Open Problems in the Security of Learning

Marco Barreno* Peter L. Bartlett*† Fuching Jack Chi* Anthony D. Joseph*
Blaine Nelson* Benjamin I. P. Rubinstein* Udam Saini* J. D. Tygar*‡

{barreno,bartlett,iamjack,adj,nelsonb,benr,usaini,tygar}@cs.berkeley.edu

*Computer Science Division †Department of Statistics ‡School of Information
University of California, Berkeley
Berkeley, CA 94720

ABSTRACT

Machine learning has become a valuable tool for detecting and preventing malicious activity. However, as more applications employ machine learning techniques in adversarial decision-making situations, increasingly powerful attacks become possible against machine learning systems. In this paper, we present three broad research directions towards the end of developing truly secure learning. First, we suggest that finding *bounds on adversarial influence* is important to understand the limits of what an attacker can and cannot do to a learning system. Second, we investigate *the value of adversarial capabilities*—the success of an attack depends largely on what types of information and influence the attacker has. Finally, we propose directions in *technologies for secure learning* and suggest lines of investigation into secure techniques for learning in adversarial environments. We intend this paper to foster discussion about the security of machine learning, and we believe that the research directions we propose represent the most important directions to pursue in the quest for secure learning.

Categories and Subject Descriptors

D.4.6 [**Security and Protection**]; G.3 [**Probability and Statistics**]: Robust Regression; H.1.1 [**Systems and Information Theory**]: Value of Information; I.5.1 [**Models**]: Statistical; I.5.2 [**Design Methodology**]

General Terms

Security, Theory

Keywords

Adversarial Learning, Computer Security, Machine Learning, Secure Learning, Security Metrics

AISec'08, October 27, 2008, Alexandria, Virginia, USA.

1. INTRODUCTION

1.1 Motivation

Current research in applications of machine learning lies at the confluence of two growing trends. First, statistical machine learning has entered the mainstream as a broadly useful technique for building applications. In adaptive systems, machine learning enjoys several advantages over handcrafted rules and other approaches: it can infer hidden patterns in data, it can adapt quickly to new signals and behaviors, and it can provide statistical soundness. Second, the need to protect systems against malicious adversaries continues to increase across computing applications. Rising levels of hostile behavior have plagued application domains such as email, web search, pay-per-click advertisements, file sharing, instant messaging, mobile phone communications, and others. As the motivation for attacks becomes increasingly fiscal [8], attackers employ more sophisticated methods and the computing landscape grows ever more treacherous.

One result of this meeting of trends is that machine learning techniques have become an invaluable tool in protecting system security. From spam filtering to malware detection to fast attack response to many other applications, machine learning is now an essential component of computer security.

But the inclusion of machine learning in a system must be done carefully to prevent the learning component itself from succumbing to attack. A growing body of literature shows that attackers can successfully attack machine learning systems, both in general [2, 6] and in specific application domains such as automatic signature generation [4, 5, 22], intrusion detection systems [7, 27], and email spam filtering [16, 19, 30]. It is imperative to ensure that learning is successful despite such attacks—in other words, to achieve *secure learning.*

1.2 Contributions

In this paper, we suggest three broad research directions leading towards the goal of *secure learning:*

1. **Bounds on adversarial influence** (Section 2)

   A set of results proving tight bounds on errors and adversarial effort would be extremely valuable for understanding the behavior of machine learning systems under attack.

2. **The value of adversarial capabilities** (Section 3)

   It is crucial to understand what capabilities an attacker has and how they relate to the difficulty of perform-

ing and preventing attacks. This direction focuses on the quality of adversarial influence, while the previous direction considers the amount of influence on the learner.

3. **Technologies for secure learning** (Section 4)

   The final piece of the secure learning puzzle is to bring defensive techniques to maturity to protect machine learning systems against interference by attackers.

Each of these directions is fundamental to the end of learning without disruptive influence from an adversary, and each encompasses a variety of important questions. We present several questions in each section along with discussion of the challenges that make the research direction important and difficult.

2. RESEARCH DIRECTION: BOUNDS ON ADVERSARIAL INFLUENCE

A significant research challenge is the discovery of fundamental limits on the effectiveness of learning attacks. The focus of this direction is on proving upper and lower bounds on an adversary's influence on a learner. In a learning attack the adversary, assumed here to have full knowledge of the learner's algorithm and state, achieves influence by contaminating the learner's training and/or test data. We consider two basic varieties of influence: modification of the learner's state to the detriment of the learner's statistical performance, and the adversary's discovery of the learner's state or the learner's mechanism of adaptation.

2.1 Influence on Learner Performance

QUESTION 1. *Can we bound the minimum amount of adversarial effort or the maximum number of mistakes the learner will make in real-world domains such as anomaly detection, spam filtering, worm signature generation, phishing detection, and click fraud detection? Are matching lower error bounds possible in these cases?*

A number of relevant results on adversarial influence of learner performance have recently appeared. Nelson and Joseph [20] bound the minimum number of examples an adversary must submit to a hypersphere-based anomaly detector in order to shift the hypersphere's center. The authors establish this learner-specific lower bound on adversarial effort by proving an upper bound on an optimal adversary's effect on the learner's state. Such learner-specific upper bounds on error quantify a learner's robustness to malicious training data. Venkataraman et al. present lower bounds on the number of mistakes *any* supervised classifier must make when generating worm signatures in the presence of an adversary with full information and control [29]. Such learner-independent lower bounds demonstrate fundamental vulnerabilities faced by all learners, reflecting the generalization-vs-security trade-off inherent to adaptive approaches. Newsome et al. [22] describe attacks on the learning algorithms in the Polygraph [21] and Hamsa [14] automatic worm signature generation systems and they consider naive Bayes spam filtering. In addition to experimental validation, the authors present theoretical bounds on their attacks' influence.

While the bounds listed above explore theoretical analyses of several attacks on practical learning systems, many more attacks have enjoyed experimental validation. Lowd & Meek [16] and Wittel & Wu [30] propose good word attacks on statistical spam filters with the aim of putting spam into user inboxes. Nelson et al. [19] develop attacks for causing false positives against the statistical spam filter, formalized by Robinson [24], which is used in the SpamBayes (spambayes.sourceforge.net), BogoFilter (bogofilter.sourceforge.net), and SpamAssassin (spamassassin.apache.org) systems. Finally Rubinstein et al. [25] outline data poisoning attacks for increasing false negatives in principal components analysis (PCA), as used in network-wide volume anomaly detection by Lakhina et al. [13]. Theoretical analysis of these learning attacks, and attacks on learners used in other security-sensitive domains, would provide valuable insight into the fundamental limits of learner security.

2.2 Influence for Reverse Engineering State

The second form of adversarial influence on machine learners aims to reverse engineer the learner's state or algorithm by selecting data submitted to the learner.

Lowd and Meek have studied the problem of learner reverse engineering [15]. Given an attacker cost function, the authors analyze the complexity of finding a minimum-cost instance that is labeled negative by the learner. The notion of *ACRE-learnability* characterizes learners that can be reverse engineered with a polynomial number of queries. The following question asks about generalizations of ACRE-learnability results for linear classifiers on Boolean and real-valued features, under linear adversarial cost.

QUESTION 2. *Can we quantify the complexity of attacks on larger classes of classifiers or learners (e.g. regressors), and adversarial cost functions?*

Some types of classifiers may be naturally robust to reverse engineering.

QUESTION 3. *What classifiers, if any, are provably hard to reverse engineer? Are there combinatorial parameters of the concept class that characterize ACRE-learnability?*

3. RESEARCH DIRECTION: THE VALUE OF ADVERSARIAL CAPABILITIES

A crucial step in protecting against threats on machine learning systems is to understand the threat model in adversarial learning domains. The threat model can broadly be described as the attacker's *goals* and *capabilities*; capabilities can be capabilities of *information* or capabilities of *control*. This second general research direction proposes a more fine grained adversarial error analysis by focusing on the role of the adversary's capabilities. Such analysis should quantify the value of information and control available to the adversary for attacks against learning systems.

Adversarial information is the adversary's knowledge of the learning system and environment, such as the learner's features, the learning algorithm, the current decision function, the policy for training and retraining, and the benign data generation process. Similarly *adversarial control* is the extent of the attacker's control over the learning system's training and/or test data.

EXAMPLE. *In email spam filtering, relevant adversarial information may include the user's language, common types of email the user receives, which spam filter the user has, and the particular training corpus or distribution used to create the spam filter (or knowledge of a similar distribution). Adversarial control may include choosing the bodies of a fraction of emails (perhaps only spam), controlling email headers directly or indirectly, and controlling how the user receives messages. This control could be exerted over messages used for training or for run-time testing.*

EXAMPLE. *In network-wide traffic anomaly detection [13], adversarial information may include the network topology, routing tables, real-time traffic volumes along one or more links, historical traffic along one or more links, and the training policies of the anomaly detection system. Adversarial control may include controlling one or more links to give false traffic reports or compromising one or more routers to inject chaff into the network.*

EXAMPLE. *In the domain of phishing webpage detection, adversarial information may include user language and country, email client, web browser, financial institution, and employer. Adversarial control may include choosing the content and/or headers of the phishing emails and potentially influencing training datasets of known phishing sites, such as PhishTank [23].*

3.1 Identifying Adversarial Capabilities

QUESTION 4. *What are natural threat models for learners used in deployed systems? How secure are such learners in relation to these threat models, and in relation to adversarial information and control?*

The theoretical framework of attacks on machine learning systems of Barreno et al. [1, 2] describes the threat model in terms of a taxonomy of attacks. This taxonomy describes attackers' goals and whether the attacker can influence the learner's training data or test data; beyond this classification, the role of the adversary's capabilities has been relatively unexplored. Two of our previous experimental studies have touched on the effects of adversarial information and control: Nelson et al. [19] and Rubinstein et al. [25] show that learning attacks on email spam filtering and network-wide anomaly detection, respectively, are more effective when additional information or control is available to the adversary. Less progress has been made on understanding the fundamental roles of adversarial capabilities, however. Kearns and Li extend the Probably Approximately Correct (PAC) distribution-free learning framework to the setting where an adversary has control over a limited fraction of the training data [12]. They show that the largest fraction tolerable is $\epsilon/(1+\epsilon)$, for PAC learning with error ϵ.

3.2 Characterizing Tolerable Capabilities

QUESTION 5. *Which forms of adversarial control can a learner tolerate under full adversarial information? More generally, how can we characterize tolerable adversarial information and control?*

Understanding threat models in real-world systems is complemented by viewing the adversarial learning problem abstractly. In some cases the arms race of improving the learner with increasingly sophisticated defenses and launching ever more covert attacks can be bypassed by proving optimality. The above question aims to formalize our observation that assessing learner security across different domains can be fruitfully organized around *common* adversarial capabilities. We now consider one natural approach to this question, from the viewpoint of online learning [3].

In online learning, the learner and attacker take part in a game: after the learner chooses a strategy, nature selects the benign data and the adversary transforms this data, each acting with knowledge of previous decisions made. The learner repeatedly makes predictions and observes adversarially transformed data, then updates its state. The learner aims to minimize its cumulative loss.

The adversary's control can be represented as the set of transformations from which it can act. Adversarial information can be modeled by forcing the attacker to decide on its transformation based only on limited information about the data. We envision characterizations of tolerable adversarial information and control being expressed in terms of properties of these sets of transformations and information mappings. The following examples demonstrate these representations of information and control in the security-sensitive learning problems of spam filtering and network-wide traffic anomaly detection, respectively.

EXAMPLE. *In email spam filtering, the adversary may not have access to the learner's training corpus but instead to a close surrogate. For example the adversary may have access to corpora sampled from a distribution close to the true distribution, as defined by Kullback-Leibler divergence or total variation. Nelson et al. [19] propose attacks exploiting surrogate corpora drawn from an English dictionary and an Internet newsgroup. Regarding control, a spammer may transform the training corpus by injecting a certain number of arbitrarily constructed spam emails into the corpus.*

EXAMPLE. *In network-wide traffic anomaly detection, the adversary may have access to measurements of the source link or links incident to a compromised node. This corresponds to information mappings that project the link traffic matrix to a single column or small bounded number of columns. Similarly the attacker may inject chaff along a flow in the network, corresponding to transformations that contaminate a small bounded number of traffic volume features.*

The final ingredient to Question 5 is the measure of success of the learner. In online learning, regret (the learner's cumulative loss compared to the minimum achieved with hindsight by a set of simple experts) can in general be achieved that is $O(\sqrt{T})$ in the length T of the game. Thus for a game-theoretic analysis "tolerable" adversarial settings could correspond to the learner suffering only $O(\sqrt{T})$ regret. A related question is that of the trade-off between statistical generalization and security.

QUESTION 6. *What are the quantitative trade-offs between the learner's generalization performance on innocuous data, the learner's hypothesis class capacity, and the adversary's capabilities?*

In statistical learning theory, a learner's ability to generalize on innocuous data is characterized by the chosen hypothesis class' expressiveness, or *capacity*, such as the Vapnik-Chervonenkis dimension [28]. Over-fitting results from representations with surplus capacity, while insufficient capacity will lead to hypotheses that fail to capture the problem's underlying complexity. On the other hand, we speculate that overly flexible learners may be easier to attack with reduced adversarial capabilities.

For binary classification, a finer-grained analysis of Question 6's trade-off would consider the trade-off between false negatives and false positives. In the intrusion detection and spam filtering domains, for instance, a more robust learner that suffers a greater rate of false positives would not be acceptable to users. This trade-off can be quantified by considering an appropriate utility function for the learner.

3.3 Defender Information

QUESTION 7. *Are there attacks that can significantly affect the behavior of a learning system without being apparent to the defender? How can we judge whether an attack is apparent to the defender? Are some attacks provably covert? If so, can we bound their influence on statistical performance?*

In many security problems, attackers attempt to evade detection. An attacker has at least two good reasons for wanting to hide attack information from the defender. First, the attacker may want to prevent the defender from discovering that an attack is in progress because the attacker often gains benefit in proportion to the time before the defender notices and reacts to the attack. Second, the attacker may want to avoid traces that might reveal the source of the attack, either to avoid accountability or to make anticipating the next attack difficult. In the case of a learning system, this interaction is of greater importance as the adversary's goal is often to change the behavior of the learner, but that change is likely to be visible to the defender.

4. RESEARCH DIRECTION: TECHNOLOGIES FOR SECURE LEARNING

To provide a reliable and trustworthy system in a security sensitive environment, one can explicitly design that system with relevant security threats in mind—this practice should be applied to designing learners for use in such environments. Broadly, this entails the design of learning agents for a security sensitive environment that are resilient to adversarial contamination in the data.

QUESTION 8. *Is there a secure learning procedure that is resilient to attacks under a realistic threat model? How can we construct this procedure?*

One must first explicitly identify the anticipated threats against the learning system by constructing a threat model as discussed in Section 3. Then an algorithm is chosen to be robust against these threats by considering the adversary's possible actions, the learner's counter-actions, and the final outcomes for both the adversary and the learner. This interaction can be cast as a game between the adversary and the learner [1]. By solving such a game, one can design an algorithm to be robust against the anticipated security threats, although solutions to such games are not necessarily feasible. Finally, after choosing the learner, one must assess its limitations and vulnerabilities under the threat model.

In the remainder of this section, we identify promising directions for the design of adversarially resilient learners. We have identified three general techniques: detecting and removing malicious data in a training set, constructing learners robust to malicious data, and designing several learners to be difficult to attack as a group. We discuss the strengths and weaknesses of these techniques and suggest key open problems for each.

4.1 Detecting malicious training instances

A simple way to reduce the impact of contaminated data on a learning algorithm is to detect and remove the malicious instances. This technique allows one to make any learner more secure by simply filtering the data before training the learner. The primary challenge is to be able to accurately identify malicious data that could affect the learner in adverse ways.

One method for detecting abnormal data is the general technique of outlier detection (refer to, for example, Markou and Singh for a survey of the field [17]). Generally an outlier detection algorithm identifies characteristics of normal data by training solely on normal data. Any subsequent data that deviates too far from the identified characteristics is considered abnormal and is discarded. Using an outlier detector for identifying malicious data has the advantage of not requiring malicious samples during the training phase. However, it may be difficult to obtain a clean dataset for the initial training of the detector itself; it depends on whether the adversary's capabilities allow them to influence the normal data.

The technique of dataset scrubbing as a preprocessing step before training suggests a follow-up question to Question 6:

QUESTION 9. *What is the trade-off between the increase to security and decrease to learning rate resulting from removing outlier instances? Which data scrubbing techniques achieve a good trade-off?*

If we aggressively remove suspect data, there will be less data to train on, so the learner will require more input before it can adequately learn the target function. However, if we take a conservative approach to removing suspect data, it will be easier for an attacker to create malicious data that avoids removal.

4.2 Designing Security Sensitive Learners

A second method to defend against malicious data is to use learning algorithms designed to be robust against an adversarial threat.

EXAMPLE. *In their paper entitled Adversarial Classification, Dalvi et al. describe a spam classifier designed to be robust against spam emails designed to fool a naive Bayes classifier [6]. To do so, they construct a threat model in which the adversary modifies the email by changing individual features (words) in the message and incurs a cost for doing so. The authors use game theory to augment the original classifier so it can better detect emails that are modified optimally against the original learner.*

In general, solving games against an adversary is a difficult problem and can be computationally intractable. However,

the framework of robust statistics addresses the problem of outliers in data. This framework provides a number of tools and techniques to construct learners robust against security threats from adversarial contamination in training data. In the following, we provide a brief description of the robust statistics framework and motivate its use in secure learning. (Several books provide additional reading about robust statistics [10, 11, 18].)

In classical statistics, one assumes that all data is generated by a common model or distribution, but outliers defy that assumption. Robust statistics augments classical models by assuming that the data comes from two sources: a known distribution and an unknown adversarial distribution. Under this setting, robust variants exist for parameter estimation, testing, linear models, and other classic statistical techniques. A well-known example is that the median is robust to contamination whereas the mean is not.

The robust statistics framework also provides tools to assess robustness against contamination. Two especially useful tools are the breakdown point and the influence function. The *breakdown point* of a procedure determines what percent of the data can be contaminated before the adversary can arbitrarily control the procedure. The *influence function* of a procedure assesses how sensitive a procedure is to adversarial contamination. These tools allow one to compare the robustness of procedures and even design optimally robust procedures.

QUESTION 10. *The breakdown point and influence function are useful first-order measures of robustness against contamination, but can they be computed for practical learning procedures? Further, are there more fine-grained measures of a procedure's robustness that would provide a better understanding of its behavior under adversarial contamination?*

QUESTION 11. *If the adversarial contamination is assumed to be limited, can we use robustness measures such as the influence function to design a practical learner that is efficient and secure?*

Relatively few learning systems are designed explicitly with statistical robustness in mind. The breakdown point and influence function can provide quantitative measurements of robustness; designers of learning systems may be able to use these tools to improve the security of learners in security sensitive tasks. They can assess the vulnerability of existing learning systems to contamination and compare existing techniques as candidates for a particular task. We believe that these tools will be useful; the challenge remains to integrate them into learning for security sensitive domains and use them to design learners resilient to attacks.

4.3 Orthogonal experts

A final element of defenses for security-sensitive settings is addressed by the game-theoretic online learning setting described in Section 3.2. In this setting, the learner receives advice from a set of experts and makes a prediction by weighting the experts' advice based on their past performance. Techniques for learning within this framework have been developed to perform well with respect to the best expert in hindsight.

QUESTION 12. *How can one design a set of experts (learners) so that their aggregate is resilient to attacks in the on-line learning framework? Can this design itself be accomplished in an automated fashion?*

The ideal case is that even if the experts may be individually vulnerable, they are difficult to attack as a group. We informally refer to such a set of experts as being *orthogonal*. Orthogonal learners have several advantages in a security sensitive environment. They allow us to combine learners designed to capture different aspects of the task. These learners may use different feature sets and different learning algorithms to reduce common vulnerabilities; e.g., making them more difficult to reverse engineer. Finally, online prediction techniques are flexible, so we can improve existing experts or add new ones as new vulnerabilities in the system are identified.

EXAMPLE. *Spam filters such as SpamAssassin decide whether or not a message is a spam by combining predictions made by a set of rules—simple heuristics humans use to identify spam messages. Such a rule could be whether any words in the message are "Viagra", "Cialis" or any of their common obfuscations. Other rules may be designed to catch penny stock scams, fake watch retailers, and so forth. These rules capture different aspects of spam and while the individual rules are not perfect, together they filter messages with a high degree of accuracy. Moreover, when spammers change their tactics to avoid these rules, the rules can be reconfigured or new ones can be added to the filter. Thus, SpamAssassin is an example of a classifier that combines experts and its rules are orthogonal by design.*

The experts in the previous example are fixed rules rather than learners, but boosting shows that this can be done with learners as well.

EXAMPLE. *A popular learning technique known as boosting [26] exemplifies how a set of orthogonal learners could be constructed. In boosting techniques such as AdaBoost [9], weak learners are sequentially trained to improve classification performance by focusing on training instances that previous learners performed poorly on. In this way, the ensemble learner is composed of a set of orthogonal learners and generally has better performance than the individual learners.*

Boosting techniques exemplify how orthogonal learners can be constructed in an ensemble method although the security properties of boosting have not been fully explored.

To properly design orthogonal experts for secure learning, one must first assess the vulnerability of several candidate learners. With that analysis, one should then choose a base set of learners and sets of features for them to learn on. Finally, as the aggregate is used, one should identify new security threats and patch the learners appropriately. This patching could be done by adjusting the algorithms, changing their feature sets, or even adding new learners to the aggregate. Ideally, this process could itself be automated or learned.

5. CONCLUSION

We have presented a broad program of research with the goal of securing machine learning against attack. We identify three research directions as particularly important: finding bounds on adversarial influence, understanding the value of adversarial capabilities, and developing technologies for secure learning. We believe these directions represent the most significant steps to take towards truly secure learning, and we have discussed many open questions within each direction. We intend this paper to help focus research efforts within the area on the problems that are most important for the ultimate goal of securing machine learning against attacks by an adversary.

Acknowledgments

We would like to thank Ling Huang, Michael Jordan, Shinghon Lau, Phil Long, Satish Rao, Charles Sutton, Nina Taft, Anthony Tran, and Kai Xia for many fruitful discussions that have influenced our thinking about secure learning.

This work was supported in part by the Team for Research in Ubiquitous Secure Technology (TRUST), which receives support from the National Science Foundation (NSF award #CCF-0424422), the Air Force Office of Scientific Research (AFOSR #FA9550-06-1-0244), British Telecom, Cisco, ESCHER, Hewlett-Packard, IBM, iCAST, Intel, Microsoft, ORNL, Pirelli, Qualcomm, Sun, Symantec, Telecom Italia, and United Technologies; in part by California state Microelectronics Innovation and Computer Research Opportunities grants (MICRO ID#06-148 and #07-012) and Siemens; and in part by the cyber-DEfense Technology Experimental Research laboratory (DETERlab), which receives support from the Department of Homeland Security Homeland Security Advanced Research Projects Agency (HSARPA award #022412) and AFOSR (#FA9550-07-1-0501). The opinions expressed in this paper are solely those of the authors and do not necessarily reflect the opinions of any funding agency, the State of California, or the U.S. government.

6. REFERENCES

[1] Marco Barreno, Blaine Nelson, Anthony D. Joseph, and J. D. Tygar. The security of machine learning. Technical Report UCB/EECS-2008-43, EECS Department, University of California, Berkeley, April 2008.

[2] Marco Barreno, Blaine Nelson, Russell Sears, Anthony D. Joseph, and J. D. Tygar. Can machine learning be secure? In *Proceedings of the ACM Symposium on InformAtion, Computer, and Communications Security (ASIACCS'06)*, March 2006.

[3] Nicolò Cesa-Bianchi and Gábor Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006.

[4] Simon P. Chung and Aloysius K. Mok. Allergy attack against automatic signature generation. In *Recent Advances in Intrusion Detection (RAID)*, pages 61–80, 2006.

[5] Simon P. Chung and Aloysius K. Mok. Advanced allergy attacks: Does a corpus really help? In *Recent Advances in Intrusion Detection (RAID)*, pages 236–255, 2007.

[6] Nilesh Dalvi, Pedro Domingos, Mausam, Sumit Sanghai, and Deepak Verma. Adversarial classification. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 99–108, Seattle, WA, 2004. ACM Press.

[7] Prahlad Fogla and Wenke Lee. Evading network anomaly detection systems: Formal reasoning and practical techniques. In *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, pages 59–68, 2006.

[8] Jason Franklin, Vern Paxson, Adrian Perrig, and Stefan Savage. An inquiry into the nature and causes of the wealth of internet miscreants. In *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, 2007.

[9] Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.

[10] Frank R. Hampel, Elvezio M. Ronchetti, Peter J. Rousseeuw, and Werner A. Stahel. *Robust Statistics: The Approach Based on Influence Functions*. Probability and Mathematical Statistics. John Wiley and Sons, 1986.

[11] Peter J. Huber. *Robust Statistics*. John Wiley and Sons, 1981.

[12] Michael Kearns and Ming Li. Learning in the presence of malicious errors. *SIAM Journal on Computing*, 22:807–837, 1993.

[13] Anukool Lakhina, Mark Crovella, and Christophe Diot. Diagnosing network-wide traffic anomalies. In *Proc. SIGCOMM '04*, pages 219–230, 2004.

[14] Zhichun Li, Manan Sanghi, Yan Chen, Ming-Yang Kao, and Brian Chavez. Hamsa: fast signature generation for zero-day polymorphic worms with provable attack resilience. In *IEEE Symposium on Security and Privacy*, 2006.

[15] Daniel Lowd and Christopher Meek. Adversarial learning. In *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 641–647, 2005.

[16] Daniel Lowd and Christopher Meek. Good word attacks on statistical spam filters. In *Proceedings of the Second Conference on Email and Anti-Spam (CEAS)*, 2005.

[17] Markos Markou and Sameer Singh. Novelty detection: a review–part 1: statistical approaches. *Signal Processing*, 83(12):2481–2497, December 2003.

[18] Ricardo A. Maronna, Douglas R. Martin, and Victor J. Yohai. *Robust Statistics: Theory and Methods*. John Wiley and Sons, New York, 2006.

[19] Blaine Nelson, Marco Barreno, Fuching Jack Chi, Anthony D. Joseph, Benjamin I. P. Rubinstein, Udam Saini, Charles Sutton, J. D. Tygar, and Kai Xia. Exploiting machine learning to subvert your spam filter. In *Proceedings of the First Workshop on Large-scale Exploits and Emerging Threats (LEET)*, 2008.

[20] Blaine Nelson and Anthony D. Joseph. Bounding an attack's complexity for a simple learning model. In *Proceedings of the First Workshop on Tackling Computer Systems Problems with Machine Learning Techniques (SysML)*, 2006.

[21] James Newsome, Brad Karp, and Dawn Song. Polygraph: Automatically generating signatures for polymorphic worms. In *Proceedings of the IEEE Symposium on Security and Privacy*, pages 226–241, May 2005.

[22] James Newsome, Brad Karp, and Dawn Song. Paragraph: Thwarting signature learning by training maliciously. In *Proceedings of the 9th International Symposium on Recent Advances in Intrusion Detection (RAID 2006)*, September 2006.

[23] PhishTank. `http://www.phishtank.com`.

[24] Gary Robinson. A statistical approach to the spam problem. *Linux Journal*, March 2003.

[25] Benjamin I. P. Rubinstein, Blaine Nelson, Ling Huang, Anthony D. Joseph, Shing-hon Lau, Nina Taft, and J. D. Tygar. Compromising PCA-based anomaly detectors for network-wide traffic. Technical report UCB/EECS-2008-73, UC Berkeley, May 2008.

[26] Robert E. Schapire. A brief introduction to boosting. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI '99)*, pages 1401–1406, 1999.

[27] Kymie M. C. Tan, Kevin S. Killourhy, and Roy A. Maxion. Undermining an anomaly-based intrusion detection system using common exploits. In *Recent Advances in Intrusion Detection (RAID)*, pages 54–73, 2002.

[28] Vladimir N. Vapnik and Alexey Y. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and its Applications*, 16(2):264–280, 1971.

[29] Shobha Venkataraman, Avrim Blum, and Dawn Song. Limits of learning-based signature generation with adversaries. In *Proceedings of the 15th Annual Network & Distributed System Security Symposium*, 2008.

[30] Gregory L. Wittel and S. Felix Wu. On attacking statistical spam filters. In *Proceedings of the First Conference on Email and Anti-Spam (CEAS)*, 2004.