The challenge of detecting sophisticated attacks: *Insights from SOC Analysts*

Olusola Akinrolabu
Department of Computer Science
University of Oxford
Oxford
olusola.akinrolabu@cs.ox.ac.uk

Ioannis Agrafiotis
Department of Computer Science
University of Oxford
Oxford
ioannis.agrafiotis@cs.ox.ac.uk

Arnau Erola
Department of Computer Science
University of Oxford
Oxford
arnau.erola@cs.ox.ac.uk

Abstract

The ever-increasing rate of sophisticated cyber-attacks and its subsequent impact on networks has remained a menace to the security community. Existing network security solutions, including those applying machine learning algorithms, often centre their detection on the identification of threats in individual network events, which is proven inadequate in detecting sophisticated multi-stage attacks. Similarly, SOC analysts whose roles involve detecting advanced threats are faced with a significant amount of false-positive alerts from the existing tools. Their ability to detect novel attacks or variants of existing ones is limited by the lack of expert input from SOC analysts in their creation of the tools; and the use of features that are closely linked to the structure of specific malware which detection models aim to identify. In this work, we conduct a literature review on malware detection tools, reflect on the features used in these approaches and extend the feature-set with novel ones identified by interviewing experienced SOC analysts. We conduct thematic analysis to the qualitative data obtained from the interviews, and our results indicate not only the presence novel generic malware characteristics based on network and application events (web proxy, firewall, DNS), but identify valuable lessons for developing effective SOCs regarding their structure and processes.

Keywords Machine Learning, Malware, SOC, SIEM, APT, Botnet, and C&C.

ACM Reference Format:
Olusola Akinrolabu, Ioannis Agrafiotis, and Arnau Erola. 2018. The challenge of detecting sophisticated attacks: *Insights from SOC Analysts*. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

Targeted and sophisticated cyber-attacks are the fastest growing security threats to organisations today [8]. The target of these malicious attacks is no longer limited to governments and financial institutions as demonstrated by Operation Shady RAT, where data collected from a single command and control (C&C) showed one attack affecting 71 companies across 31 industries [13]. The majority of these targeted cyber-attack operations originate from organised, well-funded, sophisticated and often state-sponsored groups, who can bypass traditional security solutions to achieve their aim [30]. With the traditional perimeters of organisations gradually breaking down due to the evolution of cloud computing, bring your own device (BYOD) practices and teleworking technologies [32], institutions need to do more to secure their network. While many of the targeted attacks use techniques that share commonalities with traditional attacks, they differ from conventional malware because they target specific users to gain undetected access to critical assets and utilise sophisticated tools to exploit them [13]. Despite the fact that research from industry and academia is leaning towards data-driven security, as a way of augmenting traditional workflows in Security Operation Centres (SOCs), Sandeep et al. [1] highlight that the process of identifying attacks in network traffic is more of an art than science.

SOCs are an integral part of an organisation's security incident response team and play a critical role in collecting, normalising, storing, and correlating events to identify malicious activities[1]. The security events include logs from network devices such as firewalls, Intrusion Detection Systems (IDS), proxy servers, application servers, Domain Name Server (DNS) infrastructures, etc. Due to a significant amount of logs sent from different sensors, and the increased possibility of false positives, SOC analysts have relied on the use of Security Information and Event Management (SIEM) tools to help cross-correlate logs and raise alerts based on already configured rules [1]. However, the SIEM solution is not optimal in tackling the issues of sophisticated, multi-staged attacks, nor are they efficient in dealing with enterprise-network changes. Despite the long-term event-retention capability of SIEM tools, their algorithms so far have failed to evolve with the nature of the data collected efficiently. Likewise, these SIEM tools are prone to false positives (FP),

because of the sheer amount of logs and the difficulty of automating a heuristic driven process [1]. Lambda Architecture [33] tried to partially resolve this issue with its security event processing framework, which is used to isolate the real-time and historical events. Identifying the limitations of machine learning in their solution, they suggested the use of artificial intelligence and expert systems in manual security processes to maximise personnel efficiency.

The application of machine learning to IDS events has been relatively successful to date, but the majority of this success is highly localised to the particular network environment the algorithm was optimised for, and the datasets used [27]. Some solutions have combined signature and anomaly-based detection to improve malware detection [32], while others have used machine learning together with active learning for anomaly detection [29]. There is, however, still a gap in the use of machine learning in detection systems in "real world" settings, particularly business networks and SOC environments. Sommer et al. [27], suggested that machine learning is of limited use in cybersecurity, claiming that the strength of machine-learning tools lies in finding activities that are similar to something previously seen, i.e. machine learning algorithms are more suited to finding benign traffic than anomalies. They highlight several characteristics of machine learning that do not suit IDS, including, the high cost of errors, lack of training data, and diversity of network traffic.

This paper sets out to provide the foundations for the use of machine-learning approaches in the detection of sophisticated malicious attacks in a SOC. We explore behavioural-based analytics of network traffic to generate generic features that are capable of detecting novel attacks and can be used within any SOC environment. We conduct five semi-structured interviews with experienced SOC analysts to identify behavioural features that characterise Indicators of Compromise (IOC) of a network attack. Our study identifies novel features that can be used to detect sophisticated malware and provides insight into how the SOC could be better equipped to handle the ever-increasing rate of malware infections.

The remainder of the paper is structured as follows: Section 2 reviews the literature concerning IDS, Botnets, and Advanced Persistent Threats (APTs). Section 3 describes our methodology, while Section 4 outlines our findings and discusses suggested features for malware detection. Finally, we conclude the paper in Section 5 and present ideas for future research.

2 Related work/ Literature review

There have been at least two decades of extensive research (both in academia and industry), seeking an optimal solution to network intrusion detection, either signature or anomaly-based [32]. The ever-changing nature of applications, protocols and malicious activities have made this task a serious challenge [18]. Malware is non-deterministic [10]; it is unpredictable, and its changing behaviour is more prevalent when it can communicate with its C&C. Likewise, malware writers can evade signature-based and anomaly-based IDS by using executable packing and code obfuscation to create polymorphic variants of the same malware [21]. Consequently, ongoing research in malware detection is split between analysing malicious activity on the network and detecting botnets that support malware operations [32]. Although many breakthroughs have been recorded in the malware and intrusion detection research space, some of which make use of machine learning, sandboxing and IP reputation systems, there is still a lack of generalisation of these solutions to organisations of all sizes. The reason for this is that some of the IDS evaluations have been either based on simulated data, which is optimised for a single environment, or built in a manner that operationalising them in SOCs is impossible [27].

In recent times, the application of machine learning to intrusion detection has improved the rate of detecting sophisticated attacks even before they can exploit critical systems. Studies such as that of Kwon et al. [11], built a machine learning system, after gaining insights of malware using downloader graph analytics, and this solution was shown to detect malware at an average of 9.24 days earlier than existing antivirus products. However, despite significant improvement in research on the initial exploitation activities of malware, there exists a gap in the increasing number of activities that attackers undertake when on the network [15]. Models such as the ATT&CK model, developed by Mitre Corporation [15], describe at a high-level an adversary's steps once they have control of the system, focusing on the commonly used techniques in network attack. While this model is useful as a reference model, it focuses on detection and response and not on the proactive protection of organisations' networks. The identified gaps support the "No free lunch" theorem, which states that no single model works best for every problem [12, 31].

One of the common attack vectors for malware distribution is drive-by downloads. While most malware detection studies have focused on detecting malware during the first phase of the attack (exploitation phase) using blacklists, attackers have responded by rotating malicious domains using techniques such as Domain flux [9]. Recent research carried out to detect dynamic malware behaviour involve the use of content-agnostic solutions. In [9], Invernizzi et al. developed Nazca a content-agnostic malware detection tool, which detects infection in large-scale networks by analysing the collective effect of installed malware on the network. Similarly, Rajab et al. [23], in trying to address the challenge of false positive in malware detection, proposed CAMP, a content-agnostic malware protection system based on malware binary reputation. The solution is built into the browser and can determine the reputation of most malware downloads locally, thereby protecting the system from infection.

Another solution that detects malware download events, using the download graph analysis technique is Mastino [22]. Utilising global situational awareness and being able to monitor various network and system-level events across the internet, Mastino provides a real-time classification of files and URL to clients. Also, the research of Zhang et al. [34] proposed Arrow, a novel drive-by download malware detection method, which leverages the URLs of the Malware Distribution Network's (MDN's) central servers and generates a set of regular expression-based signatures to detect new malware launched from the central servers.

Another significant aspect of malware detection is feature selection. Previous studies into the generation of features indicative of network malware include that of [10], who explored over 20,000 packet captures (PCAPS) generated by known malware to find their characteristics and determine how malicious traffic differ from benign traffic. Of the analysed samples, he observed that about 49.77% of them utilised HTTP in their 60 seconds of execution and 15,000 samples used DNS to locate their network resources (C&C). Although the paper [10] suggested detecting new malicious traffic utilising a set of rules, this turned out to be less optimal with 81% detection rate. Another research into suspicious activity detection in enterprise networks is that of Yen et al. [32]. Their automation solution, Beehive, was developed to analyse large, disparate log data, and apply behaviour-based detection techniques to identify outliers that can be reported to SOC analysts for further investigation. Beehive [32] was able to partially address significant data challenges in intrusion detection, such as, data normalisation, data optimisation and bridging the semantic gap between logs stored in SIEM and the information that security analysts need to identify suspicious behaviour. Concerning the challenges SOCs face in detecting malicious traffic, Bhatt et al. [1] ponder that these are caused by the isolation of the SOC teams from the enterprise network operations, a point also inferred by [32]. Furthermore, while Bhatt et al. argue that the effectiveness of a SOC depends on its analytical and forensic capabilities, awareness of the enterprise networks, and internal processes, they make a case for further advancement in SIEM tools, especially towards making them adaptive, context-aware, flexible and holistic [1].

Similar to the research of [10], Perdisci et al. developed a novel network-level behavioural-malware-clustering system for HTTP-based malware [21]. The authors of this work proposed the use of a clustering system capable of automatically generating network-level signatures. Similarly, DISCLOSURE [2], a large-scale botnet detection system, applied three groups of features (flow sizes, client access patterns, and temporal behaviour), to distinguish C&C channels from benign ones using NetFlow records. Likewise, Stokes et al. [29] extended the knowledge of malware detection, when they proposed a combination of active learning with rare class discovery and uncertainty identification in the statistical training of a signature-based malware classifier. Other studies include that of Pawar & Bichkar [20], who proposed the use of variable length chromosomes (VLCs) in a Genetic Algorithm-based IDS, and Chi et al. [6], who offered a novel feature-selection method that utilises genetic algorithms to maximise class separation between normal and malicious patterns of network traffic.

In Table 1, we present a cross-section of malware behavioural features that we identified in our review of the literature. We later build on this list, as part of our interviews with SOC analysts.

Reflecting on the research presented in this section, it is evident that bridging the asymmetry that currently exists between attackers and defenders of the network is challenging. It is not enough to detect suspicious activity, but one needs to consider what happens after detection. If the current solutions were 100% accurate at detecting sophisticated attacks, preventing such attacks would be easy. Chandola et al. [5] identified a similar gap in existing anomaly detection techniques, suggesting that most of these assumed no relationship between data instances, as all approaches base their solution on individual network traffic events. Any solution that is capable of optimising logs from multiple network sensors, applications and host machines, increases the situational awareness of the SOC's team in preventing sophisticated attacks. We aim to bridge this gap, by interviewing SOC analysts to elicit features that will efficiently correlate data over many days or months, in an attempt to discover any persistent threat in the environment.

3 Methodology

Typically, the detection of sophisticated attacks is based upon identifying anomalies in network traffic using intrusion detection systems. The use of machine learning is also instrumental in detecting these anomalies, and so far, the methods employed in building most machine learning models have involved selecting features based on manual network analysis, network metadata or structural similarities [5, 27]. Although these approaches have been relatively successful, we identified a gap in the existing solutions with regards to how they leverage SOC analysts' experience, whose role involves detecting advanced threats on a daily basis. We argue that because sophisticated attacks can occur over a significant amount of time, where each stage of the attack spread out in some cases for months [19], it could be difficult to model an adequate classifier in machine learning without accurate information about relevant features. We adopt a different approach to this by conducting interviews to identify features based on malware pattern and behaviours, which are agnostic of the malware structure. Therefore, we aim to improve anomaly detection by extending the feature list used in

Table 1. Malware behavioural features identified in Literature

No.	Abbrev-iation	Feature Description	Class	Src
1	Pkt_var	Strange file sizes - too large or too small (variance in size)	Web	[35]
2	TCP_NAck	More than three successive packets in the same direction without a return packet	TCP	[26]
3	Reg_Pkt	Same src/destination flow with same sized packets sent at regular intervals	TCP	[25]
4	Out_ SYN	OPEN SYN connections to 3 or more destinations without acknowledgement from servers for a duration of 3 seconds	TCP	[36]
5	Mal_Dst	The number of unpopular destination networks accessed by the same host	IP	[32]
6	DNS_Tun	Large size of request and response packets in domain name query exchanges	DNS	[17]
7	UDP_80	HTTP traffic over UDP port 80	Web	[10]
8	Encr_80	Encrypted, traffic on TCP/UDP ports 80	Web	[12]
9	HTTP_Rfr	Missing Referrer and missing X-forwarded-for in HTTP request	Web	[26]
10	No_ Resp	Zero HTTP response from the server.	Web	[26]
11	HTTP_Bin	HTTP GET request for *.exe (or POST containing config, binary or drop zone)	Web	[7]
12	HTTP_Ref	Uncommon HTTP referrer in user browsing history	Web	[32]
13	XOR_Enc	The use of XOR encryption for data obfuscation	Web	[26]
14	T_Urg	The number of urgent packets in the network traffic from and to a single host	TCP	[12]
15	Mal_IPs	Outbound traffic to IP addresses that belong to the same ASN as a set of known malicious IPs	IP	[7]
16	DNS_RR	Name server response to a domain query request, containing more than 5 IPs in the response.	DNS	[30]
17	New_Dom	Domain requests for a new domain whose age is less than one day old.	DNS	[14]

machine learning algorithms, with malware behavioural patterns which are agnostic of the malware structure and have been obtained from interviewing experienced SOC analysts.

The analysts interviewed work for our industry partner and are tasked with protecting the network of many businesses in the UK and overseas from advanced threats. We reflect on the interviews and identify features that characterise the behaviour of sophisticated attacks. We combine this insight with the features identified from our reflections on literature for a full list of features that can be used to build a machine-learning model. The responses from the participants are analysed using thematic analysis to identify important themes that emerged from the interviews.

3.1 Interviews

Our interviews were conducted in a semi-structured way, to enable smooth flow of information between the interviewer and the respondents. Semi-structured interviews provided the researcher and the SOC analysts the opportunity to elaborate more on a subject, permitted the interviewer to ask unplanned questions or emphasise on points of interest raised by the interviewee. In our discussions with the SOC analysts, we assured them of the confidentiality of the information provided, and we treated the subject as sensitive, refraining from pointing out any gaps in their SOC process, despite our knowledge of the subject area. The reason for this approach was to ensure that we had the continuous cooperation of the analysts [24].

To address the purpose of our study, each question we asked the respondents was aimed at understanding the approach of analysts to malware detection. This included asking about the tools used, the implemented SOC processes and the IOCs they look out for in detecting sophisticated malware. The following is a list of common questions we asked each analyst to establish a baseline of facts about the SOC operations and their general approach to advanced threat detection.

(a) Can you describe your role within the SOC?
(b) What is your daily routine?
(c) What are the tools you use as part of your daily activity and how do you use these tools?
(d) Do you have a written process for carrying out your SOC activities?
(e) What is the process of detecting threats in the SOC? Is it all automated?
(f) How often do you detect advanced threats? Do you track your SOCs detection rate?
(g) How do you combine knowledge and tools to identify advanced threats in network traffic? What is the percentage split?
(h) What are the typical behaviours you see with malware detected in this SOC or based on experience?
(i) How are threats handled within the SOC?

(j) How do you verify alerts that signal malware operation?
(k) Any other idea you want to share with us?

3.2 Thematic Analysis and Feature Selection

The data gathered during the interview process was subject to thematic analysis (TA), which is a proven method for identifying, analysing and recording patterns within a qualitative research dataset [3]. Our use of TA was based on its theoretical flexibility, ability to generate unanticipated insights, and its ability to highlight similarities and differences while providing meaning across the dataset [3]. Our approach to the thematic analysis was inductive since the content of our interview data directed the coding and theme development. The features identified as a result of the interviews were based on the behavioural description of sophisticated attacks that have been seen by our respondents. In addition to the generation of features for malware detection, our coding of the interview data also provided other themes applicable to the optimisation of SOC operations, particularly around its processes and stakeholder interaction.

3.3 Ethics Approval

The University of Oxford ethical approval committee, under Ref No: R45502/RE001 approved this research work involving external participants. The research plan described the use of interviews as the qualitative method of research. Other details about the anonymity of participants' data and seeking permission from interview respondents to be recorded was requested and approved by the board. All participants were presented with a consent form detailing the purpose of the study, its ethics committee approval and their ability to withdraw their data at any time. We also assured the participants that their data would be stored securely following the University of Oxford ethical standards.

4 Findings, Discussion and System Overview

In this chapter, we provide details of our interviews with security analysts and explain how insights from our qualitative research resulted in identifying malware features for detecting advanced threats in SOCs.

4.1 The Interview Process

A total of five analysts who work in various capacities within the SOC were interviewed as part of this process. The meetings took place at our partner's premises, in closed-door meeting rooms over a 3-day period and each interview lasted an average of 75 minutes. As explained in the methodology, each interview was conducted in a semi-structured manner, around a set of predetermined open-ended questions, with further discussions emerging from the responses provided by the interviewees. We targeted analysts who work in different capacities within the SOC to get a holistic overview of it, its operations and potential shortfalls. The manager of the SOC nominated the respondents based on their expertise, experience and availability. Each respondent was able to share with us their approach to detecting malware and their experience regarding malware behaviours. The details of the participants are presented in Table 2.

Table 2. The list of interview respondents, their position and job role

Name	Position	Years of Experience	Role/Activities
SOC-A1	SOC Analyst Level 2	8yrs	Working with SIEM and trigger-based tools to monitor customer network traffic.
SOC-A2	Malware Hunt Analyst	3yrs	Responsible for Malware behaviour analysis & Zero days exploit hunting.
SOC-A3	Threat Analyst	6yrs	Responsible for identifying new threats that are capable of exploiting network vulnerabilities.
SOC-A4	SOC Manager / Malware Analyst	15yrs	Being involved in Incident response and static/dynamic analysis of malware.
SOC-A5	Malware Hunt Analyst	13yrs	Responsible for Malware behaviour analysis & Zero days exploit hunting.

4.2 Industry Partners' SOC Setup

It is beneficial to establish that our industrial partner does not refer to themselves as a standard SOC, tasked with monitoring customer networks and raising alerts to inform customers of an incident or a breach of their system. Instead, they operate a new-breed SOC solution, recently referred to by Gartner as a Managed Detection and Response (MDR) Service [4]. This solution includes mapping out customers' attack surface, identifying organisations' critical assets, strategically placing security devices at vantages to capture network traffic, improving intelligence on detection and response to cyber-attacks and making use of a red team to evaluate customers' network posture.

As shown in Figure 1, the MDR is comprised of seven teams, with some analysts adopting a dual role, based on their skill-sets. Operationally, the teams involved in the day-to-day activities include threat intelligence, malware hunt, incident response, threat analysis and the event monitoring (SOC analysts I & II) team. The other teams get called

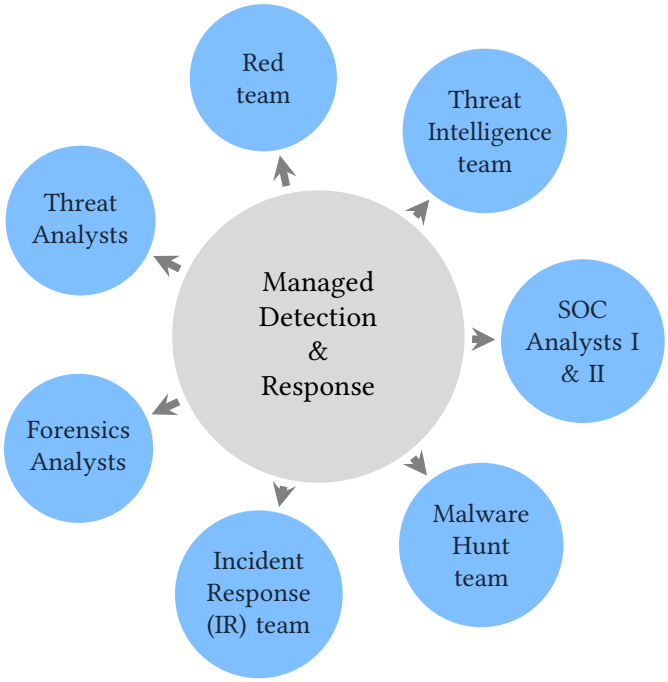

Figure 1. The components of a typical Managed Detection and Response team

upon occasionally to perform specialised duties during a significant incident.

We identified from the interviews that advanced threat detection should begin with network monitoring, where SOC analysts are tasked with using correlation capabilities of SIEM tools to detect known attacks. Threat analysts are the next in the command chain from the SOC analysts, they look into other log sources, not available in the SIEM tools, to identify unknown attacks or IOCs that could signal new threats. Threat analysts also work with malware analysts to perform static and dynamic analysis of malware, aiming to observe malware behaviour, which drives the development of appropriate rules for the SIEM tools. The malware hunt team are tasked with hunting for new zero-day attacks both online or within big data log sources. They specifically focus on IOC and recent attacks using Open Source Intelligence (OSINT) sources. In some cases, they receive closed-source intelligence from government partnerships such as the Cyber Security Information Sharing Partnership (CISP) or corporate sources. In occasions where the malware hunt team identifies a zero-day attack, they verify the findings with the malware analysis team, who works with Anti-Virus (AV) vendors for the release of a new signature indicating the occurrence of such an attack.

The threat intelligence team is tasked with understanding the nature of a zero-day attack and its potential effect on the SOC-supported organisations. They also assist SOC analysts in identifying the pattern of the malware in monitored logs and update SIEM rules to facilitate the detection of these attacks. The threat intelligence team keeps up-to-date with events in the security community through regular visits to OSINTs, establishing research collaborations with academic institutions and working with vendors of security devices they utilise within the SOC.

4.3 Findings from the Interviews

The thematic analysis of the interview transcriptions began with familiarising ourselves with the data. This analysis progressed to the generation of initial codes and the collating of relevant data applicable to each code. Our collated codes were later categorised into potential themes, and in reviewing and refining our themes, we identified three distinct and compelling themes. These three principal themes, namely people (analysts), operations (processes) and technology (tools) appeared in all five interviews (Figure 2) and are comprised of sub-themes.

What follows is a summary of the excerpts from the interview dataset, categorised based on the themes identified above. Each of the points identified below contributes to our understanding of malware behaviour and the detection of sophisticated attacks in SOCs.

Theme 1 : People

(i) As commonly discussed within the security research community [27], analysts likewise identified that their major challenge with malware detection is looking for what makes up benign traffic and then finding the anomaly.

(ii) When probed on how quickly attacks are detected within the SOC, SOC-A4 identified their limited ability to detect malware during the initial stages of the attack. He asserted that, *"Malware detection within a SOC occurs mostly after the malware has established a foothold on the network and is sending outbound traffic"*.

(iii) The respondents agreed that a big limitation to automated malware detection is due to limited visibility of the SOC to customer data. SOC-A3 summarises the problem area saying, *"failure in malware detection is due to the limited visibility of security monitoring tools to lateral movement occurring between hosts on a network"*.

(iv) Analysts agreed that OSINT helps to reduce the efforts needed to detect malware by providing IOCs, but SOC-A4 says, *"Caution is required since malware authors may also try to infiltrate such groups"*.

Theme 2 : Operations

(i) Analysts agreed that the majority of malware seen in SOCs are commodity malware without any particular target and as SOC-A4 indicated *"they have the same behaviour"*.

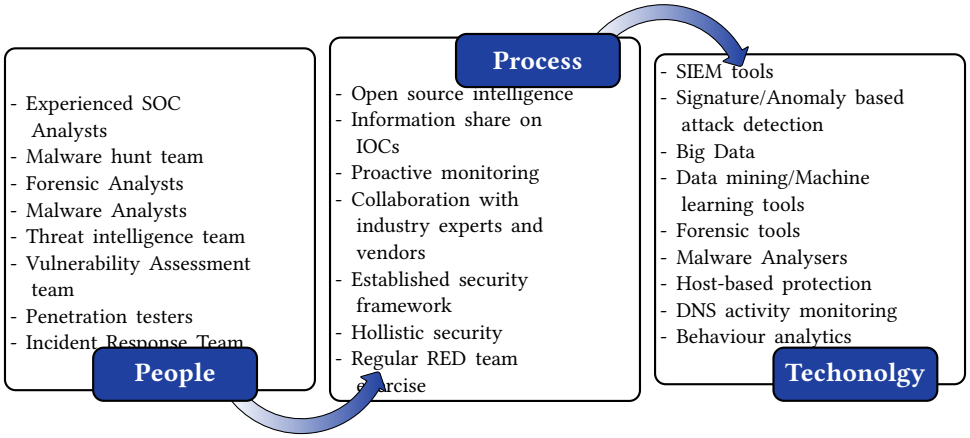

Figure 2. Three major themes for improving SOC operations

(ii) The prevalence of DNS-based malware was evident in the SIEM rules used for malware detection. When probed for the reason for this, SOC-A3 responded saying *" the majority of the malware operations rely on DNS, but DNS activity monitoring is widely ignored in enterprise security".*

(iii) Analysts identified the importance of context when analysing network data with rule-based detection tools. SOC-A1 further clarified this point by adding "attacks get missed due to the inability of rules to cover all scenarios".

(iv) The use of callback traffic is widely deployed in botnets and as SOC-A2 indicated *"monitoring such traffic is the most common means of detecting commodity malware".*

(v) We identified that nation-state APT-type attacks avoid the use of callback feature. When asked about this, SOC-A2 answered affirmatively, supporting this claim by saying that *" they operate their attacks in a self-containing manner".*

(vi) Two respondents SOC-A2 and SOC-A3 confirmed that targeted attacks normally include the name of their target as part of the malware code, e.g. @xxxx. SOC-A2 says, *"This enables malware authors to measure their success".*

(vii) Anomaly detection systems used in user profiling remain a valuable solution for malware detection within an enterprise and judging by the number of such solutions within the SOC, participant SOC-A3 says, *"it helps capture deviation in normal user network behaviour".*

Theme 3: Technology

(i) Analysts admit that malware detection is easier with signature-based systems when the malicious behaviour is known, and as SOC-A1 further elaborated, *"there are not many attacks that cannot be detected with the help of a security rule or signature".*

(ii) When asked about the current APT detection solution used within the SOC, participant SOC-A4 pointed out that *"APTs are rare, targeted, and have a short lifetime, which makes designing solutions to address them quite complex".*

(iii) Malware detection and attack response are two-fold processes: *Tools-led vs Analyst-led.* Analysts knowledge verifies the result of tools, but the intelligence gathering using these tools is the starting point.

4.4 Discussion

Our analysis of this SOC's operation, highlights the need for automation, including the use of big data analysis, cloud technologies and machine learning techniques in malware detection. Among the interesting observations regarding SOC operation was how much intelligence they gather from open source forums. The malware hunt team in charge of locating zero-day attacks use OSINT sources to collect information on recent attacks and malware mode of operation. Some OSINTs websites also provide an Application Programming Interface (API) for specific SIEM tools to be able to upload daily IOCs that have been observed across the security community. This strategy assists SOC analysts in streamlining the investigation process, providing a more efficient method of narrowing down the occurrence of malicious behaviour in customer logs, and ultimately improving response and mitigation processes.

An aspect of malware detection which received consensus amongst the respondents was the lack of adequate network and application logs. They asserted that SOC customers due to data confidentiality or fear of overexposure to an organisation's asset, limit the available logs for malware detection to web proxy and firewall logs. This level of logging was deemed inadequate for detecting sophisticated attacks or automating the detection process. At a minimum, we gathered that organisations who seek to improve attack detection and response performance from their SOC must be prepared to

share a variety of logs including, DNS, client authentication, web proxy, firewall, IPS, and, VPN logs. This helps to improve situational awareness and to close the semantic gap between logs collected in SIEMs and the information needed by security analysts to identify suspicious network behaviour [32]. As SOC-A3 pointed out, attacks such as DNS tunnelling and DNS-based Botnets [7, 17], have remained hidden within the enterprise network because of the lack of security monitoring at vantage points. In general terms, this means that *the network blind spots have limited malware detection to the entry or exit points of an organisation's network.* Although security logs contain a gold mine of analytic information, particularly in areas of user behaviour analysis, proactive monitoring, malware detection, and mapping customer IT infrastructure, their volume and structure pose a challenge to SOC analysts [28]. From our interviews, we identified the need for tools capable of normalising different system logs, for example, DNS and web proxy logs, for improved correlation between multi-system events. Conversely, there is also a need for a universally agreed format for the transfer of security logs, improving on the current solutions of using Syslog or compressed file transfer.

In summary, we found out that detecting sophisticated attacks even in advanced SOCs, still required a coordinated and complementary effort amongst the various teams of the SOC. Likewise, we observed the need for correlation of network traffic across multiple customers' environments, which are separated into partitions, based on their security level or subscribed service. Finally, we identified the need for malware detection tools to consider the context of network traffic, through correlation of events from multiple sources of network traffic.

4.5 List of Behavioural Features for detecting Sophisticated Attacks

From the features identified during the interview phase, we noticed that many of the malware detection approaches described or shown to us by the respondents were linked to the behavioural patterns of HTTP and DNS-based malware. These lists of features also include intelligence gathered by malware analysts from OSINTs. Malware behaviour analysis is described in [37] as the process of understanding the characteristics and behaviour of malware software, for easy identification, classification and clustering. From our interviews with the SOC analysts, we recognised that some of the tools used in detecting malware focused on isolated events, while others correlated multiple events. Similarly, we identified that some of the features for malware detection were dependent on a single flow between source and destination, while others required multiple flows over a fixed period.

In the following subsections, we describe the features identified from our interviews. The Tables 3 list the feature names, their description, their network traffic class and their origin.

Although each of the features identified in Tables 3 are self-explanatory, we would like to expand on a few of them. First is the (HTTP_Rnd) feature, which helps to identify a host visiting random URLs at specific times of the day. A respondent shared an example of such behaviour with us, where a malware sent an outbound query that searched for a bird on a wrestling website, and a few hours later posted similar traffic to a cycling shop website. This event was not detected with SIEM tools but was manually identified by the analyst. Another feature worth highlighting is the DNS_Spike, which helps to detect hosts in the network that makes a significant number of DNS request within 60 secs. We learnt that this sort of malware behaviour is because many of these domains are hard-coded to the software, and for the malware to know which of the C&C is active, they need to resolve the domain names. Overall, these features complement the initial list of malware behavioural features identified as part of our literature review, resulting in a comprehensive list of malware features that can be used in a machine-learning malware detection tool.

5 Conclusion/Future Work

The primary goal of this study was to determine ways of optimising the detection of sophisticated attacks in SOC environments. With the effect of targeted malicious attacks no longer limited to governments and large-sized corporate firms, SOCs have become occupied with digging out malicious traffic from an ever-increasing number of logs of benign traffic. Although the majority of these attacks are variants of one another, they succeed in bypassing traditional security solutions, increasing the asymmetry between defenders and attackers in favour of the attackers. In this study, we identified the inadequacy of existing IDS solutions in detecting multi-staged, low & slow attacks. We further observed that latest academic research in this area, mainly utilising machine learning techniques, also failed to address the gaps exploited by malware vendors. Our review of existing literature showed that many of the machine learning models were tailored to a particular user environment to yield good detection rates but lacked generalisation and adaptability. Likewise, the features selected in developing these models included statistical features which derived directly from training sets, or they were based on the structure of the malicious traffic, which in both cases limits the ability of the model to detect new attack variants.

Our strategy to address the challenges mentioned above was to propose a solution capable of detecting malware based on behavioural analytics of malicious traffic pattern. Our two-pronged approach to the problem involved interviewing SOC analysts to shed light on prevalent malware behaviours in the SOCs, as well as reflect on literature that addressed malware detection. The interviews with the SOC analysts confirmed some of our literature review findings, and also

Table 3. Malware behavioural features identified from the Interviews

No.	Abbrev-iation	Feature Description	Class	Src
1	HTTP_Rnd	Access to random URLs, with similar patterns at scheduled times of the day	Web	Intv
2	T_Pkts	The average time between packets in time interval	TCP	Intv
3	Login_FA	Consecutive failed login attempts before a successful login	Auth	Intv
4	Conn_Len	Long periodic connection (frequency 5s-60s), to a high port, with large packet sizes	TCP	Intv
5	ICMP_P	Strange pings (broad spectrum pings)	ICMP	Intv
6	PUSH_Reg	Nos. of outgoing push data towards the same dst at regular intervals	TCP	Intv
7	DNS_Spike	Domain spike (>ten requests) and connection bursts from a host within a 60 second period	DNS	Intv
8	DNS_Isol	DNS request made without corresponding request by another application e.g. HTTP	DNS	Intv
9	UA_Blank	Empty User-Agent strings in HTTP header or malformedUA string	Web	Intv
10	Cert_Ano	Anomaly in TLS, certificate -subscriber and issuer organisation fields empty	Web	Intv
11	Mal_IPD	Direct access to destinationURL using IP address instead of domain name	Web	Intv
12	HTTP_Call	Callback HTTP traffic with only one isolated request to the domain	Web	Intv
13	DNS_Ent	Domain names with high entropy	DNS	Intv
14	DNS_TTL	Short Time-To-Leave,(TTL) in domain response (e.g. <=150secs)	DNS	Intv
15	DNS_Empty	Outbound DNS request, with no named query or encoded message	DNS	Intv
16	DNS_ NX	NX (non-existent, domains) responses in DNS reply	DNS	Intv
17	TCP_Retra	The number of reconnects between IP src/dst pair	TCP	Intv

provided valuable insights into other areas of malware detection, which seems to be neglected in the literature. Some of our highlights include the rarity and short lifetime of real APT-type attacks, the emergence of DNS tunnelling as a method used by malware for exfiltration, and the requirement for automated machine learning tools to emulate SOC analyst thinking with minimal guidance. We were able to outline the setup of a typical Managed Detection and Response (MDR) team, which in lay terms can be referred to as an advanced SOC, which if implemented, would further optimise a SOC in detecting sophisticated attacks.

While it is difficult to generalise the findings of this study, due to the relatively small number of experts interviewed, and their affiliation to a single SOC, we argue that this paper enhances the understanding of SOC processes, their challenges and the nature of malware detection solution that could optimise their operation. In this study, we found out the limitation of existing machine-learning based detection solutions and how such solutions can be improved with behavioural analytics and feedback from SOC analyst for optimisation. We highlight the pivotal role of SOC analysts, arguing that this role cannot be replaced by automated systems and can only be enhanced with tools targeted to make SOC operations better. Furthermore, the thematic analysis of the interview data yielded three broad themes (people, operations, and technology), and how the optimisation of SOCs relied on the integration of these themes and their constituent elements. Overall, this study strengthens the idea that malware detection should not be based solely on individual events from a host, but on correlated network and application traffic. Our proposed set of features can form the basis for a malware detection model since they are based on malware pattern and behaviours, and are agnostic of the malware structure.

For our future work, we plan to implement a framework that aims to categorise network traffic into either benign or malicious clusters. This should be achieved based on observations from malicious behaviours or properties. We intend to validate the ability of our approach to detect malicious hosts by testing the system on historical logs. Furthermore, based on the performance of initial model and feedback from SOC analysts, our goal is to improve on the feature list by adding more behaviour-based features to the solution. We aim to carry out static and dynamic analysis of old and new malware to obtain new malware behaviours, which are not part of our current solution. The ultimate goal will be to have our solution complementing a signature-based IDS/IPS as part of an inline network security device. Our decision to develop a real-time tool is to widen its application prospect and not limit it to an analytical and audit system. To evaluate this solution on detecting new threats, we will be making use of malware samples from sources, such as Malfease [16] to determine the accuracy of the solution, and benchmark it against other industry solutions.

6 Acknowledgement

This research project has been possible thanks to a DPhil grant from EPSRC and Kellogg College, University of Oxford. The authors would like to thank the management of the Advanced SOC organisation for their assistance and openness

throughout the study. Also, we appreciate the anonymous reviewers for their helpful and constructive comments.

References

[1] Sandeep Bhatt, Pratyusa K. Manadhata, and Loai Zomlot. 2014. The operational role of security information and event management systems. *IEEE Secur. Priv.* 12, 5 (2014), 35–41. https://doi.org/10.1109/MSP.2014.103

[2] Leyla Bilge, Davide Balzarotti, William Robertson, Engin Kirda, and Christopher Kruegel. 2012. Disclosure: Detecting Botnet Command and Control Servers Through Large-scale NetFlow Analysis. *Proc. 28th Annu. Comput. Secur. Appl. Conf.* (2012), 129–138. https://doi.org/10.1145/2420950.2420969

[3] Virginia Braun and Victoria Clarke. 2006. Using thematic analysis in psychology. *Qualitative research in psychology* 3, 2 (2006), 77–101.

[4] Toby Bussa, Craig Lawson, and Kelly M. Kavanagh. 2016. Market Guide for Managed Detection and Response Services. https://www.gartner.com/doc/3314023/market-guide-managed-detection-response

[5] Varun Chandola, Arindam Banerjee, and Vipin Kumar. 2009. Anomaly detection: A survey. *ACM Comput. Surv.* 41, September (2009), 1–58. https://doi.org/10.1145/1541880.1541882

[6] Chi Hoon Chi Hoon Lee, Jin Wook Jin Wook Chung, and Sung Woo Sung Woo Shin. 2006. Network Intrusion Detection Through Genetic Feature Selection. In *Seventh ACIS Int. Conf. Softw. Eng. Artif. Intell. Networking, Parallel/Distributed Comput.* IEEE, 109–114. https://doi.org/10.1109/SNPD-SAWN.2006.52

[7] Dhia Mahjoub Dhialite, Thibault Reuille, and Andree Toonk. 2013. Catching Malware En Masse : Dns and Ip Style. *Opendns* (2013), 1–33.

[8] Paul Giura and Wei Wang. 2013. A context-based detection framework for advanced persistent threats. *Proc. 2012 ASE Int. Conf. Cyber Secur. CyberSecurity 2012* SocialInformatics (2013), 69–74. https://doi.org/10.1109/CyberSecurity.2012.16

[9] Luca Invernizzi, Stanislav Miskovic, Ruben Torres, Christopher Kruegel, Sabyasachi Saha, Giovanni Vigna, Sung-Ju Lee, and Marco Mellia. 2014. Nazca: Detecting Malware Distribution in Large-Scale Networks.. In *NDSS*, Vol. 14. 23–26.

[10] Kiel Wadner. 2013. 60 Seconds on the Wire : A Look at Malicious. *SANS Institute* (2013), 0–35.

[11] Bum Jun Kwon, Jayanta Mondal, Jiyong Jang, Leyla Bilge, and Tudor Dumitras. 2015. The dropper effect: Insights into malware distribution with downloader graph analytics. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM, 1118–1129.

[12] W Li and A W Moore. 2007. A machine learning approach for efficient traffic classification. *Proc. IEEE MASCOTS* (2007), 310–317.

[13] McAfee. 2011. Combating Advanced Persistent Threats. *Whitepaper* (2011), 1–8. https://securingtomorrow.mcafee.com/mcafee-labs/combating-malware-and-advanced-persistent-threats/

[14] Leigh B Metcalf and Jonathan M Spring. 2013. Passive Detection of Misbehaving Name Servers Passive Detection of Misbehaving Name Servers. (2013).

[15] Mitre. 2015. Adversarial Tactics , Techniques , and Common Knowledge ATT & CK Matrix Purpose. (2015).

[16] ISC OARC. 2016. Project Malfease. http://malfease.oarci.net

[17] Open DNS Inc. 2011. The Role of DNS inBotnets. *Open DNS Security Whitepaper* (2011). http://info.opendns.com/rs/opendns/images/WB-Security-Talk-Role-Of-DNS-Slides.pdf

[18] Sven Ossenbuhl, Jessica Steinberger, and Harald Baier. 2015. Towards Automated Incident Handling: How to Select an Appropriate Response against a Network-Based Attack? *Proc. - 9th Int. Conf. IT Secur. Incid. Manag. IT Forensics, IMF 2015* (2015), 51–67. https://doi.org/10.1109/IMF.2015.13

[19] Dirk Ourston, Sara Matzner, William Stump, and Bryan Hopkins. 2003. Applications of hidden markov models to detecting multi-stage network attacks. In *System Sciences, 2003. Proceedings of the 36th Annual Hawaii International Conference on*. IEEE, 10–pp.

[20] Sunil Nilkanth Pawar and Rajankumar Sadashivrao Bichkar. 2015. Genetic algorithm with variable length chromosomes for network intrusion detection. *Int. J. Autom. Comput.* 12, 3 (2015), 337–342. https://doi.org/10.1007/s11633-014-0870-x

[21] Roberto Perdisci, Wenke Lee, and Nick Feamster. 2010. Behavioral Clustering of HTTP-based Malware and Signature Generation Using Malicious Network Traces. *Proc. 7th USENIX Conf. Networked Syst. Des. Implement.* (2010), 26.

[22] Babak Rahbarinia, Marco Balduzzi, and Roberto Perdisci. 2016. Real-Time Detection of Malware Downloads via Large-Scale URL-> File-> Machine Graph Mining. In *Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security*. ACM, 783–794.

[23] Moheeb Abu Rajab, Lucas Ballard, Noé Lutz, Panayiotis Mavrommatis, and Niels Provos. 2013. CAMP: Content-Agnostic Malware Protection.. In *NDSS*.

[24] Lee Raymond and Claire Renzetti. 1990. The problem of researching sensitive topics. *American Behavioral Scientist, Vol. 33 No. 5., Sage Publications,* (1990).

[25] Konrad Rieck, Philipp Trinius, Carsten Willems, and Thorsten Holz. 2011. Automatic Analysis of Malware Behavior using Machine Learning. *J. Comput. Secur.* (2011), 1–30. https://doi.org/10.3233/JCS-2010-0410

[26] Terence Slot. 2015. Detection of APT Malware through External and Internal Network Traffic Correlation. *Master Thesis, University of Twente* March (2015).

[27] Robin Sommer and Vern Paxson. 2010. Outside the Closed World: On Using Machine Learning for Network Intrusion Detection. *2010 IEEE Symp. Secur. Priv.* 0, May (2010), 305–316. https://doi.org/10.1109/SP.2010.25

[28] Splunk. 2017. Logging with Splunk. http://dev.splunk.com/view/logging-with-splunk/SP-CAAADP5

[29] Jack W Stokes, John C Platt, and Joseph Kravis. 2008. ALADIN : Active Learning of Anomalies to Detect Intrusion. *Microsoft* (2008).

[30] Nikos Virvilis and Dimitris Gritzalis. 2013. The big four - What we did wrong in advanced persistent threat detection? *Proc. - 2013 Int. Conf. Availability, Reliab. Secur. ARES 2013* (2013), 248–254. https://doi.org/10.1109/ARES.2013.32

[31] David H Wolpert. 2012. What the No Free Lunch Theorems Really Mean ; How to Improve Search Algorithms. *Working Paper, Santa Fe Institute* (2012).

[32] Tf Yen, Alina Oprea, and K Onarlioglu. 2013. Beehive: large-scale log analysis for detecting suspicious activity in enterprise networks. *Proc. 29th Annu. Comput. Secur. Appl. Conf.* (2013), 199–208. https://doi.org/10.1145/2523649.2523670

[33] Joseph Zadeh, George Apostolopoulos, Christos Tryfonas, and Muddu Sudhakar. 2015. Defense at Scale: Building a Central Nervous System for the SOC. *Blackhat 2015* (2015), 1–8.

[34] Junjie Zhang, Christian Seifert, Jack W Stokes, and Wenke Lee. 2011. Arrow: Generating signatures to detect drive-by downloads. In *Proceedings of the 20th international conference on World wide web*. ACM, 187–196.

[35] David Zhao, Issa Traore, Bassam Sayed, Wei Lu, Sherif Saad, Ali Ghorbani, and Dan Garant. 2013. Botnet detection based on traffic behavior analysis and flow intervals. *Comput. Secur.* 39 (2013), 2–16. https://doi.org/10.1016/j.cose.2013.04.007 arXiv:arXiv:1011.1669v3

[36] Chenfeng Vincent Zhou, Christopher Leckie, and Shanika Karunasekera. 2010. A survey of coordinated attacks and collaborative intrusion detection. *Comput. Secur.* 29, 1 (2010), 124–140. https://doi.org/10.1016/j.cose.2009.06.008

[37] Mohamad Fadli Zolkipli and Aman Jantan. 2010. Malware behavior analysis: Learning and understanding current malware threats. *Proc. - 2nd Int. Conf. Netw. Appl. Protoc. Serv. NETAPPS 2010* 2009 (2010), 218–221. https://doi.org/10.1109/NETAPPS.2010.46