Privacy Issues in Light of Reconnaissance Attacks with Incomplete Information

Xiang Li, J. David Smith
CISE Department
University of Florida
Gainesville, FL, 32601
Email: {xixiang, jdsmith}@cise.ufl.edu

Thang N. Dinh
CS Department
Virginia Commonwealth University
Richmond, VA 23284
Email: tndinh@vcu.edu

My T. Thai
CISE Department
University of Florida
Gainesville, FL, 32601
Email: mythai@cise.ufl.edu

***Abstract*—A reconnaissance attack, in which attackers lure targets into becoming their friends in order to extract victims' sensitive information for sale or use in future attacks, is one of the most dangerous attacks in social networks. The core of this attack lies in intelligently sending friend requests to a small subset of users, called Critical Friending Set (CFS), so that the attacker can evade current defense mechanisms.**

Motivated by the above, we present a new paradigm to measure OSN vulnerability in light of reconnaissance attacks. Specifically, we introduce a new optimization problem, namely Min-Friending, which identifies a minimum CFS to friend in order to obtain at least Q benefit, in terms of personal information. A significant challenge of this problem is that network information (i.e. who friends with whom) is generally unknown to attackers. In this paper, we show that Min-Friending is inapproximable within a factor of $(1 - o(1)) \ln Q$ and present an adaptive algorithm which has a tight performance bound of $(1+\ln Q)$ using adaptive stochastic optimization. The key feature of our solution lies in the adaptive method, where partial network topology is revealed during each successful friend request. Thus the decision of sending each friend request is made taking into account observation about the outcomes of past decisions.

***Index Terms*—Target Reconnaissance Attacks; Social Networks Analysis; Adaptive Algorithms; Incomplete Topology.**

I. Introduction

The centralization of human interaction online into a few enormous Online Social Networks (OSNs) has created a rich repository of personal information that criminals seek to harvest. During the last two decades, we have witnessed the growth of a variety of attacks, most of which follow a conventional method of befriending target users [1]. After gaining access to their social circles, the attackers can extract sensitive information that may be exploited for a number of purposes including spear phishing and account compromise via security questions [2]. Therefore, studying such threats against privacy issues is of great importance in aiding the development of new forms of protection as well as in raising users awareness of the online threats.

Recently, a social engineering attack has emerged enabling automated infiltration of social networks [2]. In this reconnaissance attack, attackers attempt to approach their targets by sending friend requests to multiple users in the target's social circle. Only when attackers acquire enough mutual friends with the target do they attempt to lure the target in accepting their friend request. When compared to sending friend requests to the target immediately, where the attacker has no mutual friend with the target, this attack guarantees a much higher success rate [3]. One special feature of this attack is that it can avoid detection by the social network provider for two reasons: 1) Attackers carefully select certain friends of the target to approach without sending too many friend requests at once, and 2) This attack does not create a large sub-graphs that could be detected by existing methods such as Sybil defenses [4].

Motivated by the above discussion, we present a new paradigm to measure the OSN vulnerability in light of reconnaissance attacks. Although quantitative analysis of network vulnerability can be addressed from a variety of perspectives, an intuitive measure is the minimum number of users that attackers need to friend with in order to maximally collecting private information of target sets. Obviously, if this number is small compared to the benefit gain from collecting those information, one can conclude that the network is vulnerable to attacks, whereas if this number is large, then such a network is more robust with respect to reconnaissance attacks. Identifying the set of users to friend, called a *critical friending set (CFS)*, is useful from both attack and defense perspectives. In the former, the attacker identifies an optimal set of users to friend, whereas in the latter the defender has an opportunity to protect this set and potentially interfere with attacks in order to protect the targets' privacy.

The realization of this study, however, encounters several challenges. First of all, the topological information of social networks between the attacker and the target is generally unavailable. Since only two-hop topology is available by default in closed OSNs such as Facebook, connections are gradually revealed as the attacker acquires new friends. Furthermore, the reconnaissance attack is still in its infancy, based only on simulation and case studies without theoretical analysis. Additionally, the huge number of OSN users and amount of data available on OSNs pose a substantial challenge to mine the critical friending set with incomplete topology. Finally, the variety of potential social responses to friend requests make it difficult to design an efficient reconnaissance attack, and thus challenging to identify the CFS.

In this paper, we hope to make the first step towards privacy protection by introducing a new optimization problem, called *Adaptive Minimum Critical Friending Set (Min-Friending)*. The problem asks us to find a minimum number of users to friend with in order to obtain at least Q benefits, in terms of personal information, which will be formally defined later. The

978-1-5090-4470-2/16 $31.00 © 2016 IEEE
DOI 10.1109/WI.2016.50

key feature of this problem lies in the adaptive solution, where the partial network topology is revealed during each successful friend request. That is, the decision of sending friend request is made, taking into account the outcomes of past decisions. Our contributions are summarized as follows:

- We prove Min-Friending cannot be approximated within the ratio of $(1 - o(1)) \ln Q$ unless $NP \subset DTIME(n^{O(loglogn)})$. That is, no-one can design an algorithm which can guarantee its solution is within $(1 - o(1)) \ln Q$ factor from the optimal solution under all instances.
- We design an efficient heuristic, named AReST, to Min-Friending. Using adaptive stochastic optimization, we show that AReST has a tight performance ratio of $(1 + \ln Q)$ in some cases.
- We have conducted extensive experimental evaluations showing that AReST outperforms several alternate methods. Further, we find that the effectiveness of infiltration has strong dependencies on the network topology and user behavior in addition to the attacker's choice of target.

Related Work. Reconnaissance attacks on social networks may be simple attempts to collect personally identifiable information, but have shown to be very crucial [5]. Ryan & Mauch showed that fake profiles can be effectively used to befriend members of the NSA, military intelligence agencies, and Global 500 corporations [3]. These fake profiles can be automated to form socialbots [2], which can then be used to automatically infiltrate organizations [6], [7]. Furthermore, reconnaissance attacks preclude the use of Sybil defenses, which have a significant body of work [4] (and references therein). Therefore, it urgently calls for radically new models and analytical techniques to assess OSNs against these attacks.

The only work that is relevant to defending against these attacks is in [8], by monitoring a subset of users within an organization, and evaluated the cost to the organization by simulating an attacker on topologies taken from social networks. However, this and much of the related work are based on heuristics built on known sociological properties. Instead, we provide a better defense strategy with a guarantee performance bound of $(1+\ln(Q))$. Additionally, in the literature the attacker is usually assumed to have knowledge of the network topology. We advance the state of the art by relaxing this to the case of incomplete topology, where the attacker possesses only partial knowledge of the edges on the network, which matches the reality of infiltration on closed networks such as Facebook.

Further, assessing network vulnerability to infiltration and reconnaissance attacks has not been addressed. The vulnerability of networks due to malicious actors or external disasters has been characterized in a number of ways [9]–[11] (and references therein). However, the differences between these destructive attacks and reconnaissance attacks precludes applying these vulnerability measurements directly. Where prior work is concerned with the ability of an attacker to disrupt a network in term of network connectivity for example, we instead look at the ability of an attacker to extract information from it, thereby forming a new research direction.

Organization. The rest of the paper is organized as follows. Section II presents the social network models and our problem definition. The inapproximablity of Min-Friending is proven in Section III. The AReST algorithm and its theoretical performance analysis are introduced in Section IV. Section V presents our experimental evaluation and Section VI concludes the paper.

II. Network Model and Problem Definition

A. Reconnaissance Insights

In order to solve Min-Friending, we first need to understand reconnaissance attacks. To generalize our problem, we consider a target set of users T that an attacker want to collect information from. T could be a set of employees in a targeted organization, one individual with a high profile ($|T| = 1$), or the set of all users in a network. For simplicity, we assume that there is a single attacker s who is an online user in the same networking environment. This can be a socialbot or a fake account created by the attacker. The solution proposed in this paper can be easily extended to handle multiple attackers.

There are privacy settings in OSNs that allow account owners to specify who can see what in their accounts. For example, in Facebook, users can specify who are able to see their friends list based on three categories: Public (everybody), Friends (only whom you are connected with) and Only Me (nobody else but me). Whereas, LinkedIn only has two options, i.e., Only you or Your connections (Friends). Most of the time, the default privacy setting is Friends.

Conceptually, the reconnaissance attack works as follows. Attacker s first achieves a master-list of target users T through some public channels, e.g., organizations website or the OSNs themselves which make public a certain amount of personal information. This was made even easier after popular social networks, e.g., Facebook, began forcing users to use their original name for their accounts. Because of these privacy settings, the only way s can gather the information of $t \in T$, assuming t has privacy setting to "Friends", is friending t. To successfully friend t, s needs to achieve the following: 1) s needs to mimic a normal user, and thus needs to have a few friends initially. This can be easily done by sending friend requests to users that have a high number of friends as they tend to accept all friend requests [2], [6]. 2) Since s and t usually have no mutual friends, the probability of t accepting s's friend request is very low. And thus s should attempt to friend t's friends first, which in turn means, s should send friend requests to friends of friends of t, and so on.

However, s cannot send too many friend requests as he will easily get detected by any network monitoring service/manager (e.g. by employing anomaly detection). The best strategy for s is to mimic normal user behavior by sending friend requests to a small number of users, observing the response and then sending to another set of users. Once a user v accepts the friend request of s, s can collect all v's information, and all the friends of v becomes visible of s. This strategy of repeating the process of making decisions subject to previous decisions and observing new results is called an adaptive strategy.

Based on the above discussion, the central concern of any reconnaissance attack is who s should select in each decision making step to minimize the number of friend requests while

successfully gather information about T. Solving this problem is equivalent to solve our Min-Friending problem.

Two fundamental challenges in solving Min-Friending are as follows. First of all, the attacker's knowledge of the network topology (who is friends with whom) is usually incomplete due to privacy settings. The problem becomes much more challenging when the probability of accepting the friend request from s adaptively changes since the number of mutual friends of s and targeted users is dynamically increasing.

B. Network Model

From the aforementioned insights, we abstract an OSN as a directed graph $G = (V, E)$ where $V = \{v_1, v_2, \ldots, v_n, s\}$ is the set of n users and attacker s, who initially has no connections to other users. E is the set of m directed edges where each edge $(u, v) \in E$ represents the friendship between u and v. Note that due to the privacy settings, the friendship information (network topology) is incomplete. Instead, s can estimate these friendship probabilities based on link prediction methods [12]–[14] which may combine both the publicly observable connections and users public profiles. Therefore, we model this by letting each edge $e \in E$ exist with some probability $p_e \in [0, 1]$. Once v accepts the friend request of s and all of its friends are visible to s, then $p_{uv} = 1$ iff u is v's friend. Else $p_{uv} = 0$.

Friend Request Acceptance Model. Let $accept(u)$ denote the probability that u accepts the friend request from s. This function $accept(u)$ is varied due to the social behaviors of users. For example, when u has a very high number of friends in his circle, $accept(u)$ tends towards 1 [6]. Boshmaf et al. found that increasing the number of mutual friends dramatically increased the friend acceptance rate on Facebook, which they explain as a result of the Triadic Closure principle [2]. We fit their data to a degree-1 polynomial with a natural log term. Figure 1 shows the original data and the estimated function. Based on this fitting, we use this as the friend request acceptance model in our paper, which is defined as follows:

$$\text{accept}(u) = \rho_1 \log(\mathbb{E}\left[|N(u) \cap N(s)|\right] + 1) + \rho_0$$

with $\rho_1 = 0.22805837$ and $\rho_0 = 0.18014571$. $N(.)$ denotes the set of neighbors. In a more general sense, this formula incorporates the willingness of a user to accept a new, unknown friend (ρ_0) and how much sharing mutual friends improves that willingness (ρ_1). Given the limited amount of data available learning the distribution of per-user weights is currently infeasible, though we conduct experiments in the special case of each user u having independent $\rho_0(u)$ weights.

Information Benefit Model. In order to quantify the benefit that s achieves by gathering the information from the target set T, each node u is associated with a benefit $B_{fof}(u) \in \mathbb{R}^+$ when u becomes a friend of friend of s, i.e., 2 hops away from s. Each node u is also associated with a benefit $B_f(u) \geq B_{fof}(u), B_f(u) \in \mathbb{R}^+$ when u becomes a friend of s. Note that when u is both friend and a friend of friend of s, only the friend benefit $B_f(s)$ is in effect. Moreover, when each edge $(u, v) \in E$ is revealed, (i.e.the attacker learns about the existence of (u, v)), the attacker gains an information benefit $B_i(u, v) \in \mathbb{R}^+$. The existence of edge (u, v) is revealed only when either node u or v becomes a friend of s. At this point, $p_{uv} = 1$.

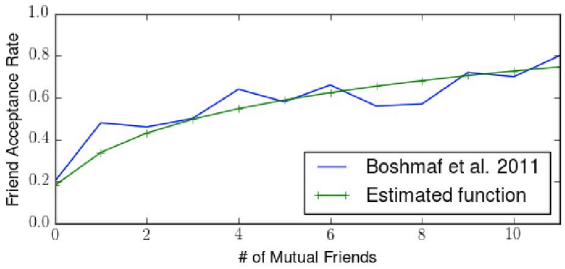


Fig. 1: The friend acceptance rate from the experiments of Boshmaf et al. [2] as a function of the number of mutual friends, with a logarithmic function fit to the data.

C. Problem Definitions and Formulations

Based on the above model, the goal of attacker s is to gain the greatest total benefit with the minimum number of friend requests. Accordingly, we study the following problem:

Definition 1 (Adaptive Minimum Critical Friending Set (Min-Friending))**.** *Given a social network $G = (V, E, p, B, accept)$ where V is the set of user accounts, E is a set of possible friendships between users, each edge $e \in E$ exists with a probability $p_e \in [0, 1]$, a target set $T \subseteq V$, and a threshold $Q \in \mathbb{Z}^+$. The benefit function B and acceptance probability function $accept(.)$ are defined earlier. The problem asks us to find a set of friending nodes $F \subset V$ with minimum size so as when s successfully friends with F, the total expected benefit gain is greater than Q.*

Note that finding F is equivalent to finding an adaptive attack strategy π, in which s will friend with $u \in F$ iteratively. Each time s becomes a friend of u, the network topology G will be updated to reveal all edges incident with u. As $|F|$ is minimized, the number of friend request steps is also minimized.

Since G is partially unknown to s and friend requests sent from s to u may fail, we use adaptive stochastic optimization to tackle our problem. We begin by introducing notations. For each node $u \in V$, let $X_u \in \{0, 1, ?\}$ denote the state of u where 1 indicates that u accepts the friend request from s, 0 indicates that u rejects the friend request, and ? represents an unknown, i.e., s has not sent a request to u yet. Initially, the states of all u should be ?. Likewise, for each edge $(u, v) \in E$, define $Y_{uv} \in \{0, 1, ?\}$. 1 means the edge (u, v) exists (revealed when s friends with u and v is u's friend), 0 indicates edge (u, v) is not present (revealed when s friends with u and we learn for certain that v is not a friend of u), and ? means unknown, i.e. u rejects the friend request from s, or s has not sent a friend request to u yet, or u has the privacy setting to himself only (not friend of friend). Let Ω be the state of all possible states of G and $\phi = \{X_v\}_{v \in V} \cup \{Y_{uv}\}_{(u,v) \in E} \rightarrow \Omega$ be a possible state, called a *realization*. Thus we call $\phi(u, e)$ is the state of node u and edge e under realization ϕ. Without abusing the notation, we use $\phi(u)$ and $\phi(e)$ to denote the state

of node u and edge e under ϕ respectively. We require each realization to be consistent. That is, each node and edge must be in only one of the states $\{0, 1, ?\}$. Clearly there are many possible realizations which follow a probability distribution $P[\phi]$. We denote Φ as a random realization and $P[\phi] = P[\Phi = \phi]$ over all realizations.

We will consider the problem where s sequentially sends a friend request to u, sees its state $\Phi(u, e)$ for all e incident to u (ie, see whether u accepts the friend request, if so reveal all its neighbors), pick the next user to friend, see its state, and so on. We use the notation $F(\pi, \phi)$ be a set of node selected by strategy π under realization ϕ. After each friend request, our observations thus far can be represented as a *partial realization* ω. We use $dom(\omega)$ to refer to the domain of ω, ie., the set of nodes and edges observed in ω. A partial realization ω is consistent with a realization ϕ if they are equal everywhere in the domain of ω, written as $\phi \sim \omega$. If ω and ω' are both consistent with some ϕ and $dom(\omega) \subseteq dom(\omega')$, we say ω is a subrealization of ω'.

Let π be an adaptive attack strategy of s. The total benefit gain from this strategy π under realization ϕ can be formulated as follows.

$$f(\pi, \phi) = \sum_{u \in N_f(\pi,\phi)} B_f(u) + \sum_{v \in N_{fof}(\pi,\phi)} B_{fof}(v) + \sum_{(u,v) \in N^i(\pi,\phi)} B_i(u, v) \quad (1)$$

where $N_f(\pi, \phi) = \{u | u \in N(\pi, \phi), \phi(u) = 1\}$, $N_{fof}(\pi, \phi) = \{v | \exists u \in N(\pi, \phi) : \phi(u, v) = 1\} \backslash N_f(\pi, \phi)$, and $N_i(\pi, \phi) = \{(u, v) | u \in N_f(\pi, \phi), \phi(u, v) = 1\}$.

Therefore, the Min-Friending problem can be stated formally as:

$$\min \mathbb{E}[F(\pi, \Phi)] \quad (2)$$
$$s.t.\ \mathbb{E}[f(F(\pi, \Phi), \Phi)] \geq Q$$

III. Inapproximability

Instead of proving that Min-Friending is NP-hard, we prove a stronger theorem, showing the inapproximability of Min-Friending.

Theorem 1. *The Min-Friending problem cannot be approximated within a factor* $(1 - o(1)) \ln Q$ *unless* $NP \subset DTIME(n^{O(loglogn)})$

Proof. Let $\Pi = (\mathcal{S}, \mathcal{U}, K)$ be an instance of the set-cover problem in which $\mathcal{U} = \{e_1, e_2, \ldots, e_n\}$ is the set of n elements and $\mathcal{S} = \{S_1, S_2, \ldots, S_m\}$ is the collection of m subsets of $\mathcal{U}$. The set cover problem asks if there are k subsets which cover at least $K \leq n$ items in $\mathcal{U}$. We construct an instance $\Pi' = G(V, E, p, B, accept, Q)$ of the Min-Friending problem as follows.

- For each element $e_i \in \mathcal{U}$, we include into V a vertex v_i. Similarly, we put into V a vertex u_j for each subset $S_j \in \mathcal{S}$.
- For each pair of element e_i and subset S_j, we connect u_j and v_i if $e_i \in S_j$. For all edges $(u_j, v_i) \in E$, we set $p_{u_j v_i} = 1$.

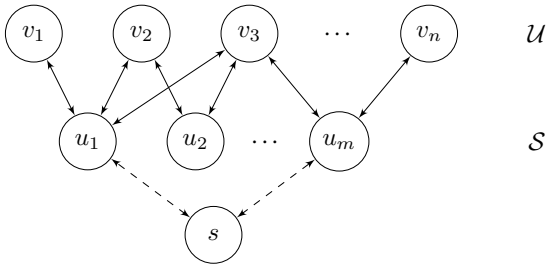


Fig. 2: An instance Π' of Min-Friending after the attacker s friends u_1 and u_m (corresponding to S_1 and S_m).

- For benefit B, we set $B_{fof}(v_i) = 1$ and $B_f(v_i) = 0$ for each $v_i \in V$ that corresponds to $e_i \in \mathcal{U}$. And we set $B_f(u_j) = B_{fof}(u_j) = 0$ for all u_j, associated with subset S_j, and $B_i(u_j, v_i) = 0$ for all edges in the graph.
- For $accept(.)$, set $accept(u_j) = 1\ \forall u_j$, set $accept(v_i) = \rho_1 \log(\mathbb{E}[|N(v) \cap N(s)|] + 1) + \rho_0\ \forall v_i$.
- Finally, set $Q = K$ and the target set $T = V$.

So the Min-Friending problem asks us if there exists a CFS of size q such that the total benefit is at least Q. The construction is illustrated in Fig. 2.

Since $B_f(v_i) = 0$, there is no incentive to friend v_i. Thus attacker s will friend with u_j. This friend request to u_j is always successful as $accept(u_j) = 1$, s. In order to have $Q = K$ benefit, s needs to have at least K v_i in his two-hop neighbors. Then the users u_j that s chooses to friend with are corresponding to the k sets S_j of Π. Clearly if we have an approximation algorithm with an approximation factor $\alpha(Q)$ for Min-Friending, then we also have an $\alpha(K)$ approximation algorithm for the set cover problem. Thus, due to the inapproximability of set cover [15], the Min-Friending problem cannot be approximated within a factor $(1-o(1)) \ln Q$ unless NP has $n^{O(\log \log n)}$ deterministic time algorithms. □

IV. Adaptive Reconnaissance Strategies

In this section, we present our solution to Min-Friending, namely Adaptive Reconnaissance Strategy algorithm (AReST), followed by the theoretical analysis.

A. Algorithm Description

At an abstract level, the Adaptive Reconnaissance Strategy algorithm (AReST) has two main phases: *Selection* and *Feedback*. At the *Selection* phase, AReST will select a node u for s to friend so as to increase the potential function (which will be discussed later) the most. After selecting u, a friend request is sent. If u accepts the friend request, AReST executes the *Feedback* phase, which will (1) update the network topology with more exact information on p_e ; and (2) update the $accept(v)$ for all $v \in N(u)$. If u rejects the friend request, AReST will continue the first phase to select another node. These two phases will be iteratively executed until the total expected benefit exceeds Q.

The main challenge of the first phase is to define an efficient potential function. As the friend request acceptance probability is changing after each successful friend request, the potential function must account for the likelihood of increasing the

Algorithm 1: Adaptive Reconnaissance Strategies (AReST)

Input: Graph $G = (V, E, p, B, accept)$, and $Q \in \mathbb{Z}^+$
Output: An ordered set of nodes $F \subset V$ for s to friend with.

$F \leftarrow \emptyset; \omega \leftarrow \emptyset$
while $\mathbb{E}[f(F)] < Q$ **do**
  foreach $u \in V \backslash F$ **do**
    $\Delta(u|\omega) = accept(u)(P_1 + P_2)$
  Select $u^* \in \arg\max_u \Delta(u|\omega)$
  Set $F \leftarrow F \cup \{u^*\}$
  Send a friend request to u^*
  Feedback: Update ω with new observed information of p_{u^*v} and $accept(v)$ for all $v \in N(u^*)$
Return F

acceptance probability in a later iteration in addition to the gain defined by the benefit function B. Let F denote the set of s' friends at this current stages ω. In order to select a node u at the next step, we define the potential function as follows:

$$\Delta(u|\omega) = accept(u)(P_1 + P_2)$$

where P_1 and P_2 represent the gain in increasing the acceptance probability for later stages and the gain in increasing the benefit function B, respectively. Mathematically, we have:

$$P_1 = \frac{1}{|N(u)|} \sum_{v \in N(u)} p_{uv} \times \Delta P_u(s, v) \times \Delta_{uv} B$$

where $\Delta P_u(s, v)$ denotes the gain in the acceptance probability when u becomes a friend of s. $\Delta P_u(s, v)$ can be calculated based on the definition of $accept(.)$ function: $\Delta P_u(s, v) = \rho_1 \log(1 + 1/\mathbb{E}[|N(u) \cap N(s)|])$. In the special cases of u placing low value on mutual friends or u having many friends, this tends to 0. $\Delta_{uv} B$ represents the benefit gain assuming s adds u as a friend, and then add v as a friend. Thus $\Delta_{uv} B = f(\omega \cup \{u, v\}) - f(\omega \cup \{u\})$ and

$$P_2 = \sum_{\omega(u)=1} B_f(u) + p_{uv} \left(\sum_{v \in N(u)} B_{fof}(v) + \sum_{(u,v) \in E} B_i(u, v) \right)$$

Algorithm AReST is depicted in Algorithm 1.

B. Performance Analysis

We are going to analyze the performance of AResT where $\Delta P_u(s, v) = \rho_1 \log(1 + 1/\mathbb{E}[|N(u) \cap N(s)|]) = 0$. This relates to u placing low value on mutual friends or u having many friends, and thus their friend acceptance probility depends on $\rho_0(u)$. We show that AReST has an approximation ratio of $(1 + \ln Q)$. As shown in Theorem 1, this ratio is tight.

Note that AReST calculates $\Delta(u|\omega)$ for all $u \in V \backslash F$ and chooses u^* with the maximal gain over all realization. Thus in this case, the expected marginal gain of u conditioned on having partial realization ω is defined as follows:

$$\Delta(u|\omega) = \mathbb{E}[f(dom(\omega) \cup \{u\}, \Phi) - f(dom(\omega), \Phi) | \Phi \sim \omega]$$

Lemma 1. *The function f of the Min-Friending problem is strongly adaptive monotonicity.*

Proof. Recall that a function $f(.)$ is strongly adaptive monotone with respect to the distribution $P(\phi)$ if the following condition holds [16]. For all ω, all $v \notin dom(\omega)$, and all possible states o of node v such that $P[\Phi(v) = o | \Phi \sim \omega] > 0$, we have:

$$\begin{aligned} &\mathbb{E}[f(dom(\omega), \Phi) | \Phi \sim \omega] \\ &\leq \mathbb{E}[f(dom(\omega) \cup \{v\}, \Phi) | \Phi \sim \omega, \Phi(v) = o] \end{aligned} \tag{3}$$

Consider a fixed ω, $v \notin dom(\omega)$, and status o. Let $A(\omega)$ be a set of nodes and edges that can be reached from s after selecting $dom(\omega)$ and observing ω. Clearly, for all paths from s to $u \in A(\omega)$ consisting of $\omega(e) = 1$. Therefore, every path from any $u \in A(\omega)$ to any $v \in V \backslash A(\omega)$ must consisting at least one $\omega(e) \neq 1$ or $\omega(w) \neq 1$ for some w on the path. Thus we have $f(A(\omega)) = \mathbb{E}[f(dom(\omega), \Phi) | \Phi \sim \omega]$.

With a similar argument, we have $f(A(\omega \cup \{v\})) = \mathbb{E}[f(dom(\omega) \cup \{u\}, \Phi) | \Phi \sim \omega, \Phi(u) = o]$. Note that $\omega \subseteq \omega'$ implies $A(\omega) \subseteq A(\omega')$. Since f is a monotone function by definition, we have $f(A(\omega)) \leq f(A(\omega'))$. Thus we obtain $\mathbb{E}[f(dom(\omega), \Phi) | \Phi \sim \omega] \leq \mathbb{E}[f(dom(\omega) \cup \{v\}, \Phi) | \Phi \sim \omega, \Phi(v) = o]$. This completes the proof. □

Lemma 2. *The function f of the Min-Friending problem is adaptive submodular.*

Proof. Recall that a function $f(.)$ is adaptive submodular with respect to the distribution $P[\phi]$ of all realizations if the conditional expected marginal gain of any fixed node does not increase as more nodes are selected and their states are observed. Formally, f is adaptive submodular w.r.t. $P[\phi]$ if for all ω and ω' such that $\omega \subseteq \omega'$ and for all $v \in V \backslash dom(\omega')$, we have:

$$\Delta(v|\omega) \geq \Delta(v|\omega') \tag{4}$$

Consider two fixed partial realizations ω and ω' where $\omega \subseteq \omega'$ and a node $v \in V \backslash dom(\omega')$, we need to prove that $\Delta(v|\omega) \geq \Delta(v|\omega')$. We first prove the following claim:

Given $\omega \subseteq \omega'$ and define a coupled distribution μ over pairs of realization $\phi \sim \omega$, $\phi' \sim \omega'$ such that $\phi(v) = \phi'(v)$ for all $v \notin dom(\omega')$. For all (ϕ, ϕ') in support of μ, we have:

$$\Delta(v|\omega, \phi \sim \omega) \geq \Delta(v|\omega', \phi' \sim \omega')$$

where $\Delta(v|\omega, \phi) = f(dom(\omega) \cup \{v\}, \phi) - f(dom(\omega), \phi)$.

Define $A(\omega)$ and $A(\omega')$ as in the proof of Lemma 1. We have:

$$\begin{aligned} \Delta(v|\omega, \phi) &= f(dom(\omega) \cup \{v\}, \phi) - f(dom(\omega), \phi) \\ &= f(A(\omega) \cup \{(v, \phi(v))\}) - f(A(\omega)) \\ &\geq f(A(\omega') \cup \{(v, \phi'(v))\}) - f(A(\omega')) \\ &= f(dom(\omega') \cup \{v\}, \phi) - f(dom(\omega'), \phi') \\ &= \Delta(v|\omega, \phi \sim \omega) \end{aligned}$$

Having proven the above claim, we can straightforwardly finish our proof. Since $\omega \subseteq \omega'$, we have:

$$\begin{aligned}\Delta(v|\omega) &= \mathbb{E}[f(dom(\omega) \cup \{v\}, \Phi) - f(dom(\omega), \Phi)|\Phi \sim \omega] \\ &= \sum_{(\phi,\phi')} \mu(\phi, \phi')\Delta(v|\omega, \phi) \\ &\geq \sum_{(\phi,\phi')} \mu(\phi, \phi')\Delta(v|\omega', \phi') \\ &= \Delta(v|\omega')\end{aligned}$$

□

Theorem 2. *The AReST algorithm has an approximation ratio of* $(1 + \ln Q)$.

Proof. According to [16], if f is strongly adaptive monotonicity, adaptive submodular, and self-certifying, the the greedy algorithm to the Min-Friending problem returns a $(1 + \ln Q)$-approximation. Therefore, the only thing left we need to prove is that f is self-certifying, which is defined as follows. An instance $(f, P(\phi))$ is self-certifying if for all ϕ, ϕ', and ω such that $\phi \sim \omega$ and $\phi' \sim \omega$, we have $f(dom(\omega), \phi) = f(V, \phi)$ iff $f(dom(\omega), \phi') = f(E, \omega')$.

Clearly, we have $f(V, \phi) = \min\{Q, f(V)\} = Q$. We have shown that $f(dom(\omega), \phi) = f(A(\omega))$ for every $\phi \sim \omega$. It follows that $f(dom(\omega), \phi) = f(dom(\omega), \phi')$ for all ω and ϕ, $\phi' \sim \omega$. Therefore, we obtain $f(dom(\omega), \phi) = Q$ iff $f(dom(\omega), \phi') = Q$, which completes the proof. □

V. Experimental Evaluations

In this section, we evaluate the resilience of several networks (shown in Table I) to reconnaissance attacks. We describe our experimental setup in Section V-A, then evaluate AReST in relation to several simple alternatives in Section V-B. We then measure the resistance of each network to reconnaissance attacks and discuss the impact of user behavior in Section V-C.

A. Experiment Setup

We evaluate AReST by simulating it on a set of networks taken from the Stanford Large Network Dataset Collection[1]. We compare against three alternate algorithms with multiple choices for accept(u) and multiple kinds of structure in the target set. The alternate algorithms we use to select friends are 1) random selection, 2) greedily choosing the node with highest degree, and 3) greedily choosing the node with highest PageRank.

In our experiments, the benefit functions were fixed as $B_f(u) = t_u$, $B_{fof}(u) = t_u/2$, and $B_i(u, v) = t_*/M$, where $t_u = 1$ iff u is targeted, 0 otherwise; t_* is 1 when neither u, or v is targeted, 2 when either is, and 4 when both are; and $M = \max_{v \in V} \mathbb{E}[\deg(v)]$ is the maximum expected degree of any node in V.

[1] https://snap.stanford.edu/data/

Network	Nodes	Edges
Facebook	4k	88k
Enron Email	37k	184k
Slashdot	77k	905k
Twitter	81k	1.77M

TABLE I: The networks used in our simulations.

1) Choice of Acceptance Function: As mentioned previously, the exact form of the function accept(u) is not known. Therefore, we run our experiments with several acceptance functions: constant (accept$(u) = \rho_0(u) \in [0, 1)$), expected fraction of shared neighbors (ESN; eqn. 5), and the expected triadic closure (ETC) function defined in section II.

$$\text{accept}(u) = \mathbb{E}\left[\frac{N(u) \cap N(s)}{N(u) \cup N(s)}\right] \tag{5}$$

We additionally model the commonly-used bootstrapping strategy in the ESN and ETC settings. This strategy prioritizes high-degree users early in the attack because they have been observed to have significantly higher acceptance rates. This is useful for evading automated detection by increasing the proportion of friend requests accepted. We model this with the function

$$\text{di}(u) = \left(\frac{\mathbb{E}[\deg(u)]}{M}\right)^5$$

where M is the maximum expected degree defined previously.

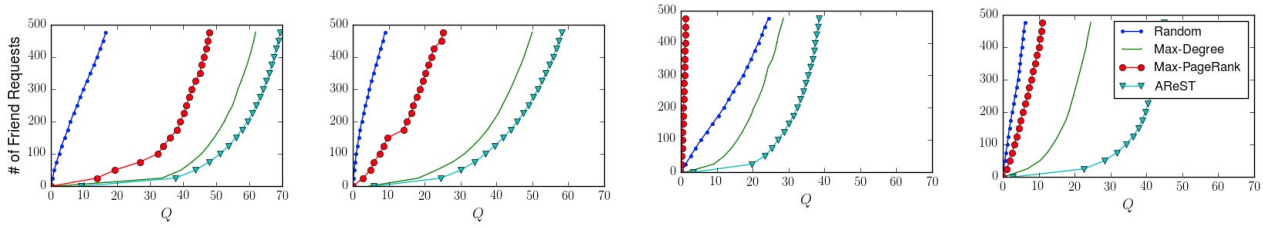
2) Choice of Target Set: In a real-world scenario, the adversary would know the target set they wished to attack. For our simulations, we model several different possible structures the target set could possess.

Single User. At one end of the spectrum, an attacker may only be interested in information about a single target user. We model this by randomly selecting a single user from the graph to be the target user.

Random Sample. The attacker may also have a set of users that are related outside the social network, but do not have any connection in the topology of the OSN. We model this by randomly sampling without replacement a set of users from the social network. In our experiments, we sample 100 users uniformly from the network.

Stochastic Breadth-First Search. The attacker's target set may also be strongly related by the topology of the OSN. We model this by randomly selecting a single user from the graph, then conducting a stochastic BFS from this user. The stochastic BFS operates in the same manner as a normal BFS with the following change: when the BFS algorithm would traverse an edge, we first flip a weighted coin to see if the edge exists in our stochastic graph and only traverse it if the edge does exist. This coin flip is *independent* of the edge reveals done during our algorithm, which models an attacker targeting a group of users on the graph but having access to only noisy data before running the Min-Friending algorithm. In our experiments, we explore 100 nodes in each SBFS run.

All Users. The final setting we consider is $T = V$, which models the scenario where the attacker is interested



(a) Enron Email (b) Slashdot (c) Facebook (d) Twitter

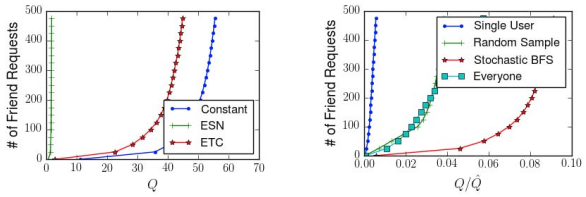
Fig. 3: Mean number of friend requests sent before reaching benefit threshold Q on each dataset using the ETC acceptance function and SBFS target setting.

in extracting as much information as possible about the graph as a whole instead of any individual part of it.

B. Algorithm Comparison

To compare the effectiveness of these algorithms, we simulate them on each combination of dataset, acceptance function, and target set structure 500 times and average the results. To conserve space, we do not plot each combination. Figure 3 shows the time taken to reach a given benefit threshold Q for each algorithm on each dataset with fixed settings of accept(u) and target set structure. From this, it is clear that our algorithm outperforms each of the alternate algorithms. The difference in performance is noticeably larger on the Facebook and Twitter networks than the Enron-Email and Slashdot networks. We note that Facebook and Twitter are more representative of what are commonly considered Online Social Networks, while Enron-Email is a communication network and Slashdot has social connections as an optional feature on a site otherwise focused on news aggregation. Figure 6 shows that of the algorithms considered, AReST gets the most benefit from friending users within the target set. The others do not make many friends within the target set, instead gaining benefit from revealing edges on the network.

From figure 4, we can see that the acceptance function and target structure have significant impact on the time to reach Q. When $T = V$, the attacker reaches Q almost instantly, while for $T = \{u\}$, it takes many friend requests. Some of this is doubtless due to the difference in size between the target sets. However, for the Random Sample and SBFS settings the sizes are the same. When a disconnected group of users is targeted, AReST does not perform as well as when the targets are closely connected. Figure 5 shows that AReST prioritizes friend requests that are likely to be accepted. The drop in acceptance rate shown in the ESN case is a result of exhausting the supply of high-degree users. Under ESN, each mutual friend is only a small marginal gain in acceptance rate, which causes the attacker to become trapped in a state where there are no highly likely friend requests to send. The impact of the acceptance function shows the importance of accurate user modeling in the study of social networks. Future work should incorporate empirical models of user behavior to ensure that any simulated results reflect real-world performance.



(a) Acceptance Function (b) Target Structure

Fig. 4: Performance of AReST on Twitter as we vary acceptance function and target structure. $\hat{Q}$ is the maximum expected benefit for the target setting.

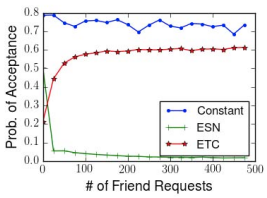


Fig. 5: Mean acceptance probability of the node targeted at each step of the AReST algorithm.

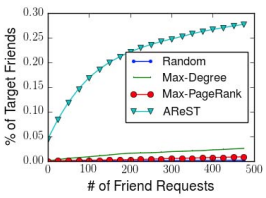


Fig. 6: Fraction of target group friended under the ETC acceptance and Stochastic BFS targeting.

C. Resistance to Reconnaissance

We can get a qualitative sense of how resistance various networks are to attack from figure 3. Facebook is the least vulnerable, because at every point the minimum time taken to reach a given benefit threshold Q is greater than any of the other networks in this experiment. The Enron-Email network is the most vulnerable for the opposite reason: it requires the least time to reach any benefit threshold. We quantify this with an intuitive metric termed *Reconnaissance Resistance Score*: mean number of friend requests to get one unit of benefit. Vulnerable networks require fewer requests per point of benefit than resistant networks. This metric depends both on the benefit function used and the algorithm that is used to crawl the network. We addressed the former problem by keeping the benefit function consistent between experiments. For the latter, we use the best performing algorithm (AReST) to give a worst-case vulnerability assessment. Table II details the resistance levels of each network against each combination

of acceptance function and target set.

From this data, we can see that both the acceptance function and target structure play a significant role in determining the resistance of a network. Across all networks, reconnaissance on the network as a whole is the easiest, while reconnaissance of a single user is most difficult. We can see from the results of the random sample (RS) targets and the SBFS targets that structure does play a role in difficulty: under the constant acceptance function RS is typically easier than SBFS. However, when the acceptance function depends on local topology, SBFS becomes easier by a large margin in most cases. From Table III, we can see that the topology of the network does impact the difficulty of reconnaissance attacks. Note that network size does not well-explain either the mean or standard deviation, as the Facebook network (the smallest, see Table I) has a mean resistance score five times as large as the Enron Email network (the second smallest). We leave the exploration of what topological features cause this to future work.

Enron Email	Constant	ESN	ETC
Single User	32.34	156.01	29.67
Random Sample	7.31	72.26	10.81
Stochastic BFS	9.87	38.03	9.69
All Users	0.09	0.45	0.09
Slashdot	Constant	ESN	ETC
Single User	35.76	549.08	39.88
Random Sample	7.94	218.883	16.25
Stochastic BFS	10.23	78.95	10.20
All Users	0.06	0.87	0.06
Facebook	Constant	ESN	ETC
Single User	36.72	1159.22	54.24
Random Sample	6.87	455.327	13.69
Stochastic BFS	11.97	219.52	14.60
All Users	0.40	13.15	0.68
Twitter	Constant	ESN	ETC
Single User	110.62	8941.19	227.06
Random Sample	8.69	2454.52	25.56
Stochastic BFS	12.554	697.05	12.33
All Users	0.05	3.75	0.1

TABLE II: *Reconnaissance Resistance Score* (mean number of friend requests required to earn one unit of benefit) for each combination of settings on each network. Larger numbers indicate higher resistance.

	Min	Mean	Max	Std. Dev.
Enron Email	0.09	30.55	156.01	44.82
Slashdot	0.06	80.68	549.08	159.81
Facebook	0.40	165.53	1159.22	340.53
Twitter	0.51	1041.08	8941.20	2585.17

TABLE III: Statistics on resistance scores over all settings for each network.

VI. Conclusion

In this paper, we present a new paradigm to quantify the OSN vulnerability in the eyes of reconnaissance attacks. Specifically, we introduce a new optimization problem, namely Min-Friending, which identifies a minimum CFS to friend in order to obtain at least Q benefit. We show that Min-Friending is inapproximable within a factor of $(1-o(1))\ln Q$ and present an adaptive algorithm which has a tight performance bound of $(1+\ln Q)$ using adaptive stochastic optimization. The key feature of our solution lies in the adaptive method, where partial network topology is revealed during each successful friend request. Extensive experiments not only confirm the performance of our algorithm, but also provide new insights into the impact of user behavior and the importance of network topology when defending against reconnaissance attacks.

References

[1] Ed Novak and Qun Li. A survey of security and privacy in online social networks. *College of William and Mary Computer Science Technical Report*, 2012.

[2] Yazan Boshmaf, Ildar Muslukhov, Konstantin Beznosov, and Matei Ripeanu. The Socialbot Network: When Bots Socialize for Fame and Money. In *Proceedings of the 27th Annual Computer Security Applications Conference*, ACSAC '11, pages 93–102. ACM, 2011.

[3] Thomas Ryan and G Mauch. Getting in bed with robin sage. In *Black Hat Conference*, 2010.

[4] Yazan Boshmaf, Konstantin Beznosov, and Matei Ripeanu. Graph-based sybil detection in social and information systems. In *Advances in Social Networks Analysis and Mining (ASONAM), 2013 IEEE/ACM International Conference on*, pages 466–473. IEEE, 2013.

[5] Inkyung Jeun, Youngsook Lee, and Dongho Won. A Practical Study on Advanced Persistent Threats. In *Computer Applications for Security, Control and System Engineering*, number 339 in Communications in Computer and Information Science, pages 144–152. Springer Berlin Heidelberg, 2012.

[6] Aviad Elyashar, Michael Fire, Dima Kagan, and Yuval Elovici. Homing socialbots: intrusion on a specific organization's employee using socialbots. In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 1358–1365. ACM, 2013.

[7] Hung T. Nguyen and Thang N. Dinh. Targeted Cyber-attacks: Unveiling Target Reconnaissance Strategy via Social Networks. In *Proceedings of the IEEE Int Conf. on Computer Com., Security and Privacy in BigData Workshop*, INFOCOM BigSecurity 2016, 2016.

[8] Abigail Paradise, Asaf Shabtai, and Rami Puzis. Hunting Organization-Targeted Socialbots. In *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015*, ASONAM '15, pages 537–540, New York, NY, USA, 2015. ACM.

[9] T. N. Dinh, Y. Xuan, M. T. Thai, P. M. Pardalos, and T. Znati. On New Approaches of Assessing Network Vulnerability: Hardness and Approximation. *IEEE/ACM Transactions on Networking*, 20(2):609–619, 2012.

[10] Md Abdul Alim, Nam P. Nguyen, Thang N. Dinh, and My T. Thai. Structural Vulnerability Analysis of Overlapping Communities in Complex Networks. In *Proceedings of the 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT) - Volume 01*, WI-IAT '14, pages 5–12. IEEE Computer Society, 2014.

[11] Sebastian Neumayer, Gil Zussman, Reuven Cohen, and Eytan Modiano. Assessing the vulnerability of the fiber infrastructure to disasters. *Networking, IEEE/ACM Transactions on*, 19(6):1610–1623, 2011.

[12] Michael Fire, Lena Tenenboim, Ofrit Lesser, Rami Puzis, Lior Rokach, and Yuval Elovici. Link prediction in social networks using computationally efficient topological features. In *Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on*, pages 73–80. IEEE, 2011.

[13] Michael Fire, Rami Puzis, and Yuval Elovici. Link prediction in highly fractional data sets. In *Handbook of computational approaches to counterterrorism*, pages 283–300. Springer, 2013.

[14] Lars Backstrom and Jure Leskovec. Supervised random walks: predicting and recommending links in social networks. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 635–644. ACM, 2011.

[15] Uriel Feige. A threshold of ln n for approximating set cover. *Journal of the ACM (JACM)*, 45(4):634–652, 1998.

[16] Daniel Golovin and Andreas Krause. Adaptive submodularity: Theory and applications in active learning and stochastic optimization. *Journal of Artificial Intelligence Research*, pages 427–486, 2011.