From Artificial Intelligence to Security: Back and Forth

Islam Faisal

May 2018

Abstract

Artificial intelligence and specially machine learning has become pervasive in all fields of computer and computational sciences. Security engineering is one of those fields where applications of AI and ML have invaded the traditional rule-based paradigms. Applications of ML in security include intrusion detection, anti-viruses, and detection of denial of service attacks. Recently, there has been interest in using machine learning to compute on encrypted traffic for providing in-network security such as deep packet inspection.

The relation between security and ML is bilateral. Concepts from security and privacy are being used to protect users’ data when fed to an ML model as well as protecting the trade-secrets of ML models. With the current hype of interest in ML and decentralization, it is expected that this bilateral relation will keep growing ever more.

1 Introduction

1.1 Machine Learning

Machine learning is the field of computer science where computers learn to perform certain tasks. This learning process can be either:

- Supervised: Ground-truth labeled data are given to the algorithm to learn from.
- Unsupervised: The algorithm learns from the similarity and patterns in the data.
- Semi-Supervised: A hybrid of both strategies where part of the learning is done in an unsupervised manner and then supervised learning is used to leverage the accuracy.

With the rapid rise of graphics processing units (GPU), machine learning has evolved to include large models with billions of parameters, which is widely known in the literature as deep learning [1].
1.2 Why ML for Security?

The wide variety of security applications including Intrusion Detection Systems (IDS), Distributed Denial of Service (DoS), and Anti-virus (AV) software require enormous efforts when installed in large-scale networks that are common nowadays. Traditional rule-based methods are no longer efficient in detecting traffic anomalies and recent vulnerabilities and generalizing to unseen signatures.

1.3 Why Security for ML?

With the development of machine learning models and algorithms, and the wide deployment of such models in the wild cloud, concerns over privacy have notably emerged. Users, specially the tech-savvy of them, and organizations, specially sensitive ones, would like to have provable security and privacy guarantees that their data are not accessed and not only depend on the trust of the cloud service provider. To resolve this, there must exist either (i) computing services that are able to compute on encrypted/blinded data, or (ii) decentralization that dissolves the trust on the computing service provider and provides privacy guarantees.

Outline The rest of the paper is organized as follows: section 2 and 3 discusses ML and AI applications in intrusion detection systems and denial of service detection and mitigation respectively. Section 4 goes on to introduce the use of machine learning in anti-virus/malware systems. Section 5 introduces then the usage of machine learning to compute on encrypted traffic. Section 6 analyzes the other side of the relation, where security and privacy concepts are used to secure the machine learning services over the cloud.

2 Intrusion Detection Systems

Intrusion Detection Systems (IDS) is a vital component of any practical network in the modern world. Several methods are being used for IDSs which is tasked with detecting intrusion into networks by attackers. However, with the recent rise in the world of machine learning, research has been directed towards the use of the powerful toolbox of machine learning algorithms to build IDSs.

2.1 Datasets

One of the challenging problems for any model-based learning technique is the availability of training data. In the field of intrusion detection systems, the scarcity of data is justified by the fear of releasing vulnerabilities publicly as well as being hosted in-house at companies as trade-secrets. This leads the data available publicly to be relatively small or out-dated or both. The Kyoto 2006+ dataset for NIDS evaluation [2] is one of the major datasets used for training and evaluation of network intrusion detection systems. The dataset was built
on 3 years of real traffic data from November 2006 to August 2009 which are obtained from diverse types of honeypots.

**Synthetic Data**  ID2T [3, 4] is a tool that can be used to generate synthetic yet realistic. This tool mainly overcomes the problem of scarcity of data in IDS training. ID2T can be used to generate labeled network datasets that contain user defined synthetic attacks. The ability of the toolkit to create realistic synthetic attacks of high quality and low bias is discussed in [3]

### 2.2 Models

Based on the training method, intrusion detection systems (IDS) are categorized into three major categories:

**Supervised learning**  Models are trained with actual intrusion attacks as well as normal traffic. The model learns to classify on-the-fly a given payload if it is normal or intrusive.

**Unsupervised learning**  Actually, this is the most common in IDS. The system is trained to detect abnormal network behavior or pattern. This relies on the assumptions that attacks usually are not similar to the mainstream network behavior normal users inflect.

**Semi-supervised learning**  A hybrid is used from supervised and anomaly detection methods, which can also be applied in hierarchy so that the unsupervised mechanism serve as a backup line of defense after the supervised one. An example of semi-supervised intrusion detection systems is [5].

### 3 Features

Features that are relevant to intrusion detection and possibly network malicious behaviour in general are for example (some are from [2]):

- Source IP address
- Packet length
- Duration between packets
- Similarity between consecutive packets
- Deviation from traditional traffic
- Connection Duration
- Service
- Transmission and Error Rates
4 DoS Detection and Mitigation

Denial of Service (DoS) refers to the type of attack during which an attacker floods either from a single stream or from distributed nodes a service for the sake of deteriorating its availability. It is essential for any organization, service provider, and network administrator to pay much attention to this type of attack. The security mechanisms that needs to be provided include:

- DoS Detection: Detecting (distributed) denial of service attacks in real-time.
- DoS Mitigation: The contingency plans to be followed once a DoS attack is detected.

The network security solutions for denial of service detection and mitigation are very similar in fashion to ones used in intrusion detection explained in the previous section. One major difference is that in distributed attacks, it is essential to account for the whole server stream in general and not only a single connection.

5 Anti-Virus/Malware Systems

Detecting viruses, malwares, and adware is perhaps the oldest security problem, even since the historical Trojan horse. Traditionally, this was achieved by adding signatures of bad applications, and traffic to a list of known viruses that gets periodically updated as more and more malicious software is discovered.

With time, viruses and malware became very complicated, self-changing, and sometimes unpredictable. This requires the use of statistical methods to determine with probability whether a certain application is vulnerable.

For example, let’s have a closer look on how Symantec’s Advanced Machine Learning engine works on endpoints [6]:

- An initial model is trained with known viruses and shipped to users
- Users install this model and periodically feeds it with telemetry data from users’ devices
- The model iteratively improves based on the collected telemetry data and discovered viruses
- The updated model is shipped to the users.

It is essential to note that in many cases, it is better to use a hybrid of rule-based and signature-based methods with model-based ones. This increases the alarm rate of the system.

Machine learning models can learn code similarities between malicious software instead of memorizing code signatures by heart. This allows for a wider range of detection and possibly detecting new viruses. If equipped with an anomaly detection system, an ML-backed anti-virus can really perform very well in front of new attacks and viruses.
The features used in building the models typically include static and dynamic features. Static features involve the appearance of certain assembly-level sequences, file sizes, and checksums, or publisher. Dynamic features are extracted during run-time (possibly in sandbox or quarantine) and include memory utilization, interaction with other processes, and CPU consumption.

6 ML on Encrypted Data

With the rise of cloud computing, machine learning-based solutions for all domains including security are widely available. In the cloud, there is an established dilemma of code and data privacy. Computing services vendors don’t want to reveal their trade secrets and want to keep their algorithms private. Organizations who want to use these services want to maintain the privacy of data. This has led to the emergence of the field of Homomorphic Encryption where computation can be done on encrypted data.

6.1 Example from DPI

Deep Packet Inspection (DPI) refers to inspecting network traffic and data in the wild to find viruses, malware, or adware, or by governments for censorship and surveillance purposes. In unencrypted traffic, the task is a piece of cake, just insert a man-in-the-middle between the client and the server. In HTTPS, the problem is a little bit different. Inserting this man-in-the-middle, requires further certifying this man-in-the-middle by digital certificates. Sometimes, this can be forged by organizations to “protect” their employees from viruses, malware, or bad traffic.

Ideally, a good DPI middlebox should be able to perform as well as one having access to plain-text traffic while actually not seeing the traffic. BlindBox [7], which was introduced in 2015 is a research effort to introduce a middlebox that can perform efficiently deep packet inspection by inserting it to observe a TLS connection. This compromises the need for in-network security solutions and the fact that TLS was designed to perform all tasks on the endpoints and nothing in between.

7 The other way around..

7.1 Differential Privacy

In a world where data analysis is done extensively, users’ data may be revealed unknowingly. Consider these two examples:

Unknownly Revealing Patient Data A hospital that encourages researchers to help them make patients’ life better periodically publishes anonymized patient records without revealing patients’ names. However, the hospital publishes patients’ ages and hospitalization dates. Imagine if John Doe is the only
42-aged patient who was hospitalized between October and December. This can lead to the revelation of the identity of the patient. An easy-to-catch remedy is to add noise to the data. For example, publishing age groups instead of exact age.

**NetFlix Data Breach**  NetFlix publishes anonymized user data for a competition they oversee on building recommender systems. Privacy researchers in 2006 [8] were able to link users’ published data to reviews and ratings from IMDB that are available publicly which resulted in clearly a privacy breach. 

*Differential Privacy* is the field that is tasked with preventing all of these possible breaches and leakage of data. “Differential privacy describes a promise, made by a data holder, curator, to a data subject: You will not be affected, adversely or otherwise, by allowing your data to be used in any study or analysis, no matter what other studies, data sets, or information sources, are available” [9].

All companies now compete to build privacy-preserving data analysis tools, of which is the famous Google RAPPOR tool 1.

### 7.2 Decentralized systems for ML

Performing machine learning, and AI in a distributed system with privacy guarantees is a challenging task. Engima 2 [10] “is a blockchain-based protocol that uses groundbreaking privacy technologies to enable scalable, end-to-end decentralized applications.” Enigma also allows for executing “secret” contracts which is a variant of smart contracts. The introduction of these types of systems along with ones that have homomorphic capabilities will build trust between data holders and computation providers and give more fluidity to the digital services in the cloud.

### 8 Conclusion

As we have seen, security and AI intertwine in harmony. With the promise of more computing resources and services in the cloud, this interaction will grow ever more.

Applying machine learning in security requires further collaboration between academics and industry professionals to bridge the gap between the academics’ talent in building promising algorithms and the companies’ resourcefulness in acquiring massive data from real-world traffic.

Securing machine learning applications is the tax that security specialists pay for the amazing power provided by machine learning applications in solving common problems in the information security domain.

---

1[https://github.com/google/rappor](https://github.com/google/rappor)

2[https://enigma.co/](https://enigma.co/)
References


