Extracting Attack Narratives from Traffic Datasets

Jose David Mireles
Department of Computer Science
University of Texas at San Antonio
San Antonio, TX 78249
Email: bgi872@my.utsa.edu

Jin-Hee Cho
US Army Research Laboratory
Adelphi, MD
Email: jinhee.cho@us.army.mil

Shouhuai Xu
Department of Computer Science
University of Texas at San Antonio
San Antonio, TX 78249
Email: shxu@cs.utsa.edu

***Abstract*—Parsing through large amounts of network traffic to extract attack signatures is a complex and time consuming process. It is an even harder process to piece together those signatures to formulate an attack narrative. An attack narrative can be defined as the set of attack signatures, that when combined provides an overview of the attack and the attacker themselves. In this paper, we propose a framework for extracting attack narratives from traffic datasets. Within this framework, we propose the re-examination of packet grepping for attack signatures in network traffic as a viable, fast, and effective means to extract attack narratives from large amounts of network traffic. By combining attack signature packet grepping with Mandiant's Attack Lifecycle Model, we increase the effectiveness of packet grepping and create a methodology that is simple and powerful for constructing attack narratives. In order to show the effectiveness of the framework, we conduct a case study by using the 2015 National Collegiate Cyber Defense Competition (NCCDC) network traffic. Our preliminary results show that the framework is promising.**

***Index Terms*—Cyber attacks, attack narratives, attack signatures, cyber attribution, attack attribution, data analytics**

1. Introduction

Attack signatures are patterns that can be used to identify attacks, just like malware signatures that can be used to recognize malware samples. Attack signatures can be described as a set of operations, such that combinations of such signatures constitute a cyber attack. Symantec defines an attack signature as "*a unique arrangement of information that can be used to identify an attacker's attempt to exploit a known operating system or application vulnerability*" [1]. Once identified, these signatures can be used in signature-based intrusion detection (IDS) and prevention systems (IPS) as rules and even for understanding the intent of attackers.

Attack signatures are useful to analyze the steps of cyber attacks. However, the *microscopic* view of cyber attacks exposed by individual signatures is not sufficient for the defenders' mission aiming to understand the *macroscopic* view of cyber attacks, such as *Mandiant's Attack Life Cycle* [2] or *Lockheed Martin's Cyber Kill Chain* [3]. Therefore, it is urgent to develop effective methods that are critical to piece together attack signatures for defenders to understand cyber attack situations. In this paper, we tackle this problem which is very challenging in the following way: (1) thousands of attack signatures exist in a detection system and represent a massive amount of work done by the intrusion detection community. However, they are difficult for any single person to understand and manage due to the large volume of the work; and (2) it is especially difficult to sift through the thousands of alerts to investigate whether or not the attack signatures are generated in any meaningful way. Alternatively, actual attacks may be detected at the point where it can no longer be stopped as the attacker has succeeded in their goal. Given such complexity, it is not trivial to derive an effective analysis from such attack signatures, and it is even harder still to assign attribution to an attacker from signatures alone. As a consequence, most research in network traffic analysis is solely focused on finding evidence for a particular event [2]. This is most likely due to the overwhelming size of network datasets and the amount of non-malicious traffic, which can be considered as white noise. Accordingly, we tackle how to find malicious activities in a large dataset in a quick and effective manner, and further piece them together to formulate a comprehensive cyber attack situational awareness.

This work has the following **key contributions**:

- We use the concept of *attack narratives* to piece together individual attack signatures into structures reflecting a macroscopic view of cyber attacks. We map attack signatures against Mandiant's Attack Life Cycle, such that an attack narrative is the story that can be woven together when each detected attack signature is mapped to a step in the lifecycle model. This enables a defender to understand the attacker's tactics to achieve attack attribution by uniquely identifying the attacker.
- We further propose a framework to extract attack narratives from traffic datasets, and conduct a case study on the effectiveness of the framework via a real dataset. More importantly, the proposed framework eliminates white noise from the dataset by extracting attack signatures from packet capture datasets

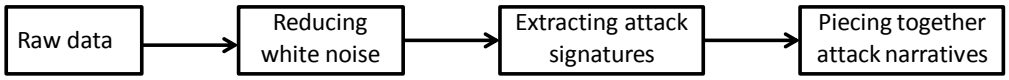

Figure 1. The framework for extracting attack narratives from network traffic.

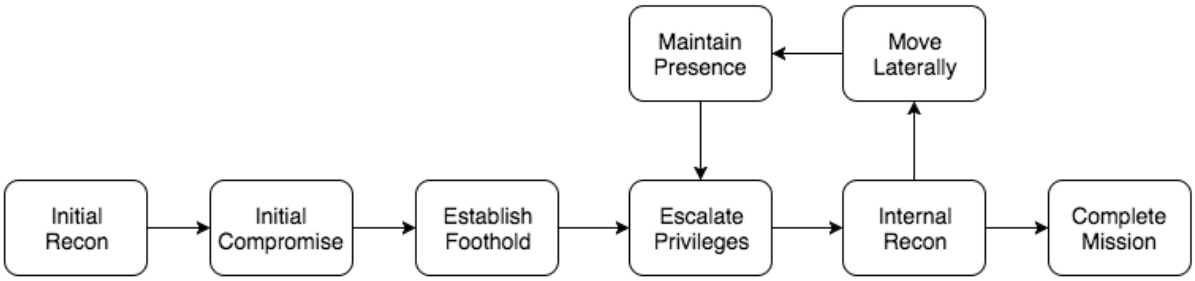

Figure 2. Mandiant's Attack Lifecycle Model (adapted from [2])

(PCAP), which is critical due to the benign nature of most network content.

- We propose a novel way to interpret attack signatures in order to create end-to-end attack narratives by mapping attack signatures in a meaningful way to the Mandiant Attack Lifecycle Model. Via this way, organizations can quickly find missing links, corresponding to lifecycle portions which were not discovered yet, such as lateral movement or secondary compromise, or the potential attribution of a cyber attack. Our experimental results are promising to support the performance of the proposed framework.

The remainder of this paper is organized as follows. Section 2 describes the framework for extracting attack narratives from network traffic. Section 3 reports a case study based on a real dataset. Section 4 reviews related prior work. Section 5 concludes the paper and suggests future research directions.

2. Framework for Extracting Attack Narratives from Traffic Datasets

In this paper, we define an *attack narrative* as the set of attack signatures, that provides an overview of the attack and the attacker themselves when combined. To the best of our knowledge, we first coined this terminology in the literature. In what follows, we will describe the framework by which one can extract attack narratives from network traffic, and discuss the applications of the attack narratives.

2.1. The framework

Figure 1 highlights the framework. The raw data may be represented in the PCAP form. In order to deal with a large amount of data, we propose reducing, if not eliminating, the white noise that is often contained in the raw data because attacks are relatively rare. In order to create end-to-end attack narratives from attack signatures, we propose mapping attack signatures to Mandiant's Attack Lifecycle Model to classify the attack signatures, while it is possible to use other models for this purpose (e.g., Lockheed Martin's Cyber Kill Chain [3]). We treat each step of the lifecycle as a feature (or marker), ultimately leading to the formation of an attack narrative. Each feature is a part of a whole profile. Just like the rifling left on a bullet can conclusively link it to a firearm, we can link an actor to a cyber attack by the methodology and techniques used at each phase of the attack. We elaborate two core components of the framework below.

Using Mandiant's Attack Lifecycle Model to classify attack signatures. As highlighted in Figure 2, the model consists of multiple components corresponding to multiple phases of attacks. During the *Initial Reconnaissance (Initial Recon)* phase of the lifecycle, we can look for port scans and port sweeps. We also expect a high volume of nonspecific traffic in an attempt to hide the probing traffic. There are other types of techniques, such as brute force attempts of usernames and passwords, which one could argue as probing. However, we consider them as a component of the *Initial Compromise* phase because of the aggressive nature of a brute force attack; as opposed to a reconnaissance scan which can be stealthier. Once those scans have finished, we then expect the attacker to begin an assault on the services they discovered. During Initial Compromise, or more specifically pre-initial compromise, we expect to see patterns that relate to brute force password attacks, SMB (Server Message Block) or NetBios attacks, directory traversal, and other such attacks. For example, attempting to traverse web directories to find *cmd.exe* or the */etc/passwd* file is an obvious attempt to compromise a system. We would consider this a classic attack signature. A key feature of this phase is that the attacker tries to get into a system. However, it is only for the initial system; if the attacker is attempting to move from one compromised system in an organization to another, this

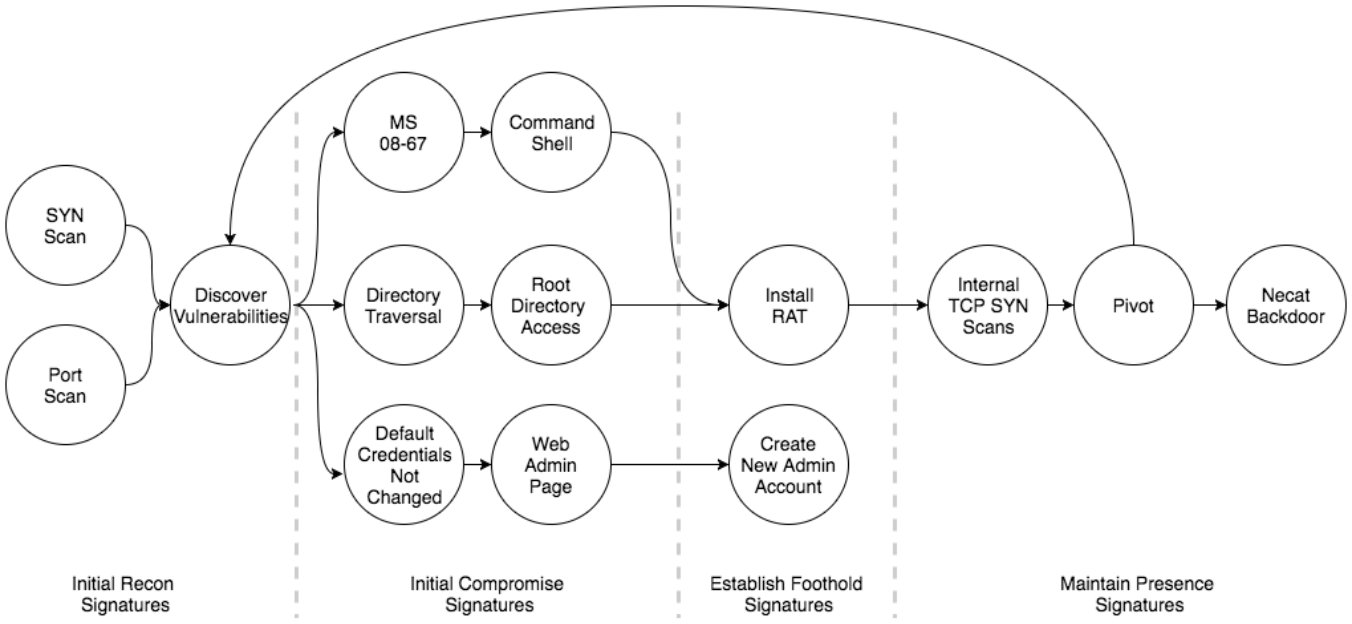

Figure 3. A hypothetical attack narrative comprises of attack signatures that are mapped to the Mandiant's Attack Lifecycle Model. In this narrative, the attacker favors SYN scanning while under the cover of other port scans, and then uses MS 08-67, directory web traversal, and unchanged default credentials gain entry into the system. The attacker then places RATs and new admin credentials before pivoting to another system, while leaving himself access with a netcat backdoor.

falls under the *Move Laterally* phase. Therefore all attack signatures in this phase should be about compromising as we have described, rather than later phases, which are about persisting, hiding, and spreading.

The *Establish Foothold* phase is about creating a presence in the compromised system such that the attacker can conduct their objective. Therefore, attack signatures in this phase will consist of techniques such as creating backdoors and installing rootkits. Due to the fact that SSH traffic is encrypted, none of our attack signatures will include this traffic due to the inability to decipher it. This is inherent to network-based extraction of attack narratives and network-based defense in general. Therefore, our attack signatures will concentrate on protocols that allow clear text to be sent over the network such as HTTP, TELNET and FTP.

The *Maintain Presence* phase includes attack signatures such as the exfiltration of data or command and control communications. We will target machines that are using unusual ports. For example, a DNS server most likely will not use 443 to call out to another DNS server. This would be a suspicious behavior, and a likely candidate for an attack signature.

Formulating attack narratives from attack signatures. Once attack signatures are collected, we begin to formulate an attack narrative based on the attack lifecycle model. For example, in the hypothetical attack narrative constructed in Figure 3, the attacker favors SYN scanning while under the cover of the other noisy port scans. After getting the result of their SYN scans, the attacker then favors the use of MS 08-67, Directory Traversal, and unchanged default credentials to gain entry to a system. Once the attacker is in the system, the attacker uses both a remote access Trojan (RAT), and the creation of new administrator accounts to establish a firm foothold on the compromised system. We can infer this preference because of their continued use in other compromised systems as the attacker pivots through the network. Finally, the attacker installs a basic netcat backdoor to retain access to the system.

2.2. The applications of attack narratives

The proposed method for extracting attack narratives is novel and has high applicability in practice. In addition to identifying malicious activities and helping defenders understand the cyber attack situational awareness, it provides cyber attack attribution for more applicability. For example, our proposed attack narratives can be used in court to help establish attribution, or demonstrate that other nations are conducting attacks. Unlike other existing cyber attack defense mechanisms aiming at improving IPS/IDS detection rates, by leveraging the Mandiant Attack Lifecycle Model, new investigators can be trained to help recognize how far an attacker is towards its attack goal. For example, attack narratives can be converted into fingerprints and then compared against other datasets to aid cyber attack attribution.

3. Case Study

In this section, we report a case study of using the framework to extract attack narratives from the 2015 National Collegiate Cyber Defense Competition (NCCDC) dataset

obtained from https://www.predict.org/. NCCDC is a college level competition aimed at providing undergraduate and graduate students with a real world cyber security experience. Students (Blue Teams) are given enterprise level network architectures (virtual and/or physical) and are expected to defend them against attackers (i.e. Red Team), while keeping hosted services (such as e-commerce) up and available to customers. The Blue Team finalists are selected nationally from round-robin elimination style competitions, which take place all over the United States from January to April of each year (online qualifiers, state regionals, and then finally the national round). Each year over 150 schools compete in NCCDC; the top ten teams represent the best of those schools. The Red Team at NCCDC is arguably made up of some of the foremost penetration testers in the United States. Team members consist of employees from nationally ranked cybersecurity firms, federal agencies, and well-known DoD contracting firms. This competition is considered by industry and government as prestigious, and as a result private companies and federal agencies often recruit students directly at the competition. It is not often that there is a public dataset that puts quality competitors against each other in a format that mimics an enterprise business infrastructure. One could even argue that there is no other competition that can get a dataset this close to a real world cyber-attack and defense scenario. Therefore, there is enormous potential to study this dataset for various security purposes, including the extraction of attack narratives.

3.1. Dataset

The dataset consists of approximately 2,300 files of network traffic captures, each of the files is on average 500 MB in size. The entire dataset is approximately 1.2 TB. The composition of the network traffic consists of, but is not limited to: scanning and attack traffic generated by the Red Team, background traffic from network packet generators, scoring engine traffic (to verify if services are up), real customer traffic (e.g. website usage, e-mail traffic, DNS requests), physical device traffic (VoIP phones, switches, routers, ICS/SCADA), and Internet and Intranet traffic from Blue Team members. According to predict.org, all teams and their various subnets are connected to each other by a star topology through a single backbone switch. The PCAP files were captured over a SPAN port from that backbone switch using TCPDUMP with no DNS resolution.

3.2. Reducing white noise

We considered packet grepping techniques such as *ngrep* and *tshark* filters. We also considered the feasibility of pushing the dataset through SNORT as a means of signature extraction. We classified these methodologies as deep packet inspection techniques, and they took several hours to parse through the data, and the output did little to help gain insight into the attack. It was simply too much information for an adequate analysis. Ultimately, we preprocessed the PCAP files and converted the output to text using a bash script that parsed through all 2,300 files using *tshark*. The conversion from PCAP to text took less than two hours to complete. This output was then passed through with another bash script that parsed through the newly created text for known attack signatures with traditional grep. This use of standard grepping over ngrep took fifteen minutes for the signature detection. These operations were completed on an Open Compute Project Server with two Intel Xeon X5650 CPUs at 2.67GHz, 24GB of memory, and running Ubuntu Server 16.04 Server.

3.3. Extracting attack signatures

After having preprocessed the network capture from the 2015 NCCDC dataset, we began parsing through the text files with grep to look for well-known attack signatures. In our first attempt, we focused on the distributed computing environment remote procedure call (DCE/RPC) to the vulnerable SRVSVC V3.0 found in unpatched Windows XP machines. We found five separate attempts to call this vulnerable service. These calls are highly indicative of an intent to gain NT Authority\SYSTEM access to Windows computers. Interestingly, we found these calls to a domain controller running Windows Server 2008, and a single workstation, which was running Windows 7. This is atypical of this type of exploit.

3.4. Formulating attack narratives from attack signatures

With respect to the 2015 NCCDC dataset, we expect the attacker to loosely follow the Mandiant Attack Lifecycle Model. We will look for attack signatures in the network traffic that map directly to particular phases of the lifecycle. We focus on four specific parts of the lifecycle when searching for attack signatures: Initial Recon, Initial Compromise, Establish Foothold, and Maintain Presence (see Figure 2).

In particular, it is challenging to verify if an exploit has been successful. We wanted to know what the network traffic looked like for a successful implementation of this attack. In order to capture the network traffic of the exploit, we set up two VMs in a sandboxed network segment. One VM was running an unpatched version of Windows XP SP2 while the other VM was running the Kali Linux 2.0 distribution. We used Wireshark on the Kali VM to monitor and capture the network traffic. We set up the exploit to call back to the Kali VM using a reverse windows shell, and then executed the exploit. Immediately after the execution of the exploit we noted the establishment of a new TCP three-way handshake shortly following the DCE/RPC call. This is highly suggestive that the exploit was successful and the new protocol is the result of the reverse TCP shell we configured. We then applied this methodology to search for a successful implementation of the attack in our dataset. We noted no establishment of a new TCP flow. This is either because the attack was unsuccessful, or it was simply the execution of a probing technique to check for the vulnerability in the service.

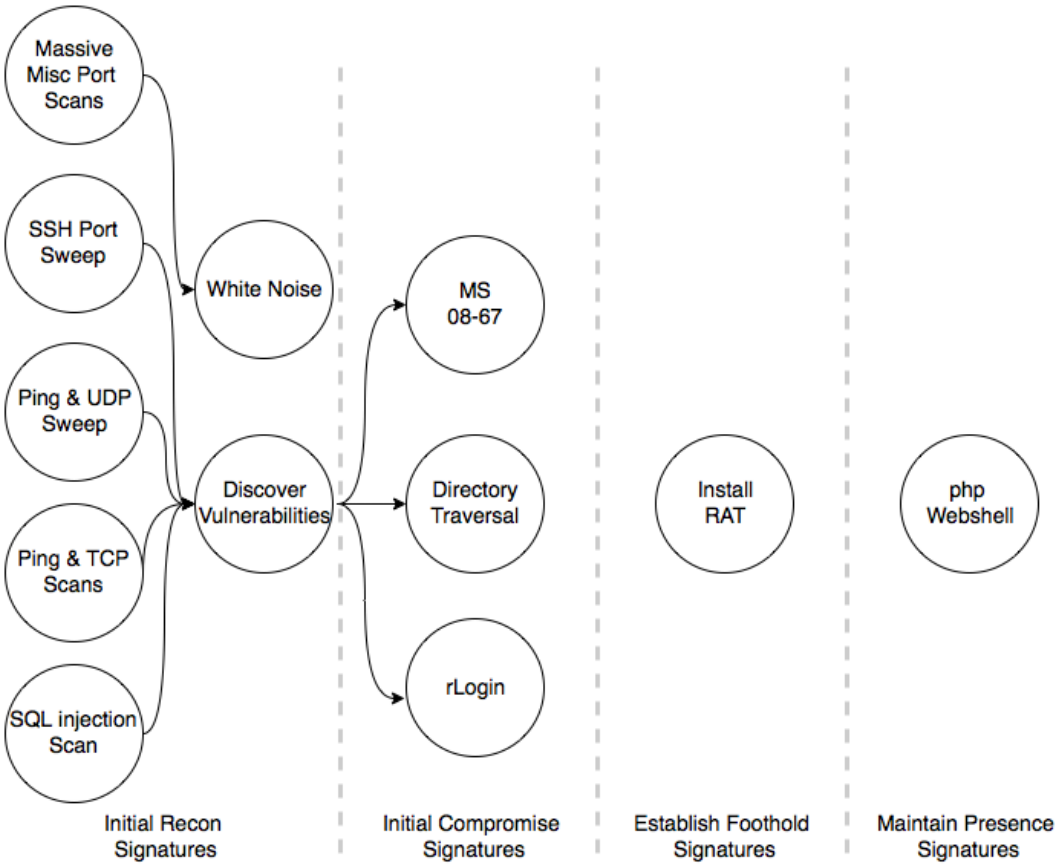

Figure 4. An attack narrative from 2015 NCCDC dataset against a participating Blue Team, which is anonymized for presentation in the present paper. In this narrative we see a coordinated and elaborate effort across many different IPs to perform a network reconnaissance. Network traffic also reveals that RATs and PHP webshells were used to establish footholds and maintain presence respectively. However, there are three unsuccessful attempts at Initial Compromise that cannot be conclusively linked to a node in the Establish Foothold or Maintain Presence signatures (note the lack of edges from the Initial Compromise phase to any subsequent phase).

We then searched for any reconnaissance efforts by the suspect IP by grepping for packets with lengths of 60 bytes or less. Our search revealed that the IP address involved used RST packets from a ping sweep to enumerate the victim network, and then used a focused UDP sweep to determine the services of the discovered machines. As we continued our search for packets with lengths of 60 bytes or less, we noticed more refined and complicated types of reconnaissance sweeps from at least 15 different IP addresses and all within minutes of each other. We categorized these scans and continued searching for egress traffic from these IPs to establish signatures for the other phases of the attack lifecycle. Ingress signatures revealed that there had been inbound calls from remote access Trojans, and php webshells by a few of the IPs discovered in the reconnaissance phase of the lifecycle.

We parsed our findings to create the attack narrative in Figure 4. From this narrative, we observed a coordinated and elaborated effort across many different IPs to perform a network reconnaissance. The reconnaissance was considered extremely loud by network bandwidth standards, approaching 47 MBs of PCAP traffic. This however seems to be by design because most of the traffic was generated by port scans with no apparent reason other than to conceal the IP addresses of a few attack machines. These conceal IP addresses then went on to compromise, establish footholds, and maintain presence throughout the network using the same techniques at each phase of the lifecycle.

We can infer that the attacker is sophisticated enough to script a complex opening salvo of very specific reconnaissance mechanisms. Timestamps indicate that this salvo is scripted and not executed in real time. We also note that information gathered from the initial reconnaissance scans are then used to narrow the focus of secondary scanning. This type of salvo would constitute the principle fingerprint of this attacker. It is specific enough to assign attribution to this attacker if it was seen again in other datasets.

4. Related Work

We mapped attack signatures to Mandiant's Attack Lifecycle Model [2], which is from a defender's point of view.

A closely related model is the Cyber Kill Chain [3], which is from an attacker's point of view and includes the following seven attack phases: reconnaissance, weaponization, delivery, exploitation, installation, command and control, and actions on objectives. We chose to use the Attack Lifecycle Model [2] because the study of attack narratives reflects a defender's point of view more than an attacker's. Nevertheless, it remains to be investigated whether or not it is advantageous to map attack signatures to the Cyber Kill Chain to piece together attack narratives.

Attack signatures [4], [5], [6] have been studied although there have been little study on the problem of recovering attack narratives by *connecting the dots*. The problem of piecing together attack signatures to formulate attack narratives is related to the problem of *alert correlation*. Alert correlation aims to detect multi-step attacks [7], [8], reduce redundant or unnecessary alerts [9], [10], and reconstruct attack scenarios or strategies based on low-level attack prerequisites and consequences [11], [12]. The key difference between them is that alert correlation takes a *bottom-up* approach to recover some high-level attack scenarios, whereas attack narratives takes a *top-down* approach by using an attack lifecycle model to guide the formulation of attack narratives. Indeed, due to scalability reasons, our case study did not even use alerts generated by intrusion detection systems such as SNORT. Nevertheless, it is an interesting future work to investigate how these two approaches may be incorporated into a single one.

To the best of our knowledge, this is the first academic study aiming to model and extract attack narratives, despite that the term *attack narrative* has been mentioned in media reports [13], [14]. Our long-term goal is to rigorously define and model attack narratives and build tools to automate the process of extracting attack narratives.

5. Conclusion

In this paper, we have presented a framework for creating attack narratives from network traffic. We have reported a case study that uses the framework to guide the extraction of attack narratives from a NCCDC dataset. Preliminary results show that the framework is effective.

The present study can be extended in several directions: (1) we will further extend to fully test our attack narratives, especially to commercial products such as SNORT; (2) our framework requires a decent amount of manual work, which may not desirable. Our future direction will work on how to utilize techniques such as machine learning to help create attack narratives, so that new attacks can be documented in a timely manner; (3) it is critical to establish ground truth in a dataset so that all techniques for finding attack narratives can be measured and compared fairly and without bias. Our future work will aim to quantify and measure the trustworthiness of attack narratives. We believe our framework can be extended to achieve this, and that this dataset has enough potential to be tested on various products.

Acknowledgments

We thank the PREDICT project for providing the dataset. This study was approved by IRB. This research was in part supported by the Department of Defense (DoD) through the office of the Assistant Secretary of Defense for Research and Engineering (ASD (R&E)) and ARO Grant #W911NF-13-1-0141. The views and opinions of the author(s) do not reflect those of the DoD, ASD (R&E), or Army.

References

[1] Symantec, "Attack signatures," https://www.symantec.com/security_response/attacksignatures/, (Accessed July 08, 2016).

[2] Mandiant, "Apt1 report," https://www.fireeye.com/content/dam/fireeyewww/services/pdfs/mandiant-apt1-report.pdf, February 16, 2013 (Accessed July 08, 2016).

[3] Lockheed Martin, "Cyber kill chain," http://cyber.lockheedmartin.com/solutions/cyber-kill-chain, (Accessed July 08, 2016).

[4] Z. Liang and R. Sekar, "Fast and automated generation of attack signatures: A basis for building self-protecting servers," in *Proceedings of the 12th ACM Conference on Computer and Communications Security*, ser. CCS '05, 2005, pp. 213–222.

[5] ——, "Automatic generation of buffer overflow attack signatures: An approach based on program behavior models," in *Proceedings of the 21st Annual Computer Security Applications Conference*, ser. ACSAC '05, 2005, pp. 215–224.

[6] Y. Afek, A. Bremler-Barr, and S. L. Feibish, "Automated signature extraction for high volume attacks," in *Architectures for Networking and Communications Systems (ANCS), 2013 ACM/IEEE Symposium on*, Oct 2013, pp. 147–156.

[7] F. Valeur, G. Vigna, C. Kruegel, and R. A. Kemmerer, "A comprehensive approach to intrusion detection alert correlation," *IEEE Trans. Dependable Secur. Comput.*, vol. 1, no. 3, pp. 146–169, Jul. 2004.

[8] O. B. Fredj, "A realistic graph-based alert correlation system," *Sec. and Commun. Netw.*, vol. 8, no. 15, pp. 2477–2493, Oct. 2015.

[9] F. Cuppens and A. Miège, "Alert correlation in a cooperative intrusion detection framework," in *Proceedings of the 2002 IEEE Symposium on Security and Privacy*, ser. SP '02, 2002, pp. 202–.

[10] T. H. Nguyen, J. Luo, and H. W. Njogu, "An efficient approach to reduce alerts generated by multiple ids products," *Netw.*, vol. 24, no. 3, pp. 153–180, May 2014.

[11] P. Ning, Y. Cui, and D. S. Reeves, "Constructing attack scenarios through correlation of intrusion alerts," in *Proceedings of the 9th ACM Conference on Computer and Communications Security*, ser. CCS '02, 2002, pp. 245–254.

[12] P. Ning and D. Xu, "Learning attack strategies from intrusion alerts," in *Proceedings of the 10th ACM Conference on Computer and Communications Security*, ser. CCS '03, 2003, pp. 200–209.

[13] V. Haran, "Adopting deception to control the attack narrative," http://www.bankinfosecurity.asia/interviews/adopting-deception-to-control-attack-narrative-i-3241, July 12, 2016 (Accessed August 22, 2016).

[14] Grecs, "Threat data vs. threat intelligence," https://www.novainfosec.com/2016/03/03/threat-data-vs-threat-intelligence/, March 3, 2016 (Accessed August 22, 2016).