Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16)

Optimizing Personalized Email Filtering Thresholds to Mitigate Sequential Spear Phishing Attacks

Mengchen Zhao
School of Computer Engineering
Nanyang Technological University
Singapore 639798
zhao0204@e.ntu.edu.sg

Bo An
School of Computer Engineering
Nanyang Technological University
Singapore 639798
boan@ntu.edu.sg

Christopher Kiekintveld
University of Texas, El Paso
El Paso, TX 79968
cdkiekintveld@utep.edu

Abstract

Highly targeted spear phishing attacks are increasingly common, and have been implicated in many major security breeches. Email filtering systems are the first line of defense against such attacks. These filters are typically configured with uniform thresholds for deciding whether or not to allow a message to be delivered to a user. However, users have very significant differences in both their susceptibility to phishing attacks as well as their access to critical information and credentials that can cause damage. Recent work has considered setting personalized thresholds for individual users based on a Stackelberg game model. We consider two important extensions of the previous model. First, in our model user values can be substitutable, modeling cases where multiple users provide access to the same information or credential. Second, we consider attackers who make sequential attack plans based on the outcome of previous attacks. Our analysis starts from scenarios where there is only one credential and then extends to more general scenarios with multiple credentials. For single-credential scenarios, we demonstrate that the optimal defense strategy can be found by solving a binary combinatorial optimization problem called PEDS. For multiple-credential scenarios, we formulate it as a bilevel optimization problem for finding the optimal defense strategy and then reduce it to a single level optimization problem called PEMS using complementary slackness conditions. Experimental results show that both PEDS and PEMS lead to significant higher defender utilities than two existing benchmarks in different parameter settings. Also, both PEDS and PEMS are more robust than the existing benchmarks considering uncertainties.

Introduction

Email is not a secure communications channel, and attackers have exploited this via spam emails for many years. However, in recent years cyber attacks using email have become increasingly targeted and much more damaging to organizations (TrendLabs 2012). These targeted email attacks are commonly known as spear phishing. They target individuals or small groups of people, but use personal information and social engineering to craft very believable messages with the goal of inducing the recipient to open an attachment, or visit an unsafe website by clicking a link. Executing a spear phishing attack is much more costly than sending a broad spam message, but it is also much more likely to succeed and the potential damage is much greater. For example, in 2011 the RSA company was breached by a spear phishing attack (Zetter 2011). This attack resulted in privileged access to secure systems, and stolen information related to the company's SecurID two-factor authentication products.

Email filtering systems are one of the primary defenses against both spam and spear phishing attacks. These systems typically use black and white lists as well as machine learning methods to score the likelihood that an email is malicious before sending it to a user (Bergholz et al. 2010). Setting the threshold for how safe a message must be to be delivered is a key strategic decision for the network administrator (Sheng et al. 2009). If the threshold is too high, malicious emails will easily pass the filtering system, but if the threshold is too low normal emails will be filtered. Recent work has proposed a game-theoretic model that can improve the effectiveness of filtering if thresholds are personalized according to individuals' values and susceptibilities (Laszka, Vorobeychik, and Koutsoukos 2015). They assume that the attacker's strategy is simply selecting a subset of users to attack that maximizes an additive expected reward, ignoring the cost of attacks and the outcome of previous attacks.

However, in many incidents such as Operation Aurora (Varma 2010), attackers launches sophisticated attacks toward few targets over months. In such cases, attackers have plenty time and attack resources and they can plan long term sequential attack strategies to achieve difficult objectives (Watson, Mason, and Ackroyd 2014). In this paper we extend the literature (and particularly the personalized filtering model of (Laszka, Vorobeychik, and Koutsoukos 2015)) to consider more sophisticated attackers who can make sequential decisions about which users to send spear phishing emails to. Specifically, we consider more complex (also realistic) objective functions for both the attacker and defender, including modeling attack costs, and situations where it is only necessary to compromise one user from a set of users that has access to important data or credentials (i.e., the user values are substitutable).

Our contributions are fourfold. First, we consider the case where there is a single important credential that the attacker seeks to gain and model the attacker's decision mak-

ing as a Markov Decision Process (MDP). We formulate a bilevel optimization problem for the defender and show that the attacker's problem (i.e., lower level problem) can be solved by a linear program. Solving the linear program is computational consuming since the number of variables and constraints grow exponentially with the number of users. Our second contribution is to find a simplified representation of the defender's utility and thus reduce the defender's bilevel program into a single level binary combinatorial optimization program (which we call PEDS) by exploiting the structure of the attacker's MDP. Our third contribution is to extend the single-credential case to a a more general case where there could be multiple sensitive credentials. For the multiple-credential case, the defender's utility cannot be represented in the same way as the single-credential case. We consider the dual program of the linear program that solves the attacker's MDP and represent the defender's loss from spear phishing attacks by a linear combination of dual variables. We then propose a single level formulation (which we call PEMS) for the defender, which is reduced from the proposed bilevel problem using complementary slackness conditions. Our fourth contribution is to evaluate PEDS and PEMS by comparing our solutions with two existing benchmarks and show that our solutions lead to significant higher defender utilities in different parameter settings and are also robust considering uncertainties.

Sequential Attacks with A Single Credential

We consider a spear phishing game between an attacker and a defender. The defender (e.g., an organization) has a *credential*[1] that can be accessed by a set of users $U = \{1, 2, ..., |U|\}$. For now we consider only a single credential, and later generalize the model to multiple credentials. The attacker, wanting to gain access to the credential, sends spear phishing emails to the users based on an attack plan taking into account the *susceptibility*, *confidentiality level* and *attack cost* of the users. We denote by a_u the susceptibility of user u, meaning that u will be compromised with probability a_u after a spear phishing email is delivered to her. There are many methods to measure a_u, e.g., by sending probe emails to the users (Sheng et al. 2010; Kelley 2010; Jagatic et al. 2007). We denote by k_u the confidentiality level of user u, meaning that user u can access the credential with probability k_u when she is compromised. The attacker sustains some costs when launching attacks, such as crafting phishing emails, investigating users and writing malware. We denote by c_u the cost of attacking user u.

When receiving emails, the filter first scores them according to their likelihood of being malicious emails, and then delivers only those with scores lower than a given threshold (Deshmukh, Shelar, and Kulkarni 2014; Bergholz et al. 2010). It is possible that malicious emails are misclassified as normal ones. We call such misclassifications *false negatives*. On the other side, some normal emails might be misclassified as malicious. We call such misclassifications *false positives*. In binary classification, a threshold determines a

[1]We use the generic term "credential" here to mean any critical data or access privilege that the attacker is seeking to gain.

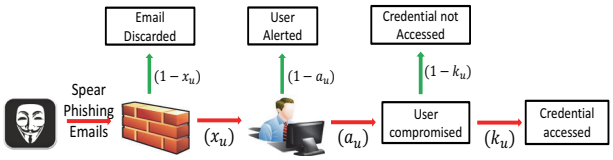


Figure 1: Spear Phishing Attack Flow

pair (x_u, y_u) where $x_u, y_u \in [0, 1]$ are the false negative rate and the false positive rate, respectively. Moreover, the relationship between x_u and y_u can be characterized as a function $\Phi : [0, 1] \to [0, 1]$, $y_u = \Phi(x_u)$, which is a Receiver Operating Characteristic (ROC) curve, with y-axis replaced by false positive rate (Fawcett 2006). In practice, Φ is represented by a set of data points and can be approximated by a piecewise linear function ϕ (Bouckaert 2006). By adjusting the thresholds, the organization can determine a pair (x_u, y_u) for each user. We will use the false negative rate vector $\mathbf{x}$ to represent the defender's strategy. But note that using $\mathbf{y}$ as the defender's strategy is equivalent to as Φ and ϕ are bijections. Intuitively, the defender actually controls the probability that malicious emails will pass the filter (x_u) and the probability that normal emails will be filtered (y_u).

Figure 1 shows the attack flow. The attacker sends a spear phishing email to a targeted user. The email will pass the filter with probability x_u and otherwise be discarded. We assume that the attacker is able to observe whether the email is delivered and opened by the user using email tracking techniques[2]. When receiving the email, the user will be tricked with probability a_u and otherwise be *alerted*. We assume that if the user is tricked, she will be *compromised*, and if the user is alerted, she will be aware of being targeted and not be tricked by subsequential phishing emails. If the user is compromised, the attacker can access the credential with probability k_u.

Stackelberg Spear Phishing Game

We model the interaction between the defender and the attacker as a Stackelberg game. The defender moves first by choosing a false negative probability vector $\mathbf{x}$. After observing $\mathbf{x}$[3], the attacker launches an optimal attack. We denote by $\pi_{\mathbf{x}}$ the attacker's optimal policy that maximizes his expected utility given the defender's strategy $\mathbf{x}$.

We denote by $P_a(\mathbf{x}, \pi_{\mathbf{x}})$ the attacker's expected utility and by $P_d(\mathbf{x}, \pi_{\mathbf{x}})$ the defender's expected utility given strategy profile $(\mathbf{x}, \pi_{\mathbf{x}})$. We denote by L the value of the credential. The attacker suffers a cost c_u each time he attacks user u and he gains L if he accesses the credential. The defender's loss is threefold. (1) The defender loses L if the credential is accessed by the attacker. (2) The defender loses FP_u for per normal email sent to user u filtered. (3) Besides

[2]For example, Yesware provides services allowing their clients to view the detailed status of outgoing emails, including whether the emails are opened and the time the receivers spend on each email(Hlatky 2015).

[3]We make the worst-case assumption that the attacker knows $\mathbf{x}$ since spear phishers collect security information about the organization before attacking (Choo 2011).

spear phishing attacks, the defender also faces mass attacks (e.g., spam and regular phishing emails), which are usually less harmful than spear phishing attacks. We assume that the probability that a mass attack email passes the filter is x_u[4] and the defender loses N_u for per mass attack email delivered to user u. Note that the defender sustains the second and the third parts of loss constantly as normal emails and mass attack emails are sent to users constantly. However, spear phishing attacks usually happen in a relatively short period. To make the three kinds of losses comparable, we assume that the defender's expected utility is measured in a time period $\mathcal{T}$. We denote by $FP_u^{\mathcal{T}}$ the expected loss of misclassifying normal emails sent to user u and by $N_u^{\mathcal{T}}$ the expected loss of delivering mass attack emails sent to user u during $\mathcal{T}$, which can be computed by

$$FP_u^{\mathcal{T}} = FP_u \times E[\text{number of normal emails sent to } u \text{ during } \mathcal{T}]$$

$$N_u^{\mathcal{T}} = N_u \times E[\text{number of mass attack emails sent to } u \text{ during } \mathcal{T}]$$

The defender's loss from filtering normal emails and delivering mass attack emails can be simply represented as the summation of the loss from every individual user, $\sum_{u\in U} x_u N_u^{\mathcal{T}}$ and $\sum_{u\in U} \phi(x_u) FP_u^{\mathcal{T}}$ respectively. However, the defender's loss from spear phishing attacks is not cumulative. We denote by $\rho^{\mathcal{T}}$ the probability that the spear phishing attacks occur in time period $\mathcal{T}$ and by $\theta(\mathbf{x}, \pi_{\mathbf{x}})$ the probability that the attacker will access the credential given the strategy profile $(\mathbf{x}, \pi_{\mathbf{x}})$. Then the defender's expected utility can be represented as

$$P_d(\mathbf{x},\pi_{\mathbf{x}}) = -\rho^{\mathcal{T}}\theta(\mathbf{x},\pi_{\mathbf{x}})L - \sum_{u\in U} x_u N_u^{\mathcal{T}} - \sum_{u\in U} \phi(x_u) FP_u^{\mathcal{T}} \quad (1)$$

We consider the widely used strong Stackelberg equilibrium (SSE) as our solution concept (Korzhyk et al. 2011; Gan, An, and Vorobeychik ; Yin et al. 2015; Yin, An, and Jain 2014).

Definition 1. *If a strategy profile $(\mathbf{x}^*, \pi_{\mathbf{x}^*})$ such that $P_d(\mathbf{x}^*, \pi_{\mathbf{x}^*}) \geq P_d(\mathbf{x}, \pi_{\mathbf{x}})$ holds for any possible $\mathbf{x}$, under the assumption that the attacker plays a best response and breaks ties among multiple optimal policies in favor of the defender, then $(\mathbf{x}^*, \pi_{\mathbf{x}^*})$ is an SSE strategy profile.*

Optimal Attack with A Single Credential

In this section, we model the attacker's decision making as a Markov Decision Process (MDP) and show that the MDP can be solved by a linear program.

Attacker's MDP

The attacker's MDP can be represented as a tuple $(\mathcal{S}, \mathcal{A}, T, R, \pi)$. $\mathcal{S} = \{s | s \subseteq U\} \cup \{s^n, s^y\}$ is the state space that consists of *non-terminal states* and two *terminal states* s^n, s^y. A non-terminal state corresponds to a subset of the user set U that represents the users who have not been alerted or compromised. The initial state is $s_0 = U$. The terminal state s^n represents the situation where the attacker stops attacking without accessing the credential, while s^y represents the situation where the attacker stops attacking with the credential accessed. $\mathcal{A} = \{a | a = u \in U \text{ or } a = stop\}$ is the attacker's action space where $a = u$ means that the attacker chooses to attack user u, and $a = stop$ means that the attacker stops attacking. We denote by $\mathcal{A}^s = \{a | a = u \in s \text{ or } a = stop\}$ the attacker's action space at non-terminal state s, since the attacker only attacks users that have not been alerted or compromised. Transition function $T(s, a, s')$ represents the probability that s transitions to s' by executing action a. Reward function $R(s, a, s')$ represents the attacker's reward when s transitions to s' by executing action a. $\pi : \mathcal{S} \to \mathcal{A}$ is a deterministic function that projects each non-terminal state to an action.

Now we define T and R. We assume that the terminal states always transition to themselves with probability 1 and with reward 0. For any non-terminal state s, if the attacker stops attacking, s transitions to s^n with reward 0. If the attacker chooses to attack user $u \in \mathcal{A}^s$, there are four possible transitions: (1) If the malicious email fails to pass the filter, s transitions to itself. The transition probability is $1 - x_u$ and the reward is $-c_u$. (2) If the email is delivered and user u is alerted, s transitions to $s^{-u} = s \setminus \{u\}$. The transition probability is $x_u(1 - a_u)$ and the reward is $-c_u$. (3) If the email passes the filter and u is compromised, however u does not have access to the credential, then s transitions to $s^{-u} = s \setminus \{u\}$. The transition probability is $x_u a_u (1 - k_u)$ and the reward is $-c_u$. Note that in both transitions (2) and (3), s transitions to s^{-u} with the same reward $-c_u$. Therefore they can be merged into one transition with probability $x_u(1-a_u) + x_u a_u (1-k_u) = x_u(1 - a_u k_u)$ and with reward $-c_u$. (4) The email passes the filter, user u is compromised and she can access the credential, s transitions to s^y. The transition probability is $x_u a_u k_u$. The transition function T and the reward function R can be summarized as:

	$T(s,a,s')$	$R(s,a,s')$
$a = stop, s' = s^n$	1	0
$a = u \in \mathcal{A}^s, s' = s$	$1 - x_u$	$-c_u$
$a = u \in \mathcal{A}^s, s' = s^{-u}$	$x_u(1 - a_u k_u)$	$-c_u$
$a = u \in \mathcal{A}^s, s' = s^y$	$x_u a_u k_u$	$L - c_u$

Solving the MDP

The value function $V^\pi : \mathcal{S} \to \mathbb{R}$ represents the attacker's expected utility when his current state is s and he follows a policy π afterwards. Moreover, we denote by V^* the value function when the attacker follows the optimal policy $\pi_{\mathbf{x}}$. Then the attacker's expected utility can be written as

$$P_a(\mathbf{x}, \pi_{\mathbf{x}}) = V^*(s_0).$$

The attacker's MDP can be solved by the following linear program (Schweitzer and Seidmann 1985).

$$\min_{\mathbf{V}_a^*} \quad \sum_{s\in\mathcal{S}\setminus\mathcal{S}^T} \mu(s) V_a^*(s) \quad (2)$$

$$\text{s.t.} \quad V_a^*(s) \geq \sum_{s'\in\mathcal{S}} T(s,a,s') \left[R(s,a,s') + V_a^*(s')\right]$$
$$\forall a \in \mathcal{A}^s, \forall s \in \mathcal{S} \setminus \mathcal{S}^T \quad (3)$$

$$V_a^*(s) = 0, \quad \forall s \in \mathcal{S}^T \quad (4)$$

[4]This assumption means that the classification accuracies for spear phishing emails and mass attack emails are the same. Note that our approach can be easily extended to the case where these accuracies are different, by introducing a function that captures the relationship between these accuracies.

where $S^T = \{s^n, s^y\}$ denotes the set of terminal states and $\mu(s)$ is the probability that the MDP starts from state s. Since we have an initial state s_0, $\mu(s) = 1$ if $s = s_0$ and 0 otherwise. The optimal policy $\pi_{\mathbf{x}}$ can be obtained:

$$\pi_{\mathbf{x}}(s) = \arg\max_{a \in \mathcal{A}^s} Q(s,a),\ \forall s \in \mathcal{S} \setminus \mathcal{S}^T,$$

where $Q(s,a) = \sum_{s' \in \mathcal{S}} T(s,a,s') \left[R(s,a,s') + V_a^*(s')\right]$.

Optimal Defense with A Single Credential

The defender seeks a false negative probability vector $\mathbf{x}$ that maximizes her expected utility given that the attacker plays the optimal policy $\pi_{\mathbf{x}}$. The defender's optimization problem is given by the following bilevel optimization program.

$$\max_{\mathbf{x}} \quad P_d(\mathbf{x}, \pi_{\mathbf{x}}) \tag{5}$$

$$\text{s.t.} \quad x_u \in [0,1],\ \forall u \in U \tag{6}$$

$$\pi_{\mathbf{x}} \in \arg\max_{\pi} V^{\pi}(s_0) \tag{7}$$

Eq.(5) represents the defender's expected utility. Eqs.(6) indicates that the false negative rate can only be chosen from $[0,1]$. Eq.(7), i.e., the lower level problem, assures that the attacker always responds optimally. The hardness of solving Eqs.(5)-(7) is twofold. First, $\theta(\mathbf{x}, \pi_{\mathbf{x}})$ in Eq.(5) does not have an explicit representation with respect to variables $\mathbf{x}$. Second, the lower level problem is hard to be characterized by a set of constraints. We will first show how to represent $\theta(\mathbf{x}, \pi_{\mathbf{x}})$, and then show that bilevel program Eqs.(5)-(7) is equivalent to a single level program called PEDS.

Representing $\theta(\mathbf{x}, \pi_{\mathbf{x}})$

In fact, $\theta(\mathbf{x}, \pi_{\mathbf{x}})$ is the probability that the attacker ends in the terminal state s^y given that he follows the optimal policy $\pi_{\mathbf{x}}$. Before we show how to represent $\theta(\mathbf{x}, \pi_{\mathbf{x}})$, we introduce two concepts: *reachable states* and *potential attack set*. Once a policy is determined, the MDP is reduced to a Markov chain where only some states (called *reachable states*) can be reached from the initial state if we consider the Markov chain as a graph. For example, if $s_0 = \{u_1, u_2\}$ and $\pi(s_0) = u_1$, then state $s = \{u_1\}$ cannot be reached from s_0 with a nonnegative probability. We denote by $\Delta(\pi)$ the set of reachable states given the policy π. A policy π projects each reachable state $s \in \Delta(\pi)$ to an action $a \in \mathcal{A}^s$. We denote by $\Gamma(\pi)$ the potential attack set, which is the set of users that are projected from the reachable states under the policy π, i.e., $\Gamma(\pi) = \{\pi(s) | s \in \Delta(\pi)\}$. Lemma 1 states that if the immediate expected gain of attacking user u (i.e., $x_u a_u k_u L$) is greater than the attack cost c_u, then the user is in the potential attack set[5].

Lemma 1. *$u \in \Gamma(\pi_{\mathbf{x}})$ if and only if $x_u a_u k_u L > c_u$.*

Lemma 2 shows that $\theta(\mathbf{x}, \pi_{\mathbf{x}})$ can be easily computed given the potential attack $\Gamma(\pi_{\mathbf{x}})$.

Lemma 2.

$$\theta(\mathbf{x}, \pi_{\mathbf{x}}) = \begin{cases} 1 - \prod_{u \in \Gamma(\pi_{\mathbf{x}})}(1 - a_u k_u), & \text{if } \Gamma(\pi_{\mathbf{x}}) \neq \emptyset \\ 0, & \text{if } \Gamma(\pi_{\mathbf{x}}) = \emptyset \end{cases}.$$

[5]All proofs of Lemmas and Theorems are in the appendix: http://www.ntu.edu.sg/home/boan/papers/AAAI16_Phishing_Appendix.pdf.

Combining Lemmas we can show that even though the attacker may have multiple optimal policies, they have the same potential attack set. Therefore, $\theta(\mathbf{x}, \pi_{\mathbf{x}})$ does not change for different optimal attack policies.

Theorem 1. *The defender's expected utility remains the same no matter how the attacker breaks ties, i.e., choosing any optimal policy.*

PEDS: Reduced Single Level Program

Now we show how to solve Eqs.(5)-(7) based on the lemmas. We define a function Λ_u for each user u:

$$\Lambda_u(x) = x N_u^{\mathcal{T}} + \phi(x) F P_u^{\mathcal{T}},\ x \in [0,1].$$

$\Lambda_u(x)$ represents the total loss from mass attacks and false positives of user u if she is assigned a false negative probability x. Λ_u is a piecewise linear function since it is the sum of a linear function and a piecewise linear function. Therefore, we can easily find a set $\arg\min_x \Lambda_u$ for each user.

We rewrite the defender's utility as

$$P_d(\mathbf{x}, \pi_{\mathbf{x}}) = -\rho^{\mathcal{T}} \theta(\mathbf{x}, \pi_{\mathbf{x}}) L - \sum_{u \in U} \Lambda_u(x_u).$$

Lemma 2 indicate that the value of $\theta(\mathbf{x}, \pi_{\mathbf{x}})$ depends on the potential attack set $\Gamma(\pi_{\mathbf{x}})$. We define a set $\mathcal{U} = \{u | \frac{c_u}{L a_u k_u} \in [0,1], u \in U\}$. If $u \in U \setminus \mathcal{U}$, the optimal false negative rate x_u^* can be any arbitrary point of $\arg\min_x \Lambda_u$ since $u \in \Gamma(\pi_{\mathbf{x}})$ holds for any $x_u \in [0,1]$. If $u \in \mathcal{U}$, it holds that $u \notin \Gamma(\pi_{\mathbf{x}})$ when $x_u \in [0, \frac{c_u}{L a_u k_u}]$ and $u \in \Gamma(\pi_{\mathbf{x}})$ when $x_u \in (\frac{c_u}{L a_u k_u}, 1]$. Given $u \in \mathcal{U}$, we denote by x_u^1 the optimal false negative rate if $u \notin \Gamma(\pi_{\mathbf{x}})$ and by x_u^2 the optimal false negative rate if $u \in \Gamma(\pi_{\mathbf{x}})$.

Theorem 2. *x_u^1 is an arbitrary point in $\arg\min_{x \in [0, \frac{c_u}{L a_u k_u}]} \Lambda_u$ and x_u^2 is an arbitrary point in $\arg\min_{x \in (\frac{c_u}{L a_u k_u}, 1]} \Lambda_u$.*

Then program Eqs.(5) - (7) are equivalent to the following binary combinatorial optimization problem, which we call PEDS (Personalized thrEsholds in Defending Sequential spear phishing attacks):

$$\max_{\alpha} \quad -\rho^{\mathcal{T}}(1 - \prod_{u \in U} \beta_u) L - \sum_{u \in U} \Lambda_u(x_u) \tag{8}$$

$$\text{s.t.} \quad x_u = x_u^0,\ \forall u \in U \setminus \mathcal{U} \tag{9}$$

$$\beta_u = 1,\ \forall u \in U \setminus \mathcal{U} \tag{10}$$

$$x_u = x_u^1 + (x_u^2 - x_u^1)\alpha_u,\ \forall u \in \mathcal{U} \tag{11}$$

$$\beta_u = 1 - a_u k_u \alpha_u,\ \forall u \in \mathcal{U} \tag{12}$$

$$\alpha_u \in \{0,1\},\ \forall u \in \mathcal{U} \tag{13}$$

where x_u^0 can be an arbitrary point from $\arg\min_x \Lambda_u$. α_u is the indicator of whether user u is in the potential attack set $\Gamma(\pi_{\mathbf{x}})$. $\alpha_u = 0$ indicates that u is in $\Gamma(\pi_{\mathbf{x}})$ and 1 otherwise. Since PEDS's decision variables are binary, we can find the optimal solutions by using CPLEX CP Optimizer.

Multiple-Credential Model

An organization may need to protect many different credentials or pieces of sensitive information. We now consider

the multiple-credential case, where the attacker's decision making can still be modeled as an MDP. Note that Eqs.(5) - (7) still represent the defender's optimization problem except that π and V represent the policy and the value function of the new MDP. In this section, we first introduce the attacker's MDP with multiple credentials. Then we give the dual formulation of Eqs.(2) - (4) and show that using complementary slackness conditions, Eq.(7) (i.e., the lower level optimization problem) can be replaced by a set of constraints, which guarantee that the attacker plays the best response. Consequently, bilevel program Eqs.(5) - (7) is reduced to a single level program which can be directly solved.

Optimal Attack with Multiple Credentials

We denote by $H = \{1, 2, ..., |H|\}$ the set of credentials, by L_h the value of the credential h and by m_u^h the probability that user u can access credential h. With multiple credentials, the attacker's MDP can be represented as a tuple $(\mathcal{S}, \mathcal{A}, T, R, \pi)$. $\mathcal{S} = \mathcal{P}(U) \otimes \mathcal{P}(H)$ is the state space, where $\mathcal{P}(U)$ ($\mathcal{P}(H)$) is the power set of U (H). A state $s \in \mathcal{S}$ can be represented as $s = s(U) \otimes s(H)$, where $s(U) \subseteq U$ represents the set of users that have not be alerted or compromised and $s(H) \subseteq H$ represents the set of credentials that have not been accessed by the attacker. s is a terminal state if either $s(U) = \emptyset$ or $s(H) = \emptyset$, i.e., all the users have been alerted or compromised, or all credentials have been accessed. We use $\mathcal{S}^T$ to represent the set of terminal states. At each non-terminal state, the attacker's action space is $\mathcal{A}^s = \{a | a = u \in s(U) \text{ or } a = stop\}$, in the sense that the attacker does not attack users that have been alerted or compromised. $\pi : \mathcal{S} \to \mathcal{A}$ represents a policy of the attacker.

$T(s, a, s')$ represents the probability and $R(s, a, s')$ represents the attack reward that s transitions to s' by executing action a. We assume that terminal states transition to themselves with probability 1 and reward 0. We define the transitions and rewards as follows. (1) If $s = s(U) \otimes s(H)$ is a non-terminal state and the attacker chooses to stop attacking at s, s transitions to the terminal state $s' = \emptyset \otimes s(H)$ with probability 1 and reward 0. (2) If $s = s(U) \otimes s(H)$ is a non-terminal state and the attacker chooses to attack a user $u \in s(U)$ at s, there are 3 kinds of transitions. (2.1) The malicious email is filtered, in which case s transitions to itself. The transition probability is $1 - x_u$ and the reward is $-c_u$. (2.2) The malicious email is delivered and user u is alerted, in which case s transitions to $s' = s(U) \setminus \{u\} \otimes s(H)$. The transition probability is $x_u(1 - a_u)$ and the reward is $-c_u$. (2.3) The malicious email is delivered and user u is compromised, after which the attacker will access each credential $h \in H$ with probability m_u^h. We have $T(s, a = u, s') =$

$$\begin{cases} 0, \text{ if } s'(U) \neq s(U) \setminus \{u\} \text{ or } s'(H) \not\subseteq s(H), \\ x_u a_u \prod_{h \in s(H) \setminus s'(H)} m_u^h \prod_{h \in s'(H)} (1 - m_u^h), \text{ otherwise.} \end{cases}$$

The associated rewards $R(s, a=u, s') =$

$$\begin{cases} 0, \text{ if } s'(U) \neq s(U) \setminus \{u\} \text{ or } s'(H) \not\subseteq s(H), \\ \sum_{h \in s(H) \setminus s'(H)} L_h, \text{ otherwise.} \end{cases}$$

The MDP can still be solved by linear program Eqs.(2)-(4).

Defender's Loss from Spear Phishing Attacks

When there are multiple credentials, the probability of losing the credentials cannot be computed in the same way as in the single-credential case. We introduce another way to represent the defender's expected utility. Consider the dual of linear program Eqs.(1)-(3):

$$\max_{\mathbf{W}} \sum_{s \in \mathcal{S} \setminus \mathcal{S}^T} \sum_{a \in \mathcal{A}^s} \sum_{s' \in \mathcal{S}} T(s, a, s') R(s, a, s') W(s, a) \tag{14}$$

$$\text{s.t.} \sum_{a' \in \mathcal{A}^{s'}} W(s', a') = \mu(s') + \sum_{s \in \mathcal{S} \setminus \mathcal{S}^T} \sum_{a \in \mathcal{A}^s} W(s, a) T(s, a, s')$$

$$\forall s' \in \mathcal{S} \setminus \mathcal{S}^T \tag{15}$$

$$W(s, a) \in \mathbb{R}^+, \forall a \in \mathcal{A}^s, \forall s \in \mathcal{S} \setminus \mathcal{S}^T \tag{16}$$

The dual variable $\mathbf{W}$ is called the *occupation measure* (Borkar and Ghosh 1992). $W(s, a)$ can be interpreted as the expected total number of times that the system is in state s and action a is executed. $\sum_{a \in \mathcal{A}^s} W(s, a)$ is the expected total number of visits to state s. We define a reward function $R_d(s, a, s')$ for the defender.

$$R_d(s, a, s') = \begin{cases} -(R(s, a, s') + c_u), & \text{if } a = u \in \mathcal{A}^s, \\ 0, & \text{if } a = stop. \end{cases}$$

Recall that $R(s, a, s')$ is the attacker's reward when he executes action a and the state transitions from s to s'. In fact, $R(s, a, s')$ consists of the gain of accessing some credentials (positive) and the cost of attack (negative). The defender's loss can thus be represented as $-(R(s, a, s') + c_u)$ if $a = u \in \mathcal{A}^s$, and 0 if $a = stop$. Therefore the defender's expected loss from spear phishing attacks can be represented as $\sum_{s \in \mathcal{S} \setminus \mathcal{S}^T, a \in \mathcal{A}^s} W(s, a) \sum_{s' \in \mathcal{S}} T(s, a, s') R_d(s, a, s')$.

Single Level Formulation

It follows that feasible solutions $\mathbf{V}_a^*$ and $\mathbf{W}$ are optimal for the original LP and its dual problem if the following complementary slackness conditions are satisfied:

$$\{V_a^*(s) - \sum_{s' \in \mathcal{S}} T(s, a, s')[R(s, a, s') + V_a^*(s')]\} W(s, a) = 0,$$

$$\forall a \in \mathcal{A}^s, \forall s \in \mathcal{S} \setminus \mathcal{S}^T. \tag{17}$$

Then the bilevel program Eqs.(5)-(7) can be converted to the following single level program, which we call PEMS (Personalized thrEsholds in protecting Multiple credentialS):

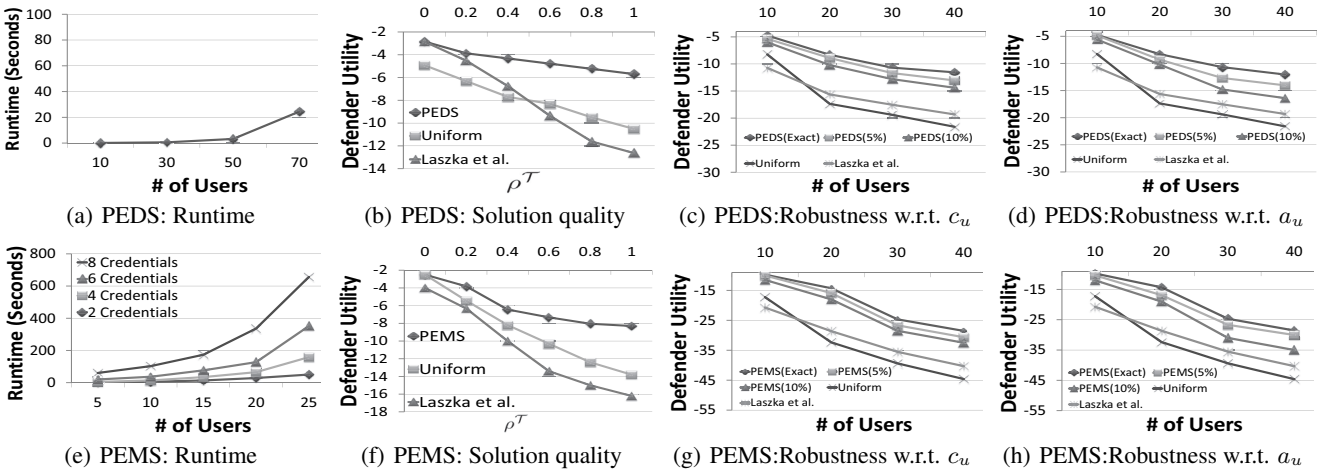
$$\max_{\mathbf{x}} \quad \rho^{\mathcal{T}} \sum_{s \in \mathcal{S} \setminus \mathcal{S}^T, a \in \mathcal{A}^s} W(s, a) \sum_{s' \in \mathcal{S}} T(s, a, s') R_d(s, a, s') - \sum_{u \in U} \Lambda_u(x_u) \tag{18}$$

$$\text{s.t.} \quad \text{Eqs.(3), (4), (15) and (17)}$$

$$W(s, a) \in \mathbb{R}^+, \forall a \in \mathcal{A}^s, \forall s \in \mathcal{S} \setminus \mathcal{S}^T \tag{19}$$

$$x_u \in [0, 1], \ \forall u \in U \tag{20}$$

PEMS is a nonlinear program as Eqs.(3), (15), (17) and (18) are nonlinear. We can use solver KNITRO to solve PEMS.



(a) PEDS: Runtime (b) PEDS: Solution quality (c) PEDS:Robustness w.r.t. c_u (d) PEDS:Robustness w.r.t. a_u

(e) PEMS: Runtime (f) PEMS: Solution quality (g) PEMS:Robustness w.r.t. c_u (h) PEMS:Robustness w.r.t. a_u

Figure 2: Runtime Performance: (a) and (e), Solution Quality Comparisons: (b) and (f), Robustness Analysis (c), (d), (g) and (h)

Experimental Evaluation

We evaluate PEDS and PEMS in terms of runtime, solution quality and robustness. All values of parameters are uniformly randomly generated from an interval unless otherwise specified. Specifically, values of credentials are generated from [10,15]. Attack costs c_u are generated from [0,2]. Users' susceptibilities a_u and their accesses to credentials k_u (m_u^h in multiple-credential case) are generated from [0,0.5]. Losses from mass attacks $N_u^{\mathcal{T}}$ and losses from false positives $FP^{\mathcal{T}}$ are generated from [0,1] and [0,5], respectively. PEDS is solved by CPLEX CP Optimizer (version 12.6) and PEMS is solved by KNITRO (version 9.0). All computations were performed on a 64-bit PC with 16 GB RAM and a quad-core Intel E5-1650 3.20GHz processor. We use a 10-section piecewise linear function ϕ to approximate the original false negative-false positive function Φ, which is drawn from prior work (Laszka, Vorobeychik, and Koutsoukos 2015).

We compare the solutions computed by PEDS and PEMS with two existing benchmarks. **Uniform**: All users have a uniform false negative rate $x^* \in [0, 1]$ that maximizes the defender's expected utility. We discretize the interval [0,1] into 1000 equal-distance points and search among these points to find the optimal value x^*. In addition, we will use x^* as the starting point when solving PEMS. **Laszka et al.** (Laszka, Vorobeychik, and Koutsoukos 2015): An existing approach for personalized threshold setting assumes that the defender's expected loss from spear phishing attacks is the sum of users' individual expected losses. Following our notations, user u's individual loss is set to the immediate expected loss $x_u a_u k_u L$ in the single-credential case and $x_u a_u \sum_{h \in H} m_u^h L_h$ in the multiple-credential case.

Scalability Analysis We first evaluate the scalability of PEDS and PEMS. We assume that each credential can only be accessed by 30% of total users with nonzero probability considering that sensitive information is usually accessed by a small portion of total users. Figure 2(a) shows that PEDS can solve games with 70 users in 23s. Figure 2(e) shows that both the number of users and the number of credentials have significant influence on the runtime of PEMS. PEMS runs slower than PEDS since nonlinear programs are usually more computationally consuming. However, we argue that both PEDS and PEMS are applicable in real-world cases due to two reasons. First, spear phishing attacks, unlike mass attacks, usually jeopardize a small group of people. For example, in the attack towards the US Nuclear Regulatory Commission, only 16 employees are targeted (Rosenblatt 2014). Second, in our model the defender does not need to update her strategy adaptively so that the runtime requirement is not very high.

Solution Quality Comparisons We compare our approaches with two benchmarks for different values of $\rho^{\mathcal{T}}$, which measures the probability that spear phishing attacks happen in $\mathcal{T}$. Note from Figure 2(b) and Figure 2(f), when $\rho^{\mathcal{T}} = 0$, meaning that there is no spear phishing attacks, our approaches lead to the same defender utilities as Laszka et al.. In this case the defender's optimal strategy is simply setting $x_u = \arg\max_x \Lambda_u(x)$ for each user u, considering only mass attacks and false positives. With $\rho^{\mathcal{T}}$ growing, Laszka et al. performs significantly worse than our approaches.

Our approaches outperform the optimal uniform strategy. This is because that the optimal uniform strategy is computed under the constraints that all users thresholds are equal. We compare our approaches with the optimal uniform strategy to show how much improvement personalization can bring. Our approaches also outperform Laszka et al.. The reason is, when computing defender strategy of Laszka et al., the attacker is assume to launch a non-sequential attack. It's not surprising that this strategy performs poorly when against a sequential decision making attacker. Moreover, note from Figure 2(b) that Laszka et al. performs even worse than the optimal uniform strategy when $\rho^{\mathcal{T}} > 0.5$. This indicates that estimation about the attacker's behaviour may be even more important than "personalization".

Robustness Analysis Defender's estimation of the attack cost and user susceptibility may not be perfect. We consider a noise on c_u and a_u. In this section of experiments, estimations of c_u are drawn uniformly from two intervals $c_u \cdot [1-5\%, 1+5\%]$ and $c_u \cdot [1-10\%, 1+10\%]$. Estimations of a_u are drawn from $a_u \cdot [1-5\%, 1+5\%]$ and $a_u \cdot [1-10\%, 1+10\%]$. We use these estimations to compute the defender strategy and then use this strategy to compute the defender's utility in the accurate parameter setting. Figure 2(c) (Figure 2(g)) shows that PEDS (PEMS) outperforms both benchmarks even with a 10% error range on attack cost c_u in single-credential (multiple-credential) case. Similarly, Figure 2(d) (Figure 2(h)) shows that PEDS (PEMS) outperforms both benchmarks w.r.t. the susceptibility measurement a_u in single-credential (multiple-credential) case.

Conclusion

This paper studies the problem of setting personalized email filtering thresholds against sequential spear phishing attacks. We first consider a simple single-credential case and then extend it to a more general multiple-credential case. Our approach features the following novelties. (1) An MDP framework is proposed to model the sequential decision making attacker. (2) An efficient binary combinatorial optimization formulation PEDS is proposed for computing solutions for the single-credential case. (3) With multiple credentials, the defender's loss from spear phishing attacks is represented by a linear combination of dual variables. (4) A single level formulation PEMS, which is reduced from the defender's bilevel program using complementary slackness conditions, is proposed for computing solutions for the multiple-credential case.

Acknowledgement

This material is based upon work supported by the National Science Foundation under Grant No. IIS-1253950.

References

Bergholz, A.; De Beer, J.; Glahn, S.; Moens, M.; Paaß, G.; and Strobel, S. 2010. New filtering approaches for phishing email. *Journal of computer security* 18(1):7–35.

Borkar, V., and Ghosh, M. 1992. Stochastic differential games: Occupation measure based approach. *Journal of optimization theory and applications* 73(2):359–385.

Bouckaert, R. R. 2006. Efficient auc learning curve calculation. In *AI 2006: Advances in Artificial Intelligence*. Springer. 181–191.

Choo, K. R. 2011. The cyber threat landscape: Challenges and future research directions. *Computers & Security* 30(8):719–731.

Deshmukh, P.; Shelar, M.; and Kulkarni, N. 2014. Detecting of targeted malicious email. In *IEEE Global Conference on Wireless Computing and Networking (GCWCN'14)*, 199–202.

Fawcett, T. 2006. An introduction to ROC analysis. *Pattern recognition letters* 27(8):861–874.

Gan, J.; An, B.; and Vorobeychik, Y. Security games with protection externalities. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI'15)*, 914–920.

Hlatky, P. 2015. How does yesware tracking work? http://www.yesware.com/blog/how-does-yesware-tracking-work/.

Jagatic, T. N.; Johnson, N. A.; Jakobsson, M.; and Menczer, F. 2007. Social phishing. *Communications of the ACM* 50(10):94–100.

Kelley, P. G. 2010. Conducting usable privacy & security studies with amazons mechanical turk. In *Proceedings of the 6th Symposium on Usable Privacy and Security (SOUPS'10)*.

Korzhyk, D.; Yin, Z.; Kiekintveld, C.; Conitzer, V.; and Tambe, M. 2011. Stackelberg vs. Nash in security games: An extended investigation of interchangeability, equivalence, and uniqueness. *Journal of Artificial Intelligence Research* 41(2):297–327.

Laszka, A.; Vorobeychik, Y.; and Koutsoukos, X. 2015. Optimal personalized filtering against spear-phishing attacks. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI'15)*, 958–964.

Rosenblatt, S. 2014. Nuclear regulator hacked 3 times in 3 years. http://www.cnet.com/news/nuclear-commission-hacked-3-times-in-3-years/.

Schweitzer, P. J., and Seidmann, A. 1985. Generalized polynomial approximations in Markovian decision processes. *Journal of mathematical analysis and applications* 110(2):568–582.

Sheng, S.; Kumaraguru, P.; Acquisti, A.; Cranor, L.; and Hong, J. 2009. Improving phishing countermeasures: An analysis of expert interviews. In *Proceedings of the 4th APWG eCrime Researchers Summit*, 1–15.

Sheng, S.; Holbrook, M.; Kumaraguru, P.; Cranor, L. F.; and Downs, J. 2010. Who falls for phish? A demographic analysis of phishing susceptibility and effectiveness of interventions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 373–382.

TrendLabs. 2012. Spear-phishing email: Most favored APT attack bait. Technical report, Trend Micro.

Varma, R. 2010. Combating Aurora. Technical report, McAfee Labs.

Watson, G.; Mason, A.; and Ackroyd, R. 2014. *Social Engineering Penetration Testing*. Elsevier. chapter 4, 71–74.

Yin, Y.; An, B.; and Jain, M. 2014. Game-theoretic resource allocation for protecting large public events. In *Proceedings of the 28th AAAI Conference on Artificial Intelligence (AAAI'14)*, 826–834.

Yin, Y.; Xu, H.; Gan, J.; An, B.; and Jiang, A. X. 2015. Computing optimal mixed strategies for security games with dynamic payoffs. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI'15)*, 681–687.

Zetter, K. 2011. Researchers uncover RSA phishing attack, hiding in plain sight. http://www.wired.com/2011/08/how-rsa-got-hacked/.