MACHINE LEARNING FOR COMPUTER SECURITY*

HAMMAD AFZALI

CONTENTS

ABSTRACT

The rise of sophisticated cyber-crimes has made the area of computer security quite interesting. In the past years, security specialists have tried to prevent computer systems from being compromised by attackers. As an illustration, machine learning has been used to identify malicious behavior particularly in the intrusion detection systems. Though machine learning techniques give a better understanding of various data collected from different complex systems, the majority of such algorithms face challenges in adversarial environments. Addressing this issue, has fueled a strong interest in the study of learning in adversarial environments has been increased. This paper firstly focuses on the role of learning in computer security applications in order to identify the security problems cane be solved with the help of learning methods. Secondly, we distinguish the explain the concept of secure learning to clarify the limitations of learning algorithms under different attacks.

* *This report is based on [1].*

[1] *Department of Computer Science, NJIT*

1 THE ROLE OF MACHINE LEARNING IN SECURITY

The rapid development of security exploits has increased the the necessity for security tools. According to Symantec security threat report [2] more than 317 million new pieces of malware created in 2014 which means nearly one million new threats have been sighted daily. Though, recent developments in the learning approaches have improved the abilities of security applications, there are some drawbacks in this area which need a deeper understanding of the security aspects of machine learning.

The early work on intrusion detection formulated intrusion detection as a data analysis problem in which a decision function is based on a model automatically derived from previous benign examples [3]. This model is based on the hypothesis that security violations can be detected by monitoring a system's audit records in order to find abnormal patterns of system usage. The model profiles the behavior of subjects with respect to objects in terms of metrics and statistical models, and rules for acquiring knowledge about this behavior and detecting anomalous behavior.

2 LEARNING BASED APPROACHES

In this section we briefly explain a couple f learning based approaches in the area of computer security, in particular, malware detection.

2.1 Anomaly-Based Approaches

The pioneer approach was introduced by [4] which present a method for recognizing anomalies in system calls' behaviors. This approach has proposed two stages to detect malwares. In the initialization phase, a scanner traces the normal behavior of standard UNIX systems by observing sequences of system calls which leads to build up a database of characteristic normal patterns. In the second stage, they scan new traces (during the run-time) that might contain abnormal behavior,looking for patterns not present in the normal database. Using natural immune systems, this methods analyzes system call traces aiming at detecting any violations.

Another system which takes advantages of decision tree algorithms has been developed by Laskov et. [5]. This approach is based on a geometric framework for unsupervised anomaly detection. In this framework, the data is mapped into a feature space, and anomalies are detected as the entries in sparsely populated regions. However, this method proposes a one-class Support Vector Machine (SVM) for typical IDS data features. The key idea of this "quarter-sphere" algorithm is to encompass the data with a hypersphere anchored at the center of mass of the data in feature space.

2.2 Supervised Learning

Machine learning techniques such as supervised classification and clustering have proved to be useful to various security problems. These tools group individual malware samples into malware families by executing malicious programs in a controlled environment and produce reports that summarize the program's actions. Bayer et. [6] is automated clustering techniques that tries to discard reports of samples and focus on novel, interesting threats.

The key point is to generalize the observed activity well enough to recognize related malwares.

Rieck et. [7] exploits shared patterns for classification of malware and proposes a method for learning and distinction of malicious activities. The main goal is to addressing the following questions:

- Does an unknown malware instance belong to a known malware family or does it constitute a novel malware strain?
- What behavioral features are discriminative for distinguishing instances of one malware family from those of other families?

To address the above questions, a three-step methodology for learning the behavior of malware from labeled samples has been developed.

1. Monitoring the collected malwares in a sandbox environment:
   Malware binaries are collected via honeypots and spam-traps, and malware family labels are generated by running an anti-virus tool on each binary. To find shared patterns, each binary is monitored in a sandbox environment and some operations, like opening an outgoing IRC connection or stopping a network service, are summarized into reports.
2. Training the classifier:
   In the second step, using learning techniques (SVM), a classifier is trained to analyze reports. A document is characterized by frequencies of contained strings. While a set of considered strings are features and this technique determine the number of occurrences of a given string. The frequency of the string acts as a measure of its importance in a report. However, computation of these measures might seem infeasible at a first glance, as the reports may contain arbitrary many strings. There are some efficient algorithms that exploit the sparsity of that exploit the sparsity of the features in a linear run-time complexity [8].
3. Ranking discriminative features of the behavior models for future decisions:
   The learning model determines weights for behavioral patterns encountered during the learning phase. By sorting these weights and considering the most prominent patterns, the characteristic features for each malware family are obtained for future decisions.

2.3 Rule Inference

There are several types of data mining algorithms which are useful for mining audit data and extracting intrusion detection rules.

- Classification: maps a data item into one of several predefined categories. Such algorithm outputs "classifiers," in the form of decision trees, rules, etc. If we divide the audit data to "normal" and "abnormal", then a classifier should learn to predict or label new unseen audit data.

- Link analysis: determines relations between fields in the database record. For example, the shell history of a user, can be identified as the normal usage profile.

- Sequence analysis: models sequential patterns. This method can discover what time-based sequence of audit events frequently occur to-

gether. As an instance, several per-host and per-service in the data can be considered denial-of service (DoS) attacks.

One of the famous inference methods which makes use of data mining is called MADAM ID[1] [9], a framework to compute activity patterns from system audit data and extracts predictive features from the patterns. Actually, MADAM ID firstly process audit data which is summarized into connection records containing a number of basic features, such as service and duration. Data mining programs are then applied to the processed records to compute the frequent patterns and additional features for the connection records. Classification programs, for example, RIPPER, then used to inductively learn the detection models. This process finally generates intrusion detection rules.

3 EVALUATION

Sommer and Paxson discussed several practical difficulties faced by learning-based intrusion detection systems [10]. Such problems can lead to a semantic gap between detection results and actual threats. We can divide them into to categories.

DATA ANALYSIS ISSUES Certain characteristics of security problems are atypical for classical learning methods and require the development of customized techniques. These characteristics include unbalanced data (attacks are very rare), unbalanced risk factors (low false positive rates are crucial), difficulties in obtaining labeled data for all security applications and the most crucial is that adversarial data manipulation is not addressed by classical machine learning methods. Actually, the enormous variability and non-stationarity of benign examples causes difficulties in the training phase.

FEASIBILITY ISSUES While each week millions of threats are reported, there might be a high cost of classification. Also there is difficulty to perform a sound evaluation of such systems.

4 BEST PRACTICES

4.1 Web Application Security

Considering the discussed limitations, learning-based methods in the general intrusion detection context need to include the semantics of the application. However, in certain applications, learning-based systems significantly outperform conventional depending on expert knowledge. For example, extreme diversity of web applications make it is impossible to generate signatures for specific attack patterns. But the learning systems can overcome this difficulty by automatically inferring models of benign network traffic. Such models can be used in various issues.

- detecting malicious web queries [11]
- detecting logical state violations in web applications [12]
- sanitization of web queries [13]

1 Mining Audit Data for Automated Models for Intrusion Detection

4.2 Dynamic Malware Analysis

Characterizing malwares includes analysis their executions in a sandbox using hierarchical clustering which lead to find known malware families and detection of novel malware strains.

4.3 Automatic signature generation (ASG)

Combination of machine learning and malware analysis can be applied to extraction of the frequency of features (e.g. by Naive Bayes learning). However, it is possible to decrease such system's abilities by increasing the false alarm rate. ASG is useful for detection of botnet communications[101], network protocol reverse engineering.

5 TRADITIONAL LEARNING APPROACHES: EVALUATION

5.1 Open Issues

Though the vast majority of security-related data can be handled using simple rule-based detection methods, a number of open issues arise in the field of machine learning for computer security. Actually, most algorithms are not powerful enough to handle cleverly novel threat samples. Regarding the the architecture of learning-based malware detection, integration of machine learning with security mechanism must deal with the below problems. It is notable changes in adversarial data patterns make periodic re-training of learning system is a necessary practice.

- Learning-based rule generation

  Although rule-inference algorithms are well-known in machine learning (e.g., RIPPER), they can be easily evaded by an adversary. Novel rule inference methods are needed to deal with adversarial data and detect anomalous events in the absence of label information. An example for advanced rule-oriented learning is automatic discovery of regular expression patterns for spam detection.

- Human-aided machine learning The supervisory role of security experts is essential for the success of learning methods in this domain. However, security expertise is not only the general knowledge about traditional binaries or multi-class categories of classical learning methods. So new techniques for interaction between the learning methods and the security experts need to be investigated.

- Machine-learning-aided knowledge discovery Security analyst's work can also be greatly facilitated by applying appropriate learning techniques. Such approaches would be quite beneficial comparing to manual analysis which is very time-consuming and requires profound expertise.

5.2 Future Applications

In the following, we briefly discuss several such applications that address recent security problems and establish a bridge between academic work and practical solutions.

- Detection of advanced persistent threats Such algorithms help to detect unknown attacks.
- Dynamic and continuous authentication As the authentication data (esp. passwords) can be lost or forgotten, a more flexible authentication scheme, can be proposed. This approach relies on the history of user's activities. Actually machine learning can use stored data to facilitate the authentication process.

  1-question driven authentication using previous user's activities (e.g. an authorized user should be able to answer some question about his/her web-browsing history) .

  2-generating secret questions for resetting forgotten passwords.

  3-continuous and incremental authentication during a session. It can transparently check if the user does normal activities.(e.g. when the users request additional privileges, we can infer if an attacker impersonate the user's role)
- Assisted malware analysis Some ML algorithms could accelerate the detection process by looking for specific patterns instead of search all possible patterns.
- Computer forensics ML can benefit forensic techniques (like image forensics, network traffic forensics) by sifting large volumes of forensic data and prioritizing the information.

6 SECURE LEARNING

6.1 Basic Concepts

While machine learning is a powerful tool for data analysis and processing, traditional machine learning methods were not designed to operate in the presence of adversaries. They are based on statistical assumptions about the distribution of the input data, and they rely on training data derived from the input data to construct models for analyses. However, adversaries may exploit these characteristics to disrupt analytic, cause it to fail, or do malicious activities that fail to be detected. Attacks have been widely against learning algorithms in the fields of intrusion detection and spam classification where learning methods were actively used. Two main strategies taken by attackers.

- poisoning Altering normal data model
- imitation/mimicry Insertion of a normal content into the target data. For example a transformation of packet payload to match a certain histogram of byte occurrences (polymorphic blending technique). However, finding an optimal blending for byte sequences is an NP-complete problem.

Some scholars believe that when a learning algorithm performs well in adversarial conditions, it can be an algorithm for secure learning. But the problem is how do we characterize the quality of a learning system and determine whether it satisfies the requirements for secure learning?

Regarding the type of the attacks, we can conclude learning algorithms have different behavior under adversarial noise, depending on the part of the data controlled by an attacker. For those ML algorithms which use Boolean functions, an attacker must alter the labels of nearly half of the training data to cause the incorrect classification of selected data points. So by controlling attributes of the data, the error rate can be nearly as high as the fraction of data under the attacker's control [56]. So we need to clearly emphasize the assumptions on the adversarial power (Threat Model Boundaries).

6.2 Open Issues

There are a couple of issues in the field of secure learning which need to be addressed.

- Formalization of Secure Learning

  Evaluating the worst-case case effect of an attack scenario When the attacker is able to manipulate the training data to mislead the learning algorithm.

- Defining new security-aware loss functions

  It is really important to measure the "damage" done to the estimator under the non-stationarity introduced by the adversary's contamination.

7 RESULTS

REFERENCES

[1] Anthony D Joseph, Pavel Laskov, Fabio Roli, J Doug Tygar, and Blaine Nelson. Machine learning methods for computer security (dagstuhl perspectives workshop 12371). *Dagstuhl Manifestos*, 3(1), 2013.

[2] Symantec Enterprise. 2015 internet security threat report. *Volume 20*, 2015.

[3] Dorothy E Denning. An intrusion-detection model. *Software Engineering, IEEE Transactions on*, (2):222–232, 1987.

[4] Stephanie Forrest, Steven A Hofmeyr, Aniln Somayaji, and Thomas A Longstaff. A sense of self for unix processes. pages 120–128, 1996.

[5] Pavel Laskov, Christin Schäfer, Igor Kotenko, and K-R Müller. Intrusion detection in unlabeled data with quarter-sphere support vector machines. *Praxis der Informationsverarbeitung und Kommunikation*, 27(4):228–236, 2004.

[6] Ulrich Bayer, Paolo Milani Comparetti, Clemens Hlauschek, Christopher Kruegel, and Engin Kirda. Scalable, behavior-based malware clustering. Citeseer.

[7] Konrad Rieck, Thorsten Holz, Carsten Willems, Patrick Düssel, and Pavel Laskov. Learning and classification of malware behavior. In *Detection of Intrusions and Malware, and Vulnerability Assessment*, pages 108–125. Springer, 2008.

[8] Konrad Rieck and Pavel Laskov. Linear-time computation of similarity measures for sequential data. *The Journal of Machine Learning Research*, 9:23–48, 2008.

[9] Wenke Lee and Salvatore J Stolfo. A framework for constructing features and models for intrusion detection systems. *ACM transactions on Information and system security (TiSSEC)*, 3(4):227–261, 2000.

[10] Robin Sommer and Vern Paxson. Outside the closed world: On using machine learning for network intrusion detection. In *Security and Privacy (SP), 2010 IEEE Symposium on*, pages 305–316. IEEE, 2010.

[11] Patrick Düssel, Christian Gehl, Pavel Laskov, and Konrad Rieck. Incorporation of application layer protocol syntax into anomaly detection. In *Information Systems Security*, pages 188–202. Springer, 2008.

[12] Marco Cova, Davide Balzarotti, Viktoria Felmetsger, and Giovanni Vigna. Swaddler: An approach for the anomaly-based detection of state violations in web applications. In *Recent Advances in Intrusion Detection*, pages 63–86. Springer, 2007.

[13] Tammo Krueger, Christian Gehl, Konrad Rieck, and Pavel Laskov. Tokdoc: A self-healing web application firewall. In *Proceedings of the 2010 ACM Symposium on Applied Computing*, pages 1846–1853. ACM, 2010.