2012 IEEE International Conference on Systems, Man, and Cybernetics
October 14-17, 2012, COEX, Seoul, Korea

Predictability-Oriented Defense Against Adaptive Adversaries

Richard Colbaugh
Sandia National Laboratories
Albuquerque, NM USA
colbaugh@comcast.net

Kristin Glass
New Mexico Institute of Mining and Technology
Socorro, NM USA
kglass@icasa.nmt.edu

***Abstract*—There are substantial potential benefits to considering *predictability* when designing defenses against adaptive adversaries, including increasing the ability of defense systems to predict new attacker behavior and reducing the capacity of adversaries to anticipate defensive actions. This paper adopts such a perspective, leveraging the coevolutionary relationship between attackers and defenders to derive methods for predicting and countering attacks and for limiting the extent to which adversaries can learn about defense strategies. The proposed approach combines game theory with machine learning to model adversary adaptation in the learner's feature space, thereby producing classes of predictive and "moving target" defenses which are scientifically-grounded and applicable to problems of real-world scale and complexity. Case studies with large cyber security datasets demonstrate that the proposed algorithms outperform gold-standard techniques, offering effective and robust defense against evolving adversaries.**

***Keywords*—**-predictive defense, moving target defense, game theory, machine learning, adaptive adversaries, cyber security.

I. INTRODUCTION

Adaptive adversaries are a principal concern in many security domains, including cyber defense, border security, counterterrorism, and crime prevention [e.g. 1-3]. Consequently, there is great interest in developing defenses which maintain their effectiveness despite evolving adversary strategies and tactics. A potentially powerful approach to pursuing such goals is to explicitly consider system *predictability,* for instance in order to design defenses which are able to anticipate adversary behavior and/or decrease their own predictability. Studies that employ predictability assessment in a security context include [4,5].

The coevolving "arms race" between Spammers and Spam filters provides an illustrative example of the phenomenon of interest [e.g. 6,7]. Spam filter designers would like to produce filters that work well against both present and future Spam, and one way to accomplish this goal is to develop techniques for predicting the way Spammers will adapt to currently-deployed filters and to account for these expected adaptations during the filter design process. Spammers, on the other hand, are motivated to "reverse-engineer" existing Spam filters as quickly as possible, so they can generate Spam which circumvents these filters. Spam filter developers are therefore interested in both sides of the predictability question: they wish to construct filters that can predict (and defeat) new Spammer techniques while remaining unpredictable themselves. Many other security problems involve adaptive adversaries and coevolutionary dynamics, and we propose that valuable insights can be obtained by examining these dynamics through the lens of predictability; Spam is merely a simple, familiar example of such systems.

Because predictability-based defense design includes strategic considerations, it is natural to approach this design problem as a game [8], in which defense attempts to predict and counter adversary behaviors while reducing its own predictability. Unfortunately, previous attempts to apply game-theoretic methods to adversary defense [e.g. 9-15] have encountered a number of challenges, and we mention two that have been especially daunting. First, the set of possible attacker actions is typically very large in real-world settings, and because the complexity of most game models increases exponentially with the number of actions available to the players, this has often made these models intractable in practice. And second, it has proved difficult to derive models that capture evolving attacker behavior in any but the most idealized situations.

In this paper we overcome these challenges by developing our game-based models for attack-defend interaction within a machine learning (ML) framework [16], enabling the design of robust defenses for practical applications. We formulate the defense task as one of behavior classification, in which innocent and malicious activities are to be distinguished, and assume only limited information is available regarding prior attacker behavior or attack attributes. The defense's classifiers model attacker actions in ML *feature space,* that is, in the space of variables the ML algorithms use for learning and decision-making. Formulating attack prediction/defense synthesis in this "compressed" and abstract space enables derivation of algorithms that can be applied to practical, large-scale problems.

The first of the proposed defense systems explicitly attempts to predict and counter adversary adaptation as a means of providing effective defense against both current and future attacks. A key step in the approach is modeling the way attackers *adapt* their behaviors rather than modeling the behaviors themselves. Crucially, the proposed approach seeks to design optimal defenses for evolving attacks, rather than to predict new attacks perfectly, and therefore enjoys robust performance in the presence of (inevitable) prediction errors. To permit the performance of this predictive defense method to be evaluated, we have assembled for this investigation a large collection of Spam and non-Spam emails reflecting the evolution of Spammer tactics over an eight year period. A case study with this

978-1-4673-1714-6/12/$31.00 ©2012 IEEE

dataset demonstrates that the proposed defense significantly outperforms a gold-standard Spam filter.

An important consideration when applying classifier-based defense techniques, even predictive ones, is the extent to which adversaries can reverse-engineer the learning algorithm and use this knowledge to circumvent the defense. The goal of the second proposed defense is thus to reduce defense system predictability and increase the difficulty of the adversary's reverse-engineering task. We adopt a "moving target" (MT) perspective, in which the defense presents a dynamic posture to the adversaries as a way of increasing the adversaries' uncertainty concerning defense operation [17]. By leveraging recent advances in the theory of repeated, incomplete information games [18,19], we derive a simple MT defense procedure which can be shown to be optimal for an important class of adversarial dynamics; interestingly, the optimal MT schedule can be specified independently of the details of the adversaries' strategies. The efficacy of the proposed MT defense is evaluated via case studies with the set of Spam and non-Spam emails mentioned above and also with a well-known publicly-available network intrusion dataset. These tests reveal that the MT defense substantially outperforms well-tuned static classifiers against adaptive adversaries.

II. Predictive Defense

A. Problem Formulation

There are significant potential benefits to developing *predictive* methods of defending against adaptive adversaries, in which opponents' evolving strategies are anticipated and these insights are employed to counter novel attacks. This section considers the following concrete instantiation of the predictive defense problem: given some history of attacker actions, design a defense system which performs well against both current and future attacks. It is reasonable to expect that concepts and techniques from game theory might be helpful in understanding adversary adaptation, and indeed such approaches have been explored in a variety of domains [e.g. 9-15]. However, as indicated in the Introduction, these investigations have encountered scalability and complexity challenges which have limited their practical utility. In this section we address these challenges by deriving our game-based model within an ML framework, enabling effective defense in realistic settings. (See [20] for a general discussion of the value of combining behavioral modeling with data mining algorithms for discovery and prediction applications.)

We approach the task of countering adversarial behavior as an ML classification problem, in which the objective is to distinguish innocent and malicious activity. Each instance of activity is represented as a feature vector $x \in \Re^{|F|}$, where entry x_i of x is the value of feature i for this instance and F is the set of instance features. In what follows, F is a set of "reduced" features, obtained by projecting measured feature vectors into a lower-dimensional space. While feature reduction is standard practice in ML [16], we show below that *aggressive* reduction allows us to efficiently manage the complexity of our game models. Behavior instances x belong to one of two classes: positive/malicious and negative/innocent (generalizing to more than two behavior classes is straightforward [16]). The goal is to learn a vector $w \in \Re^{|F|}$ such that classifier orient = sign($w^T x$) accurately estimates the class of behavior x, returning +1 (−1) for malicious (innocent) activity.

As indicated above, it is useful to assess the predictability of a phenomenon before attempting to predict its evolution; for example, such an analysis permits identification of measurables which possess predictive power [21]. There has been limited theoretical work assessing predictability of adversarial dynamics, but existing studies suggest attack-defend coevolution often generates predictable dynamics. For instance, although [22] finds that certain player strategies lead to chaos in a simple repeated game, [20] shows that large sets of player strategies and repeated games exhibit predictable adversarial dynamics. Here we supplement this theoretical work by conducting an empirical investigation of predictability, and select as our case study a cyber security problem – Spam filtering – which possesses attributes that are representative of many adversarial domains.

To conduct this investigation, we first obtained a large collection of emails from various publicly-available sources for the period 1999-2006, and added to this corpus a set of Spam emails acquired from B. Guenter's Spam trap for the same time period. Following standard practice, each email is modeled as a "bag of words" feature vector $x \in \Re^{|F|}$, where the entries of x are the frequencies with which the words in vocabulary F appear in the message. The resulting dataset consists of ~128,000 emails composed of more than 250,000 features. We extracted from this collection of Spam and non-Spam emails the set of messages sent during the 30 month period between January 2001 and July 2003 (email in other periods exhibit very similar evolutionary dynamics). Finally, the dimension of the email feature space was reduced via a singular value decomposition (SVD) analysis [16], yielding a reduction in feature space dimension of four orders of magnitude (from ~250K to 20).

We wish to examine, in a simple but meaningful way, the predictability of Spam adaptation, and propose two intuitively reasonable criteria with which to empirically evaluate predictability: *sensibility* and *regularity* (a comprehensive theoretical framework for defining and assessing predictability is given in [21]). More specifically, and in the context of Spam, it would be *sensible* for Spammers to adapt their messages over time in such a way that Spam feature vectors x_S come to resemble the feature vectors x_{NS} of legitimate emails, and *regularity* in this adaptation might imply that the values of the individual elements of x_S approach those of x_{NS} in a fairly monotonic way.

To permit convenient examination of the evolution of feature vectors x_S and x_{NS} during the 30 month period under study, the emails were first binned by quarter. Next, the average values for each of the 20 (reduced) features was computed for all the Spam emails and all the non-Spam emails (separately) for each quarter. Figure 1 illustrates the feature space dynamics of Spam and non-Spam messages for one representative coordinate (F1) of this reduced feature space. It can be seen in the plot that the value of feature F1 for Spam approaches the value of this feature for non-Spam, and this increasing similarity is a consequence of changes in the composition of Spam messages (the value of F1 for non-Spam emails is essentially constant). The dynamics of the other feature values are analogous.

Observe that the Spam dynamics illustrated in Figure 1 reflect *sensible* adaptation on the part of Spammers: the features of Spam email messages evolve to appear more like those of non-Spam email, making Spam more difficult to detect. Additionally, this evolution is *regular*, with feature values for Spam approaching those for non-Spam in a nearly-monotonic fashion. Thus this empirical analysis indicates that coevolving Spammer-Spam filter dynamics possesses some degree of predictability, and that the features employed in Spam analysis may have predictive power; this result is in general agreement with the conclusions of the theoretical predictability analysis reported in [20]. Moreover, because many of the characteristics of Spam-Spam defense coevolution are shared by other adversarial systems, this result suggests these other systems may have exploitable levels of predictability as well.

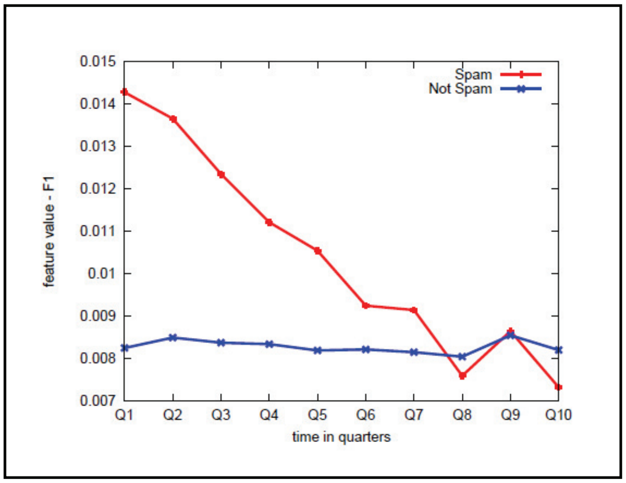

Figure 1. Spam/non-Spam evolution in feature space. The plot depicts evolution of feature F1 for Spam (red) and non-Spam (blue) over time (horizontal axis).

B. Predictive Defense Algorithm

The proposed approach to designing a predictive defense system which works well against both current and future attacks is to combine ML with a simple game-based model for adversary behavior. In order to apply game-theoretic methods, it is necessary to overcome the complexity and model-realism challenges mentioned above. We address problem complexity by modeling adversary actions directly in an aggressively-reduced ML feature space, so that the (effective) space of possible adversary actions which must be considered is dramatically decreased. The difficulty of deriving realistic representations for attacker behavior is overcome by recognizing that the actions of attackers can be modeled as attempts to *transform* data (i.e., feature vectors x) in such a way that malicious and innocent activities are indistinguishable. (This is in contrast to trying to model the attack instances "from scratch"). It is possible to model attacker actions as transformations of data because, within an ML problem formulation, historical attack data are available in the form of training instances.

We model adversarial coevolution as a sequential game, in which the attacker and defender iteratively optimize the following objective function:

$$\min_{w} \max_{a} \left[-\alpha\|a\|^3 + \beta\|w\|^3 + \sum_{i} \mathrm{loss}\big(y_i, w^T(x_i + a)\big) \right] \qquad (1)$$

In (1), the loss function represents the misclassification rate for the defense system, where $\{y_i, x_i\}^n_{i=1}$ denotes pairs of "nominal" activity instances x_i and labels y_i, and vector w parameterizes the defense (recall that the defense attempts to distinguish malicious and innocent activity using the classifier orient = sign($w^T x$)). The attacker attempts to circumvent the defense by transforming the data through vector $a \in \Re^{|F|}$, and the defender's goal is to counter this attack by appropriately specifying classifier vector $w \in \Re^{|F|}$. The terms $-\alpha\|a\|^3$ and $\beta\|w\|^3$ define "regularizations" imposed on attacker and defender actions, respectively, as discussed below.

Note that (1) models the attacker as acting to increase the misclassification rate with vector a, subject to the need to limit the magnitude of this vector (large a is penalized via the term $-\alpha\|a\|^3$). This model thus captures in a simple way the fact that the actions of the attacker are in reality always constrained by the goals of the attack. For instance, in the case of Spam email attacks, the Spammer tries to manipulate message x in such a way that it "looks like" legitimate email and evades the Spam filter w. However, transformed message x+a must still communicate the desired information to the recipient or the attacker's goal will not be realized, and so the transformation vector a cannot be chosen arbitrarily.

The defender attempts to reduce the misclassification rate with an optimal choice for vector w, and avoids "over-fitting" through regularization with the $\beta\|w\|^3$ term [16]. Notice that the formulation (1) permits the attacker's goal to be modeled as counter to, but not exactly the opposite of, the defender's goal, and this is consistent with many real-world settings. Returning to the Spam example, the Spammer's objective of delivering messages which induce profitable user responses is not the inverse of an email service provider's goal of achieving high Spam recognition with a very low false-positive rate.

The preceding development can be summarized by stating the following predictive defense (PD) algorithm:

Algorithm PD

1. Collect historical data $\{y_i, x_i\}^n_{i=1}$ which reflects past behavior of the attacker as well as past legitimate behavior.
2. Optimize objective function (1) to obtain the predicted actions a* of the attacker and the optimal defense w* to counter this attack.
3. Estimate the status of any new activity x as either malicious (+1) or innocent (−1) via orient = sign($x^T w^*$).

Observe that Step 2 of this algorithm can be interpreted as first predicting the attacker strategy through computation of attack vector a*, and then learning an appropriate countermeasure w* by applying ML to the "transformed" data $\{y_i, x_i+a^*\}^n_{i=1}$.

C. Algorithm Evaluation

This case study examines the performance of Algorithm PD for the Spam filtering problem. We use the Spam/non-Spam email dataset introduced above, consisting of ~128,000 messages that were sent during the period 1999-2006. The study compares the effectiveness of Algorithm PD, implemented as a Spam filter, with that of a well-tuned naïve Bayes (NB) Spam filter [5]. Because NB filters are widely used and work very well in Spam applications, this filter is referred to as the gold-standard algorithm. We extract from our dataset the 1000 oldest legitimate emails and 1000 oldest Spam messages for use in training both Algorithm PD and the gold-standard algorithm. The email messages sent during the four year period immediately following the date of the last training email are used as test data. More specifically, these emails are binned by quarter and then randomly sub-sampled to create balanced datasets of Spam and legitimate emails for each of the 16 quarters in the test period.

Recall that Algorithm PD employs aggressive feature space dimension reduction to manage the complexity of the game-based modeling process. This dimension reduction is accomplished here through SVD analysis, which reduces the dimension |F| of feature vectors from ~250K to 20) [16]. (The orthogonal basis used for this reduction is derived by performing SVD analysis using the 1000 non-Spam and 1000 Spam training emails.) Note that good classification accuracy can be obtained with a wide range of (reduced) feature space dimensions. For example, a filtering accuracy of ~97% is achieved with the training data when using an NB classifier implemented with feature dimension ranging from |F|=100,000 to |F|=5.

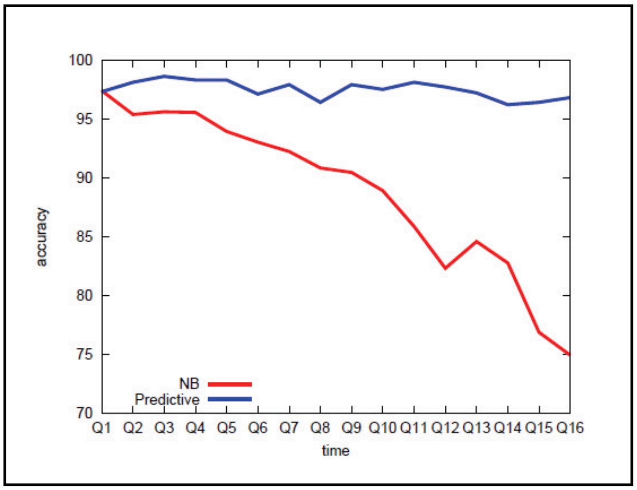

Figure 2. Results for predictive defense Spam filtering case study. The plot shows how filter accuracy (vertical axis) varies with time (horizontal axis) for the gold-standard NB filter (red) and Algorithm PD filter (blue).

The gold-standard strategy is applied as described in [5]. Algorithm PD is implemented with parameter values $\alpha = 0.001$ and $\beta = 0.1$, and with a sum-of-squares loss function. To evaluate the utility of the defenses against evolving adversaries, we train Algorithm PD and the gold-standard algorithm *once*, using the 1000 non-Spam/1000 Spam dataset, and then apply the filters without retraining to the four years of emails that follow these 2000 emails.

Sample results from this study are depicted in Figure 2. Each data point in the plots represents the average accuracy over ten trials (two-fold cross-validation). It can be seen that the filter based upon Algorithm PD significantly outperforms the gold-standard method: the predictive defense experiences almost no degradation in accuracy over the four years of the study, while the gold-standard method suffers a substantial drop in accuracy during this period. These results suggest that combining ML with simple game-based models offers an effective means of defending against evolving adversaries.

III. Moving Target Defense

A. Problem Formulation

A defining characteristic of classification-based defense is the fact that adversaries continually attempt to reverse-engineer the classifier and use this knowledge to make informed adjustments to their behavior and circumvent the defense. One way to increase the difficulty of the adversary's reverse-engineering task is to employ moving target (MT) ideas, in which the defense adopts a time-varying posture in order to increase adversary uncertainty concerning defense operation [17]. In this section we derive an MT defense procedure which minimizes the predictability of defensive actions from the perspective of the attacker.

We investigate MT defense within the framework provided by two-player repeated games with incomplete information [18,19]. In these games one player, the *informed* player, has access to information that is unavailable to the other, *uninformed,* player. The informed player must weigh the relative benefits of exploiting her private information to achieve short-term advantage against the possibility that this exploitation may reveal information which results in the sacrifice of future gains. Because repeated incomplete information games explicitly account for the payoff-predictability tradeoff, they afford a convenient setting for deriving and comparing MT strategies.

Consider the following defense problem. Suppose the task of countering adversarial behavior is formulated as one of ML classification, in which the objective is to distinguish innocent and malicious activity. Each instance of activity is represented as a feature vector $x \in \Re^{|F|}$, where F is the set of ML features. Behavior instances x belong to one of two classes, positive/malicious and negative/innocent, and the goal is to learn a vector $w \in \Re^{|F|}$ such that classifier class = sign($w^T x$) accurately estimates the class of behavior x.

A plausible way to reduce the degree to which adversaries can predict, and then adapt to and evade, the actions of a classifier is to introduce randomness into the way the ML features F are selected and used. One simple way to accomplish this is delineated in the following three steps: 1.) divide the original feature set F into K randomly-selected, possibly overlapping subsets $\{F_1, \ldots, F_K\}$, where $|F_i| = m \ \forall i$; 2.) train one classifier for each feature subset F_i, yielding a collection of K classifiers $\{w_1, \ldots, w_K\}$; 3.) during operation, alternate between the classifiers w_i according to some randomized scheduling policy. In order to implement this MT defense, it is necessary to define a procedure for selecting which classifier is to be "active" at each time period. Thus the MT defense problem of interest can be stated: given a collection of classifiers W={w_1, …, w_K}, specify a policy for switching among classifiers which minimizes defense predictability (from the point of view of the attacker).

B. Moving Target Scheduling Policy

A classifier schedule which minimizes defense predictability is sketched in the following theorem. Perhaps surprisingly, the optimal schedule is very simple to implement.

Theorem MT: Suppose we are given a collection of K classifiers W = {w_1, …, w_K} associated with randomly-selected feature subsets {F_1, …, F_K}, an ecology of adversaries that wish to reverse-engineer the defense, and a sequence of times t_1, t_2, … at which it is permissible to switch classifiers. Under mild assumptions regarding the accuracy of the classifiers W prior to adversary reverse-engineering and the effectiveness of the reverse-engineering methods, defense performance is optimized if, at each time t_i, the active classifier w_a is selected uniformly at random from the set W.

Proof: The proof is given in [23]. ■

We now provide a concise, intuitively-accessible summary of the proof of Theorem MT. Additionally, we describe empirical tests of the theorem's conclusions in Section IIIC below. Readers interested in the technical details of the proof are referred to the report [23]. We model the interaction between an MT defense and an ecology of adversaries as a *hidden mode hybrid dynamical system* (HM-HDS) (see, for instance, [19] for background on this class of dynamical systems). More precisely, the MT defense model is

$$\Sigma_{\text{HM-HDS}} = \{C(w,a), W, P(w,a)\} \qquad (2)$$

where

- the *continuous system* C(w,a) evolves according to sequential attack-defend game dynamics (such as (1));
- the *discrete system* {W,P(w,a)} evolves as a Markov chain with state set W (the set of candidate classifiers) and state transition probability matrix P(w,a); note that, in general, state transition probabilities may depend upon the continuous system state variables (w,a);
- the *hidden mode* is the discrete system state, that is, the currently active classifier $w_a \in W$.

A schematic of this HM-HDS model is depicted in Figure 3.

The dynamics of the HM-HDS (2) evolve as follows. The discrete system specifies the currently active classifier w_a, and this information is communicated to the defender (but not the attacker) in the continuous system game. The attacker attempts to infer which classifier is active by observing defense actions, and computes attack vector a based on this estimate. The discrete system has access to continuous system state (w,a) and may use this information when choosing the next active classifier.

We interpret these dynamics as a repeated incomplete information game, in which the discrete system is the informed player and the attacker dynamics is the uninformed player [18]. (This formulation, although less intuitive than the two-player game model adopted in Section II, facilitates analysis of MT dynamics.) The payoff to the discrete system is defined to be the negative of the misclassification rate, so that maximizing this payoff is equivalent to maximizing the performance of the defense.

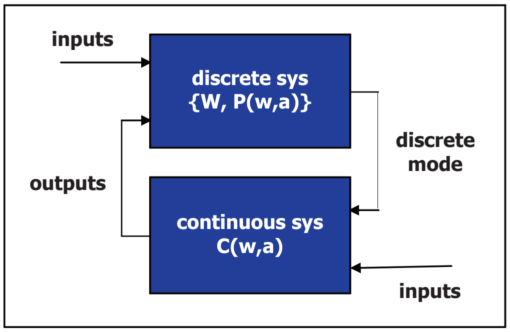

Figure 3. Schematic of basic HM-HDS feedback structure. The discrete and continuous systems in this framework model the selection of "active" classifier w_a and the resulting attack-defend dynamics, respectively.

Now suppose: 1.) each classifier $w_i \in W$ is effective against nominal, "pre-reverse-engineering" attacks (they need not be equally effective), and 2.) the attackers collectively have good reverse-engineering capabilities (i.e., reverse-engineering produces a substantial drop in classifier accuracy for each $w_i \in W$); these conditions are defined more quantitatively in [23]. Under these assumptions, $\Sigma_{\text{HM-HDS}}$ (2) belongs to a class of HM-HDS which is studied in [19]. In that paper, the control of such HM-HDS is formulated as an incomplete information game between a "controller" (the uninformed player) and a "disturbance" (the informed player), where the actions of the disturbance can reveal to the controller exploitable information about the current value of the discrete mode. It is shown in [19] that, in this setting, the best strategy for the disturbance is to maximize the controller's uncertainty regarding the (hidden) discrete mode. This result in turn implies that, in the case of MT defense system (2), the optimal scheduling policy for discrete system {W, P(w,a)} is to select the active classifier w_a uniformly at random from the set W at each time t_i.

Observe that the optimal choice of a new w_a does not depend upon the currently active classifier or the continuous state variables (w,a), basically because any such dependence has the potential to be exploited by the attacker. Additionally, and perhaps counterintuitively, each of the classifiers w_i has an equal probability of being selected to be active, even though some may be more accurate that others. Roughly, if classifier w* is implemented with greater frequency than the others, say because it is especially accurate, the attackers will have increased opportunity to successfully reverse-engineer it, rendering w* less effective than the others in the long run.

C. *Algorithm Evaluation: Spam*

In this section we evaluate the effectiveness of the MT defense strategy summarized in Theorem MT by employing the Spam filtering data and task introduced in Section II. To facilitate convenient comparison with gold-standard defense systems and to reduce complications in the assessment, a few simplifications are made:

- standard NB Spam filters are used for the classifiers $\{w_1, \ldots, w_K\}$ (rather than using, say, the predictive filters generated by solving (1));
- only K=2 classifiers/feature subsets are used;
- attack vector a is computed in an optimal manner via (1), so that the adversary possesses strong reverse-engineering capabilities.

To enable the efficacy of the proposed MT defense to be quantified, its performance is compared to that of a well-tuned static NB filter trained using the full set of (reduced-dimension) features F. We examine a range of attack "strengths" by varying the parameter α in the optimization (1) (recall that the term $-\alpha||a||^3$ governs the magnitude of attack vector a). Attacks are normalized by assigning an attack strength of AS=1 to attacks with magnitude ||a|| equal to the largest attack observed in the (real-world) Spam dataset.

We apply the static NB filter and the optimal two-mode (K=2) MT filter to the 2000 email training dataset described in Section IIC. Additionally, to allow the results of Theorem MT to be tested, we implement a suboptimal MT filter obtained by favoring the more accurate of the two classifiers in the random scheduling process; specifically, the more accurate of the two filters is selected to be active with 2/3 probability (with the less accurate filter then being selected 1/3 of the time). Feature set F is taken to be the collection of 20 features with largest singular values (see Section IIC), and feature subsets F_1 and F_2 are constructed by randomly sampling F (with replacement) until each subset contains 10 features. The filters are "attacked" by solving (1) for the optimal attack a* and then transforming Spam instances x according to the formula x+a*. To allow exploration of a range of attack strengths, (1) is solved for different values of α, yielding the following AS values: AS=0, 0.25, 0.5, 0.75, 1.0, 1.25, 1.5 (thus attacks vary in strength from 'no attack' to attacks with magnitude 1.5 times larger than any seen in the Spam dataset).

Sample results are displayed in Figure 4. Each data point in the plots represents the average accuracy over ten trials (two-fold cross-validation). It can be seen that the filter based upon Theorem MT (red curve) significantly outperforms the static NB filter (magenta curve). For instance, MT defense achieves a classification accuracy of ~90% when subjected to attacks of strength AS=1, compared with the ~65% accuracy obtained with the static filter. Under attacks of magnitude AS=1.5 the optimal MT defense provides an accuracy of ~80%, while the static filter is only slightly more effective than random guessing in this case (accuracy ≈ 54%).

Moreover, this empirical study offers support for the conclusions of Theorem MT. As can be seen from Figure 4, the filter which schedules the more accurate classifier with greater probability (blue curve) does not perform as well as the optimal (according to Theorem MT) MT filter, particularly when the filters are subjected to fairly strong attacks corresponding to effective adversary reverse-engineering. These results suggest that the proposed MT defense is capable of substantially increasing the difficulty of reverse-engineering tasks, even for highly effective (e.g., optimal) attackers.

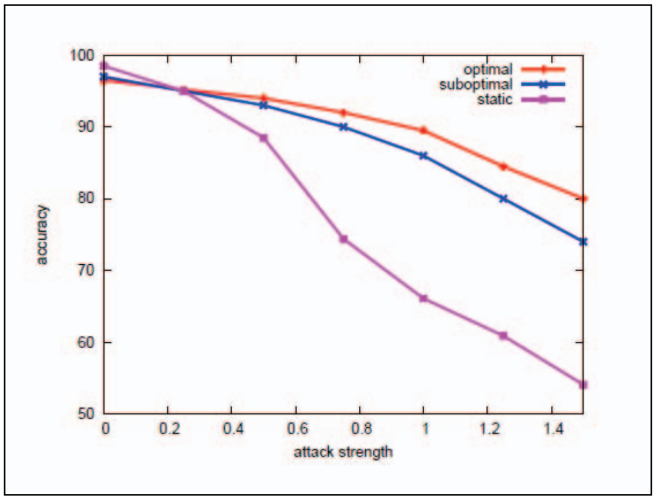

Figure 4. Results for moving target defense Spam filtering case study. The plot shows how filter accuracy (vertical axis) varies with attack strength (horizontal axis) for the optimally scheduled MT filter (red), a suboptimally scheduled MT filter (blue), and the static NB filter (magenta).

D. *Algorithm Evaluation: Network Intrusion*

We now examine the performance of the MT defense strategy summarized in Theorem MT for the problem of distinguishing innocent and malicious computer network activity. The empirical data used for this case study is the KDD Cup 99 dataset, a publicly-available collection of network data consisting of both normal activities and attacks of various kinds [24]. For this study we randomly selected 1000 Normal connections (N) and 1000 denial-of-service attacks (DoS) to serve as our test data.

To enable the efficacy of the proposed MT defense to be quantified, its performance is compared to that of a well-tuned static NB classifier [5]. This NB classifier uses the full set of 30 "continuous" features adopted in previous studies (see, e.g., [5] for a discussion). The optimal two-mode (K=2) MT classifier employs feature subsets F_1 and F_2 constructed by randomly sampling F (with replacement) until each subset contains 15 features. The classifiers are attacked by solving (1) for the optimal attack a* and then transforming DoS network activity instances x according to the formula x+a*. As in the preceding case study, we obtain a range of attack strengths by solving (1) for different values of α (recall $-\alpha||a||^3$ governs the magnitude of attack vector a).

Sample results are displayed in Figure 5. Each data point in the plots represents the average accuracy over ten trials (two-fold cross-validation). It can be seen that the classifier based upon Theorem MT (blue curve) significantly outperforms the static NB classifier (red curve). For instance, the accuracy of the MT defense system never goes below 90%, even when subjected to large attacks, while the accuracy of the static defense quickly falls to 50% as attack strength is increased (this corresponds to random guessing, as the dataset is balanced). Note that this case study illustrates the ease with which the proposed approach can be implemented in different adversarial settings.

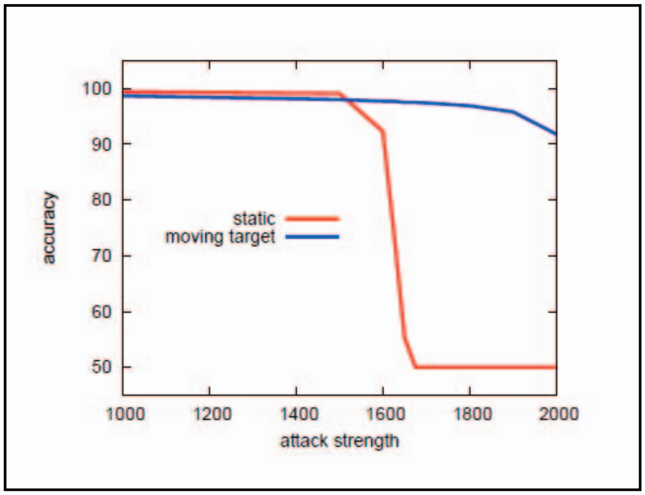

Figure 5. Results for moving target defense network intrusion case study. The plot shows how classifier accuracy (vertical axis) varies with attack strength (horizontal axis) for the optimally scheduled MT defense (blue) and static NB defense (red).

Acknowledgements

This work was supported by the Laboratory Directed Research and Development Program at Sandia National Laboratories. We thank Chip Willard and Curtis Johnson for numerous helpful discussions on aspects of this research.

References

[1] *Proc. 2010 IEEE International Conference on Intelligence and Security Informatics*, Vancouver, BC Canada, May 2010.

[2] *Proc. 2011 IEEE International Conference on Intelligence and Security Informatics*, Beijing, China, July 2011.

[3] *Proc. 2012 IEEE International Conference on Intelligence and Security Informatics*, Washington, DC USA, June 2012.

[4] Colbaugh, R., "Does coevolution in malware adaptation enable predictive defense?", *IFA Workshop Series: Exploring Malware Adaptation Patterns*, San Francisco, CA, May 2010.

[5] Colbaugh, R. and K. Glass, "Proactive defense for evolving cyber threats", *Proc. 2011 IEEE ISI*, Beijing, China, July 2011.

[6] Cormack, G., "Email Spam filtering: A systematic review", *Foundations and Trends in Information Retrieval,* Vol. 1, pp. 335-455, 2008.

[7] Guzella, T. and W. Caminhas, "A review of machine learning approaches to Spam filtering", *Expert Systems with Application*, Vol. 36, pp. 10206-10222, 2009.

[8] Peters, H., *Game Theory*, Springer, Berlin, 2008.

[9] Dalvi, N. et al., "Adversarial classification", *Proc. ACM KDD '04*, Seattle, WA, August 2004.

[10] Roy, S. et al., "A survey of game theory as applied to network security", *Proc. HICSS 2010,* Honolulu, HI, January 2010.

[11] Williams, E., *Surveillance and Interdiction Models: A Game Theoretic Approach to Defend Against VBIED*, Thesis, Naval Postgraduate School, June 2010.

[12] Parameswaran, M., H. Rui, and S. Sayin, "A game theoretic model and empirical analysis of Spammer strategies", *Proc. CEAS 2010*, Redmond, WA, July 2010.

[13] Gkonis, K. and H. Psaraftis, "Container transportation as an interdependent security problem", *J. Transportation Security*, Vol. 3, pp. 197-211, 2010.

[14] Pita, J. et al., "GUARDS: Game theoretic security allocation on a national scale", *Proc. AAMAS '11*, Taipei, Taiwan, May 2011.

[15] Manshaei, M. et al., "Game theory meets network security and privacy", *ACM Computing Surveys,* December 2011.

[16] Hastie, T., R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, Second Edition, Springer, New York, 2009.

[17] *Trustworthy Cyberspace: Strategic Plan for the Federal Cybersecurity Research and Development Program,* December 2011.

[18] Sandholme, T., "State of solving large incomplete information games, and application to poker", *AI Magazine,* pp. 13-32, 2010.

[19] Verma, R. and D. Del Vecchio, "Safety control of hidden mode hybrid systems", *IEEE Trans. Automatic Control,* Vol. 57, pp. 62-77, 2012.

[20] Colbaugh, R., "Arctic ice, George Clooney, lipstick on a pig, and insomniac fruit flies: Combining kd and m&s for predictive analysis", *Proc. ACM KDD '11*, San Diego, CA, August 2011.

[21] Colbaugh, R. and K. Glass, "Predictive analysis for social processes I: Multi-scale hybrid system modeling, and II: Predictability and warning analysis", *Proc. 2009 IEEE MSC*, Saint Petersburg, Russia, July 2009.

[22] Sato, Y., E. Akiyama, and J.D. Farmer, "Chaos in learning a simple two-person game", *Proc. National Academy of Sciences USA,* Vol. 99, pp. 4748-4751, 2002.

[23] Colbaugh, R. and K. Glass, "Predictive dynamic defense against adaptive adversaries", Sandia National Laboratories Technical Report, April 2012.

[24] http://kdd.ics.uci.edu/databases/kddcup99/; accessed Dec. 2010.