A New Intrusion Detection Benchmarking System

Richard Zuech, Taghi M. Khoshgoftaar, Naeem Seliya,
Maryam M. Najafabadi, and Clifford Kemp

Florida Atlantic University, Boca Raton, FL 33431
rzuech@fau.edu, khoshgof@fau.edu, nseliya@gmail.com, mmousaarabna2013@fau.edu, cliffkempfl@gmail.com

Abstract

This paper presents a new quality network-based dataset for the purpose of intrusion detection system (IDS) evaluation, and is referred to as the IRSC (Indian River State College) dataset. Network flows and full packet capture (FPC) data are collected creating two types of datasets. The IRSC dataset represents a real-world network that gives us the advantage of collecting actual normal and attack traffic data reflecting a real-world environment. The attack portion of the traffic contains both controlled attacks (which are intentional attacks generated by our team) and uncontrolled attacks (which are real attacks on the IRSC network not created by our team). One main goal is to produce a reliable dataset with normal and attack traffic that is realistic and meets real world criteria. Another major goal is to produce a systematic process which would allow others to generate high quality IDS evaluation datasets. Our work's main contributions are that we have both accurate labeling through the inclusion of controlled attacks, and also realistic data by including real-world attacks.

1. Introduction

The increased dependence on network-based computing has put much emphasis on data-centric investigations for effective Intrusion Detection Systems (IDSs). Intrusion detection plays a critical role in network defense by aiding network security personnel in alerting them to malicious behaviors, i.e. intrusions, attacks, and anomalies. An effective Network Intrusion Detection System (NIDS) would yield low false-alarm (or false-positive) rates and high intrusion-detection (or true-positive) rates, where a false-alarm occurs when normal traffic data is misidentified as malicious.

An important component for developing an effective NIDS solution is the need for a good benchmark IDS evaluation dataset. However, an important problem faced by the NIDS research community is the lack of benchmark IDS evaluation datasets that are relatively-current, contain real-world representative traffic data, and are collected from modern complex networks. The focus of this paper is to present a unique and novel approach to generating and collecting NIDS evaluation datasets that are practical and representative of modern-day attacks and networks.

The presented approach for generating benchmark IDS datasets reflect the characteristics of a good dataset as observed by the authors, as well as by other researchers (Shiravi et al. 2012). Denoted for the institution (Indian River State College, Florida, USA) that was the source of the network traffic data, the IRSC dataset reflects network traffic from a real-world networking environment. The framework for data generation and collection includes workstations and servers with modern operating systems, facilitating a quality framework for multiple capture methods including both full packet captures (which completely stores all the data from network traffic) and network flows (which store an aggregated unidirectional summary of network traffic between two networked devices). The IRSC network framework consists of various segments, including wired network, wireless network, and virtual network segments.

2. Related Work

In this section, we present a discussion for some of the existing IDS evaluation datasets, largely from a perspective of demonstrating the need for IDS dataset(s) that reflect the characteristics of a good dataset as stated earlier.

The DARPA datasets were constructed for network security analysis purposes through data-centric intrusion detection (McHugh 2000), and it is criticized for issues associated with the artificial injection of normal and attack traffic data types into the generated datasets. While IDS researchers have examined other problems with the DARPA datasets, the key problems are that they do not reflect real-world network traffic data, contain irregularities in the dataset such as the absence of false positives, are outdated for effective IDS evaluation of modern networks both in terms of attack types and network infrastructure, and lack actual attack data records.

The KDD Cup 1999 dataset was created by processing the tcpdump portions of the 1998 DARPA Intrusion Detection System (IDS) evaluation dataset (Tavallaee et al. 2009). However, despite its extensive use as a benchmark IDS dataset, several experts have identified critical problems with its use for evaluating current NIDS solutions. This dataset has now become outdated for researchers because its network traffic patterns of attacks and normal data are largely irrelevant in the context of modern production computer networks. The network traffic records were generated through simulations performed on a military networking environment that consisted of normal background traffic and attack traffic data records and where the two types are merged together in a simulated environment. This dataset has a large number of redundant records that led to skewed testing results. Moreover, this dataset lacks the very important characteristic of reflecting real-world traffic patterns in modern-day complex computer networks.

The CDX (Sangster et al. 2009) dataset demonstrates that network warfare competitions can be utilized to generate modern-day labeled datasets. Their results indicate that network warfare competitions can be used for generating attack-only traffic to test IDS alert rules. A weakness of current network warfare games, such as the CDX, is the lack of the volume and diversity of traffic normally seen in production networks.

The Kyoto dataset (Song et al. 2011) was first generated in 2009 using the process of capturing and analyzing network traffic that is directed through honeypots, which have both advantages and disadvantages. On the positive side, there is no need for manual labeling and anonymizing, while on the negative side is the limited view of the network traffic, i.e., in this case experts can only observe attacks directed at the honeypots and not those directed at other systems in the target network infrastructure. Another limitation is that all the traffic from honeypots is attack traffic data and there is no normal traffic. There are no false positives since all traffic is malicious which represents behavior that does not reflect in the real-world. However, false positives are very much a part of networks today and represent an area that is consistently researched to improve on minimizing the number of alerts. In the Kyoto dataset, normal traffic is simulated repeatedly during the attacks and producing only DNS and mail traffic data, which is not reflective of real-world "normal" traffic.

The UNB (University of New Brunswick) ISCX 2012 dataset (Shiravi et al. 2012) represents dynamically generated data which reflects network traffic and intrusions. Various multi-stage attacks scenarios are carried out to stream the anomalous segment of the dataset. Normal background traffic is provided by executing user-profiles that were synthetically generated at random synchronized times creating profile-based user behavior. However, this lacks realistic Internet background noise and the overall normal traffic is not comparable to a real live network. While the authors present a good guideline for generating useful IDS evaluation datasets, their approach does not include unknown real and live attacks which are only observed through evaluating a live production network. These datasets represent a good sampling of the most widely used datasets by the research community, and their collective weaknesses indicate a clear need for additional high quality datasets.

3. Data Collection Process

In our study, the framework setup for collecting the FPC and network flows is based on the Security Onion (SO) (Bejtlich 2013) Linux-based distribution system for recording FPC with Snort (Sanders and Smith 2014) and network flows with the Silk "System for Internet-Level Knowledge" tool (Shimeall et al. 2010). SO collects full packet captures and also creates network flows by extracting them from its FPC data using Silk. An additional source of network flows is produced from a Cisco firewall that sends network data in a commonly used standard called NetFlow[1] v9, and this is sent to a Linux machine where it is stored (refer to Section 3.2 for further details). The framework of the network consists of multiple segments referred to as VLANs which encompass different parts of the network such as a wireless subnet, the internal network for end user client computers, various servers, and the demilitarized zone.

3.1 Full Packet Captures (FPC)

For packet captures, our study focused on two logs: Snort's full packet captures and its alert data. Snort's full packet captures continue on a 24 hour basis, and saves the full packet data to a directory with daily logs. The alert logs are generated when packets activate one of the detection rules. These logs are used when analyzing uncontrolled attacks, and are typically the main source of data for analyzing those types of attacks.

Prior to labeling our data records, it was necessary to perform a data-cleansing process. For the purpose of improving the reliability of the controlled attack session to prevent data loss from dropped packets, tcpdump[2] is used in addition to Snort for capturing network traffic during the

[1] http://www.cisco.com/c/en/us/products/ios-nx-os-software/netflow-version-9/index.html

[2] http://www.tcpdump.org/

entire controlled attack period. Later, Wireshark[3] is used as an analyzer to compare the captures from tcpdump versus the Snort logs, and it compares both of their packet capture results to find any missing packets from either packet capture tool. If any missing packets are found from either capture tool, they will be merged into a new single packet capture file (any duplicate packets in the newly merged file are discarded).

3.2 Network Flow Data Captures

Network flows collect summarized and aggregated network traffic between two networked devices, and they contain much less data than their full packet capture counterparts. Network flows can provide a higher level of abstraction to more quickly ascertain anomalies in network traffic. In our study the network flow data is collected from two sources: (1) The Cisco firewall and Linux machine for collecting NetFlow v9 data, (2) The Silk program which captures IPFIX standard network flow data in real-time by extracting it from FPC data.

The NetFlow v9 data is collected and logged by tools from the NFdump[4] tool suite which is installed on a designated Linux machine. In our experiment, NetFlow v9 data is being captured on a 24-hour basis and is used for collection of all network traffic. The second type of network flow being generated in our experiment is called IPFIX network flow data, and it is collected in real-time by Silk by extracting the session data from the 24-hour Snort FPC logs, i.e. Silk is used to extract network flows from the Snort log files. The extracted data is then written to files as a Silk record through a command line interface and can be used as an IDS evaluation dataset (or even for real-time analysis of the uncontrolled attacks).

4. Labeling

Two of the main contributions of our work are discussed in this section to ensure high quality in correctly labeling and classifying attacks. One contribution is assuring accurate attack labeling by including controlled attacks, and another contribution is utilizing expert-based labeling of live production data records that were obtained from actual uncontrolled attacks.

4.1 Manual Labeling for Uncontrolled Attacks

Identifying uncontrolled attacks is a highly valuable feature of our work. Uncontrolled attack data is collected and extracted from our daily FPCs in Snort log files. Any attack data that may be in a Snort log file is unknown in the sense that Snort could have mislabeled the data record. The targets for these attacks are directed toward actual live machines ranging from production servers to all workstations. The source of these attacks could be internal or external and could take place at any time of the day due to its uncontrolled characteristic. The Snort log, which represents a full packet capture for that day, can be read by tcpdump and Wireshark, or converted to network flow data with NFdump or Silk. Combining tools used for full packet capture and network flow data analysis is an important step to identifying uncontrolled attacks.

Labeling the uncontrolled attacks in the captured network data is done in two steps. In the first step, Snort alerts are used to analyze the FPC data and label some attacks. Then in the second step, additional analysis is conducted on the network flows in order to complement the first step and detect any attacks which were not detected in the first step. This manual inspection is applied to our IRSC case study dataset to allow us to detect stealthier attacks, and thus we can improve our labeling accuracy for the portion of the dataset which was not generated from the controlled attacks.

4.2 Labeling Controlled Attacks

Labeling full packet captures for controlled attacks is a relatively simple process because the attacks are originating from machines with IP addresses that are known. As mentioned previously, full packet captures for controlled attacks originate from two sources, tcpdump and Snort logs. After the cleansing process by Wireshark is used to find and merge any missing packets from both sources, then the file can easily be exported to a format like CSV and additional preprocessing steps can be performed with tools like Microsoft Excel. It is easy to label the controlled attacks simply by identifying the source IPs from the computers which were used to conduct the controlled attack. The file is then saved and ready for data mining and data analytics.

The NFdump and Silk suites come with a set of tools to analyze data. As mentioned previously, Snort collects FPCs on a daily basis in our case study. Silk can take any existing FPC file and convert it to a Silk network flow record. Labeling network flow data for controlled attacks follows the same procedure as for full packet data controlled attacks. The IP address is known, and we apply a filter to the flow records to filter out those IP addresses. This can be done with both NFdump and Silk. Silk can convert the Silk flow record to CSV format (Shimeall et al. 2010). The flow records with those IP addresses can then be marked as Attack and the rest as Normal.

[3] http://www.wireshark.org/

[4] http://nfdump.sourceforge.net/

5. Discussion

The overall goals of this study were to create high-quality IDS evaluation datasets than can be used for data mining and machine learning analysis, and to also create a standardized process (that can be replicated) for generating quality datasets. When compared to the IDS evaluation datasets mentioned in the Related Works section, we believe that our goals have been achieved. We were able to make improvements by focusing on the important characteristics of the process for generating quality datasets. The quality of our controlled attacks was improved by performing them in a live network. Attacks that take place in a live network are an invaluable enhancement to background noise that is truly realistic and is an improvement from the normal traffic created by the ISCX and Kyoto datasets. On the same token, CDX suffered from extremely poor synthetic data using only SMTP, DNS, or simple HTTP services. Real normal data helps to promote a more realistic portion of false negatives and false positives within the dataset.

ISCX provided respectable controlled attack methods but these attacks were not executed in a live network. Our network is not isolated. There is no need to use artificial insertions that replay or script normal background traffic like what is done in ISCX. Normal and attack data from our experiment is collected from multiple segments of a real production network, as compared to Kyoto and CDX. Kyoto also lacked realistic false positives due to such poor normal synthetic data.

6. Conclusion

The lack of publicly available IDS evaluation datasets that are reliable is a fundamental concern for researchers investigating data mining solutions with NIDSs. The quality of benchmark datasets is critical. Current public datasets for assessing IDS systems are limited because of the time, effort, and privacy assurance difficulties associated with generating them.

Our data is collected in a live production network that reflects a "real-world environment" that helps to ensure the collection of quality datasets in terms of both attack and normal network traffic. In the controlled attacks carried out by qualified experts in our study, there is a possibility that an actual attack could be taking place. Manual inspection by an expert will find these actual attacks as well. Random live attacks from uncontrolled internal or external sources are an excellent representation of real-world network activities due to the truly random nature in their attacks. Uncontrolled attacks are manually analyzed and labeled by qualified personnel.

Our work includes various capture methods for Full Packet and network flow data captures that have the benefit of verifying the integrity of the captures against each other as well as labeling uncontrolled attacks. FPC and Network flow data are important in networks today as they complement each other when analyzing and labeling the uncontrolled attacks. Network flow data is collected in two formats, IPFIX and NetFlow v9, giving us a chance to compare them in the future. Collecting two types of network traffic provides the benefit of creating two types of datasets and is invaluable when assessing uncontrolled attacks.

Because the process itself we are proposing is a contribution which other researchers can benefit from, we will continue to enhance our capture and labeling process. Our goal is to make the collection of useful IDS evaluation datasets an on-going process and plan to maintain that feature by updating our collections every six months. We will adapt to evolving attack methods, and apply newer techniques to ensure a quality dataset generation process.

Our future work will include making our dataset available to the public. We also plan to add honeypots to our data collection, and we intend to add additional attacks that are targeted at mobile users and network routers.

References

Bejtlich R. 2013. *The Practice of Network Security Monitoring*, 29-31. No Starch Press.

Chappell L. 2013. *Wireshark 101 Essential Skills for Network Analysis*, 161. Protocol Analysis Institute Publishing.

McHugh, J. 2000. Testing intrusion detection systems: a critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by Lincoln Laboratory. *ACM transactions on Information and system Security*, *3*(4): 262-294.

Sanders C. and Smith J. 2014. *Applied Network Security Monitoring* Syngress 82-83. Publishing, Third Edition.

Sangster, B., O'Connor, T. J., Cook, T., Fanelli, R., Dean, E., Morrell, C., and Conti, G. J. 2009. Toward Instrumenting Network Warfare Competitions to Generate Labeled Datasets. In *CSET*

Shimeall T., Faber S., DeShon M., and Kompanek A. 2010. Using SiLK for Network Traffic Analysis, Carnegie Mellon University, Pittsburgh, PA.

Shiravi, A., Shiravi, H., Tavallaee, M., & Ghorbani, A. A. 2012. Toward developing a systematic approach to generate benchmark datasets for intrusion detection. *Computers & Security*, *31*(3): 357-374.

Song, J., Takakura, H., Okabe, Y., Eto, M., Inoue, D., & Nakao, K. 2011. Statistical analysis of honeypot data and building of Kyoto 2006+ dataset for NIDS evaluation. In Proceedings of the First Workshop on Building Analysis Datasets and Gathering Experience Returns for Security, 29-36. ACM.

Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. 2009. A detailed analysis of the KDD CUP 99 data set. In Proceedings of the Second IEEE Symposium on Computational Intelligence for Security and Defence Applications.