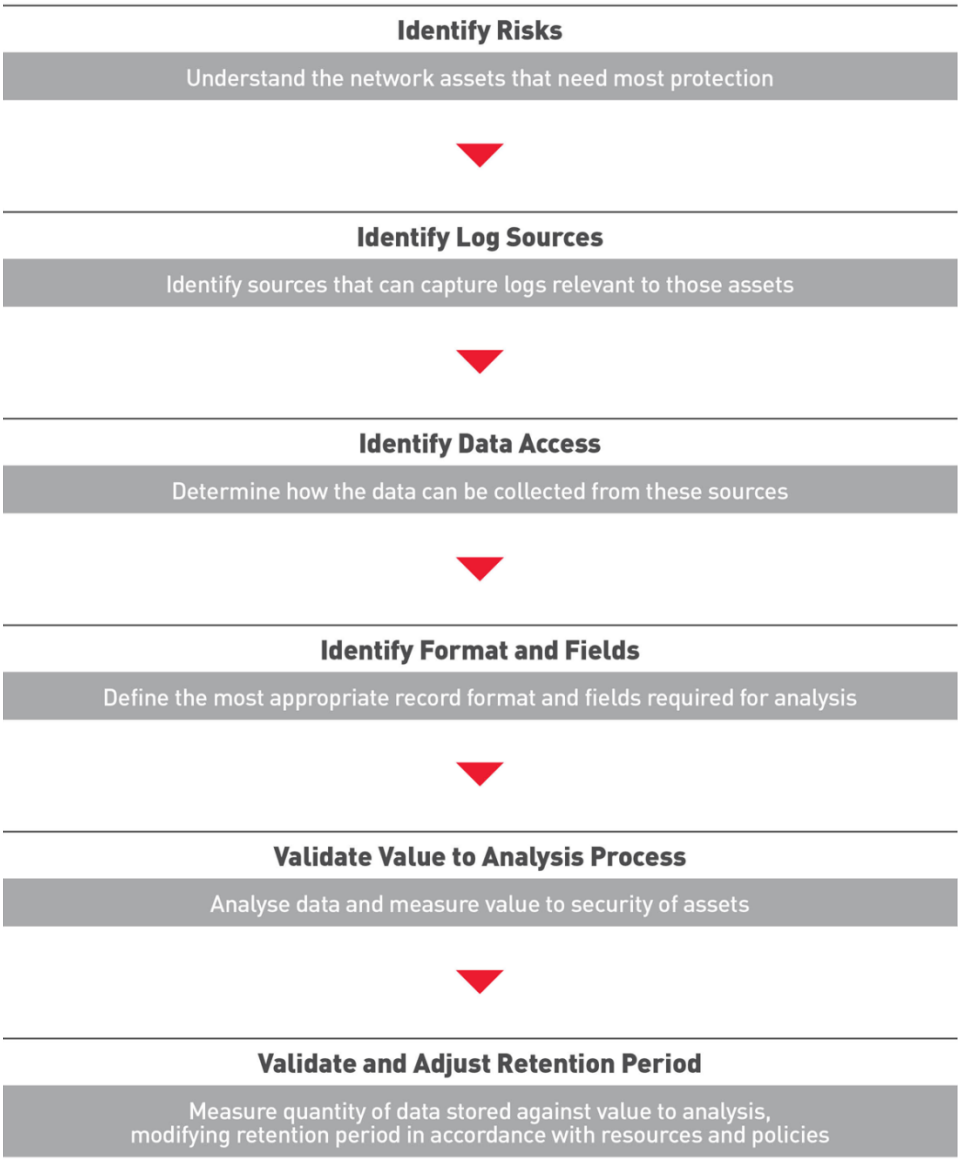
Analysis of log data pertaining to network traffic levels over time (such as NetFlow) may establish a general baseline of activity on a per host basis and could be used to identify abnormal behaviour. For example, outbound transfer data volumes far exceeding expected levels from a single host might indicate that the machine has been compromised and data exfiltration is occurring.

5.3.4 Detection of Anomalous Process Execution

If standardised master installation images are used for each client workstation on a network in conjunction with restrictive user policies on what applications can be installed, it may be possible to develop a baseline understanding of normal process execution and identify when anomalous events occur. If process execution auditing is enabled, a log entry for the process name and folder path can be generated each time a program is run, which can then be compared to known good entries to identify abnormalities. While this approach will regularly generate false positives even after a suitable baseline has been established, especially when new software is deployed or system patching occurs, process execution events from unknown software occurring on small numbers of hosts on the network are worthy of further investigation.

6 Log Management

An effective log management policy is required to store and enable analysis of recommended data sources. While there are ideal scenarios for such policies, the reality is that they must be tailored around the resources available to your organisation, the configuration of the network and hardware in your environment, the requirements of your security plan and any legal or policy constraints you must work under.



6.1 Log Storage

Once the evaluation of each available log source has been completed, the appropriate location to store each type of log record must be determined. There are three main log location strategies that can be employed: local storage on each device, centralised storage on a log server or a combination of the two, depending on the required detail and accessibility of log records.

Local logging distributes storage requirements across the enterprise as each device is responsible for supplying its own data store and therefore no separate database provision is required. While this policy also minimises the network bandwidth resources required as data does not need to be routinely transmitted to a centralised store, it may be complex to interrogate each log source and correlate it with other records during incident response. Other disadvantages of this policy include reduced data retention periods as the space allocated for logging on each device must usually be smaller than could be provided by a log server, and the security implications of storing log files on the same devices on which the intruders may have administrator-level access.

Centralised logging can take the form of a networked file store that various agents can write their logs to, or a purpose-built solution in the form of a Security Information and Event Management (SIEM) tool, which is discussed further in section 6.3. Although some form of centralised logging is generally recommended as it ensures that larger amounts of log data can be pre-collected and correlated to enable real-time analysis for both intrusion detection and response purposes, centralised logging comes with a greater up-front provisioning requirement for primary storage and log backups. Storing data centrally increases the potential for longer data retention periods that can scale with increases in network size as additional storage is more easily added to a single log server. However, increased log retention also comes with the disadvantage that any processing and analysis framework used to view the data must intelligently store and filter records to minimise the impact on analyst query times.

A combination of the two approaches may be considered to overcome many of these disadvantages at the expense of increasing overall administration overhead. In this case, the most important log sources and data fields would be automatically forwarded to a centralised location for ingestion and real-time analysis while more in-depth data is stored for shorter periods on individual devices and interrogated on-demand when an incident is detected. For example, all server (proxy, web and mail) and infrastructure (routers, IDS and anti-virus) traffic may be forwarded to a centralised location for intrusion detection, while more verbose records about process execution and file access on individual hosts are stored locally and only processed when required.

Once a log storage strategy has been established, the data retention policy for each source must be determined. As available log sources and logging requirements will vary from organisation to organisation, firm recommendations for storage requirements are hard to provide. The table below provides some general indications of effective log retention periods for classes of device based on the verbosity of their logging and effectiveness for incident response purposes, although the size of some networks may preclude storing log records for these periods of time. During the development of a log management strategy, an assessment of the current and future storage requirements for each data source should be undertaken to determine realistic retention times for your organisation.

Log Type	Retention	Notes
Proxy, web, authentication, remote access, firewall	2 years+	Longer retention periods are required for these high value data sources that can be used to identify the impact of new and existing persistent threats over typical attack timescales. For very large organisations this retention period may represent many Terabytes of storage and may not be practical.
Email, IDS, anti-virus, database, network infrastructure	6 months – 1 year	While valuable, these data sources are often coupled with active alerting facilities that can notify security teams of incidents as they are detected. Other data sources such as network infrastructure will change rapidly and are more valuable for general telemetry purposes.
Host process execution and file access	4 weeks – 6 months	Depending on usage, host logging can generate large numbers of events that are mainly used for verification after an intrusion has been discovered. As such, lower retention times are recommended.

Table 15: Suggested log retention periods by log source

6.2 Log Security

However your organisation chooses to store log files, it is important to ensure that they are suitably protected from manipulation or destruction by a malicious party intent on covering the tracks of their intrusion. While early systems for logging events often did not consider intentional or inadvertent modification of log records as a primary consideration in their design, additional security measures are now appropriate as log stores present an attractive target for attackers. Although the security capabilities vary between log storage solutions, the following general recommendations can be made:

- **Ensure the integrity of the processes generating log records.** Unauthorised users should not be permitted to modify or interrupt the processes that are used to create log files on devices, whether these are in-built commands or 3rd party logging agents. For example, users and local administrative accounts should not be able to turn off anti-virus software logging.

- **Limit access to local log files.** If log data is stored on workstations, users should not be given access to log files unless it is to append to existing records.

If possible, no read, rename or delete privileges should be granted on log files.

- **Implement security on centralised log files.** Centralised log stores represent an attractive target for attackers wishing to modify log data across a range of devices. For log data that is sent to a centralised location, security measures such as digital signing or encryption should be explored to ensure the integrity of this data, although for large volumes this may have unacceptable efficiency or administrative overheads.

- **Secure log data transmission.** It is recommended that any data transmitted between log sources and a centralised log store should be secured to prevent unauthorised modifications. While some devices and log agents support encrypted communication channels natively, other log sources that use plain-text protocols can be protected through the use of additional layers of encryption such as Internet Protocol Security (IPsec) tunnels. This may also be impractical for many organisations but is worth considering when high-value log sources have to transmit their data to centralised locations.

6.3 Security Information and Event Management software

Security Information and Event Management (SIEM) software offers the capability to aggregate, store and display log data in a way that can be used to enable near real-time analysis of the security of a network from the perspective of each of the log sources. If implemented and resourced appropriately, SIEM technology can help solve some of the more complex problems facing an organisation looking to improve the security of their network.

SIEM deployments generally consist of one or more database and log analysis servers and can acquire data either through agent-based or agentless mechanisms. Some data sources, particularly those relating to networked hardware, possess the ability to push their log data directly to a remote source. In this situation, agentless log management can be used by configuring the device to transmit records to the SIEM. In other cases, the SIEM may be required to authenticate with each remote device and request log records on a regular basis. While agentless log acquisition can reduce the complexity of installing additional software on the network, filtering is generally undertaken on the server for both push and pull mechanisms, potentially resulting in large transfers of data across the network and increased amounts of time to analyse events within the SIEM.

Agent-based log acquisition often employs a level of filtering and normalisation on the host, reducing data overheads transmitted to the SIEM albeit at an increased administration overhead created through the deployment, update and configuration requirements for each agent. Although agent deployments can provide support for log formats not natively processed by the SIEM, several agents may need to be deployed on each host to capture the required number of data points.

The selection of agent-based or agentless acquisition process will strongly depend on the type of SIEM deployed and the devices and topology of an organisation's network. However, in all circumstances it is important to ensure that the integrity of the log files

is ensured before transmission to the SIEM and that communications between host and server are authenticated so that false information cannot be injected into the analysis process by an intruder.

Most SIEMs work within Wide or Local Area Networks to aggregate data from the assets owned by an organisation. However, the increasing popularity of cloud services means that fundamental parts of a company's operations may be owned and run by a 3rd party. To complement this, some SIEM providers offer variants of their product that can poll systems run by other organisations using encrypted channels across the public internet. Further extending this concept, some SIEMs are offered wholly as a remote service in themselves, eliminating the requirement for locally managed storage solutions.

While available products differ in the overall capabilities they offer, common functionality includes:

- **Log record aggregation.** SIEM software provides a centralised location to aggregate logs from multiple data sources. This is achieved by configuring each data source to send their log data directly to the SIEM, where it can be parsed and ingested for analysis.

- **Data retention.** Centralised storage provides the possibility for long-term retention of log data beyond limits that may be in place if logging to the source itself. A SIEM is typically run on a hardware platform with considerable amounts of storage that can be used to retain log records for extended periods of time, maximising the chance that relevant data will be available to describe the full extent of an attack after it has been discovered.

- **Event correlation.** SIEM software often includes the capability to integrate multiple log data sources and correlate events between them. This is particularly useful when 'pivoting' between logs from different devices; for example, a SIEM may be able to automatically correlate outbound web requests to physical machines and logged on users using a combination of proxy, DHCP and authentication server logs. To enable effective correlation of data, it is essential that properly configured and synchronised time sources are used to store time stamps for each log record to act as a common temporal point of reference for each source.

- **Visualisation dashboards**. Many SIEM tools offer a range of customisable visualisation dashboards that can provide a high-level overview of the condition of the network and can highlight abnormalities in the data. Dashboards often allow analysts to directly query and manipulate the data to 'pivot' through multiple data sources and perform queries on large quantities of data without having to build separate parsers for each log source. Dashboards can also be configured to generate alerts, where anomalies in baseline data sets or the receipt of error messages from devices can be automatically forwarded to a security team.

Although SIEM technology offers a number of benefits with regards to the automation of log storage and processing while limiting opportunities for intruders to modify

stored information to cover their tracks, they should not be viewed as a complete solution for automatic detection of malicious activity. Although some vendors offer built-in signature sets to detect known-bad or anomalous behaviour on a network, it is vital that the routine use of any deployed SIEM solution is combined into overall security workflows as opposed to being used reactively after an incident has occurred. A strategy of this type will provide opportunities to detect sophisticated attacks that would not be highlighted by signature-based detection and to limit the effectiveness of intrusions as they are occurring, although it requires analyst experience in intrusion detection and an ongoing resource commitment towards monitoring the data collected by each system.

An increasing issue relating to the deployment and use of SIEM software is the increase of log data "noise" as a result of the number and complexity of systems feeding into the centralised store. As more and more security and monitoring tools are brought online, a SIEM tool has to cope with increased correlation complexities in order to display the information most relevant to analysts. Beyond this, the analysts themselves will have to spend additional time reviewing the presented data to ensure that they are identifying the relevant information during intrusion detection. This underscores the importance of establishing SIEM monitoring as part of an ongoing workflow rather than relying on the product to automatically detect malicious activity. By constantly reviewing and manipulating log records, analysts will gain more familiarity with the data being stored in the SIEM and will be able to continually validate the appropriateness of each source for intrusion detection and response purposes.

7 Case Studies

In order to illustrate how differing log management strategies can influence an incident response investigation, this section introduces three real-world case studies discussing some investigations undertaken by Context, including how particular log sources were used and where log creation policies influenced the understanding of how the intruder interacted with the network.

7.1 Case Study: Limited Logging over the Range of Network Assets

Headline: Investigation was complicated by limited log availability and the complexity of working with a third party hosting company.

7.1.1 Scenario

Company X was alerted to an incident after signatures in the IDS of the third party hosting provider generated alerts, suggesting that a possible SQL injection attack was launched against a vulnerable web application. Analysis took place using logs from three Microsoft IIS webservers, along with supporting information in the form of database transaction logs and IDS alerts. Almost a quarter of a billion IIS log events were analysed in total.

With only log data available from the company, the investigation focused on four areas:

- The initial compromise vector
- Any actions the attacker may have carried out while on the network
- Detection of any potential data loss
- If actions taken by the third party host and Company X successfully removed attacker access

Company X also provided IIS logs from overseas infrastructure in order to determine whether the attack was limited to UK systems.

The initial compromise was identified as an SQL injection attack that provided shell access to Company X infrastructure in the third party host. The attackers used a Metasploit module to exploit the SQL server and create a privileged account on the server. Once the attackers had gained access to the network it was very difficult to comprehensively understand what they had done; there was evidence of database enumeration with the aim of identifying user credentials, but when attacker activity appeared to stop it was not clear if this was because they had migrated to a secondary network access channel not covered by the web logs.

Limitations of the log data available prevented investigators from establishing beyond doubt where data had been exfiltrated. Although it was concluded that this situation was unlikely based upon the data provided, further forensic analysis of the hosts involved would have been required to confirm this was the case. The log records gave an indication that the attackers no longer had a presence on the company network after remediation had taken place, although it could not provide full assurance that the activity was completely mitigated.

7.1.2 Log Management Strengths and Weaknesses

Strengths:

- Long log retention times assisted in measuring duration of activity
- Web server logs contained application responses to SQL queries
- Web server logs contained details of SSL requests that wouldn't be exposed with network capture or proxy logging

Weaknesses:

- No firewall logs available for the network where the servers were located
- Incident response investigation process including log acquisition was hampered by lack of established processes with third party providers and time zone differences
- Inappropriate logging formats for various SQL server interactions complicated analysis
- Logs were not routinely reviewed – both SQL exploitation activity and account creation may have been revealed through analysis or SIEM review while the incident was occurring
- Proactive log review of security tools may have revealed that the IDS was only monitoring traffic on port 80 and was missing HTTPS requests. Although SSL logging was enabled on the web server, expansion of the IDS monitoring role may have assisted the company's overall security posture
- The attackers created an account with a name which could have easily been identified by the hosting systems administrators as being inconsistent with those belonging to Company X if the domain controller logs had been routinely reviewed

7.2 Case Study: Incomplete Data Logging

Headline: Log capture facilitated a thorough investigation, although incomplete logging reduced the speed of analysis and the full impact of the incident could have been avoidable though proactive log analysis.

7.2.1 Scenario

Two users in Company Y triggered an internal incident response investigation when they reported that they were locked out of their accounts by remote sessions. Examination of login events identified several other machines exhibiting the same issue and forensic images of these hosts were taken for offline analysis.

Three sets of logs gave investigators the evidence they required to understand the compromise: proxy logs, anti-virus logs and NetFlow records. It was quickly apparent that the attackers had identified a machine that had been intentionally exposed to the

internet from the border of the network which offered remote access services to users who had a valid account. The attacker's modus operandi was to obtain valid user credentials and log in to the server, but how these credentials had been acquired was never fully established.

Analysis of log and host forensics data identified that the attacker had probed the network, searched for additional hosts of interest and made connections to the remote server outside of UK office hours using un-attributable connections. Once initial access was obtained on Company Y's network, they used Remote Desktop Protocol (RDP) to log directly into additional machines, allowing them to remotely access a machine and its resources as if sitting in front of it. Over a period of six days the attackers were able to probe additional machines on the internal network and download tools that enabled them to steal additional user credentials. The attacker also attempted to install tools that would allow them backdoor access to the system at a later time. Due to the nature of the attacker's modus operandi, Company Y was able to quickly limit their impact by restricting access to the server which the attackers had been traversing through to gain their foothold.

While all of the hosts involved in the incident were identified during the response phase, it was not possible to establish with a high degree of confidence exactly what data had been stolen as the log files generated by the company did not include key data fields relevant to the investigation. However, analysis using other data sources allowed the quantity of data exfiltrated to be estimated, with approximately 400Mb of data sent from the network by the attacker. Investigators determined that the attackers also visited webmail providers while they active on the network, although this traffic was transmitted over SSL and therefore the exact nature of the requests were unknown. The number and size of sessions were useful to determine possible additional channels of data exfiltration or tool transport.

The actions the Company Y took in relation to limiting access to the server removed the attacker's ability to communicate with the network. Subsequent examination of the machines involved showed no signs of backdoors left by the attackers, thus denying continued access to the network as part of this attack. The follow-on remediation events initiated by Company Y included resetting passwords and work continues on additional hardening of the network.

7.2.2 Log Management Strengths and Weaknesses

Strengths:

- Account logging was comprehensive and allowed for an initial timeline to be quickly built up
- Proxy logs included the name of the domain account being used, allowing direct correlation with network access logs
- Anti-virus logs allowed analysts to determine whether requests for attacker tools were successfully installed, indicating the extent of the compromise

Weaknesses:

- Proxy servers only logged HTTP and HTTPS traffic, hampering analysis of lateral movement between network segments using other protocols
- No logging of RDP connections meant that analysis relied on identifying outbound web connections made by the attacker to determine when they were active on a particular host
- No logging was enabled on the firewalls, which would have logged RDP traffic traversing network zones to contribute to a wider picture of the compromise
- Host log data was not centralised, requiring separate host forensics work to be conducted to evaluate extent of compromise
- Proactive analysis of logs was not conducted; established workflows may have alerted the company to unauthorised accounts logging on to particular network assets or identified anti-virus alerts being generated on detection of password dumping tools
- A pre-incident evalutation of exactly what was being logged would have highlighted immediate and easy improvements to data coverage which would have greatly enhanced future investigations

7.3 Case Study: Comprehensive Logging

Headline: Comprehensive logging enabled a detailed picture of the incident and a high level of post-investigation assurance that the incident was contained and remediated.

7.3.1 Scenario

Company Z received a victim notification issued through the Cyber Incident Response scheme[6] informing them that a legitimate third-party website had been compromised and was serving malware (via a “watering hole attack”) from a suspected state-sponsored actor. There were indications that hosts on the network had been compromised after visiting the site.

Company Z had a dedicated incident response and investigative capability and as such were able to stand up an internal team of significant size to conduct the investigation, with external experts providing oversight and direction.

During the investigation analysis of significant volumes (approximately 5 terabytes of log files compromising around nine billion records) of log data took place, primarily consisting of web proxy logs and host data logs. While the watering hole attack was found to have been successful and some malware was found to be active on the network, further investigation via log analysis revealed the attacker had not actively developed this foothold any further and there was no evidence of attacker activity elsewhere on the network. If the deployed malware had been activated by the attacker, the log store would have proved even more valuable as commands sent to the particular malware used in this attack could be decoded from data stored within proxy

[6] http://www.cpni.gov.uk/advice/cyber/cir/

logs, allowing investigators to understand the attacker's activities more thoroughly.

As a result of the initial notification, the organisation was able to rapidly identify seven compromised hosts. Forensic analysis of the machines and reverse engineering of the identified malware gave a thorough understanding of the initial infection, provided further development of indicators such as IP addresses and traffic signatures, but not whether it had spread beyond those seven machines. Comprehensive log analysis also resulted in an additional benefit: a number of other cyber-crime related malware infections were identified and remediated.

7.3.2 Log Management Strengths and Weaknesses

Strengths:

- Good data retention periods on proxy logs (aided by compliance to regulations) allowed for a thorough investigation
- Comprehensive and organised logging enabled rapid data retrieval and a detailed view of the attack
- Host-based logging and tools were available to support incident response processes

Weaknesses:

- Although the company employed a comprehensive approach to logging and had deployed security appliances, they relied heavily on vendor supplied signatures to identify and prevent attacks
- Proactive analysis of logs could have identified malicious activity earlier and reduced the attacker's impact to the company; for example, SIEM alerting for suspicious user agents issuing HTTP GET requests for executable files may have pointed to initial malicious activity progression

8 Preparation for Incident Response

Security breaches are inevitable. At some point an organisation will need to conduct an investigation where log data will be beneficial and may even be of critical importance. Developing an overall strategy for analysing log data while your company is responding to a security event is not recommended[7].

Organisations should conduct exercises to test their ability to gather log evidence by developing readiness exercises build around mock incidents. These exercises will necessarily be different for each organisation depending on where their most vulnerable systems may be found; financial institutions may simulate an attack against their online banking infrastructure, whereas an e-commerce company may focus on attacks against a client credit card database. A small company may only employ a small number of devices that are capable of generating log data thereby creating challenges for comprehensive logging of network activity, while a multinational organisation may encounter issues with third-party service providers and log sources spread across different time-zones and varied security cultures to contend with.

The exercise scenario employed does not have to be complex. It should focus on the sort of incident the organisation is likely to face and then seek to identify the devices able to provide useful log data, establish where that data is and how to retrieve it. Each source should then be validated to ensure that it effectively enables analysis by being in an appropriate format and contains the appropriate data fields required for incident response. Finally, the retention period of each data source should be validated to ensure that it balances the timescale requirements of responding to identified threats with resource constraints. Organisations should take a view as to the length of time which is acceptable to gather the log data for an investigation and this will vary depending on the potential impact of an incident and the complexity in collecting it.

Initially, these exercises will not depend on already having a finalised log management strategy and may actually benefit from repetition to validate that the approach taken is appropriate for the organisation as a whole. If at all possible, this should be one element of a broader response training exercise involving stakeholders from across the business.

Although these incident response readiness scenarios should focus on the core components of an event and do not require every eventuality to be tested in initial instances, it is acknowledged that an organisation may not have the expertise or resources to develop these exercises. In this case, their incident response provider of choice will certainly be able to develop and lead relevant tests.

[7] For more information, see http://www.contextis.co.uk/research/blog/day-ball-not-time-learn-dance/

Context Information Security

London (HQ)
4th Floor
30 Marsh Wall
London E14 9TP
United Kingdom

Cheltenham
Corinth House
117 Bath Road
Cheltenham GL53 7LS
United Kingdom

Düsseldorf
1.OG
Adersstr. 28
40215 Düsseldorf
Germany

Melbourne
4th Floor
155 Queen Street
Melbourne VIC 3000
Australia