SybilFence: Improving Social-Graph-Based Sybil Defenses with User Negative Feedback

Qiang Cao Xiaowei Yang
Duke University
{qiangcao, xwy}@cs.duke.edu

ABSTRACT

Detecting and suspending fake accounts (Sybils) in online social networking (OSN) services protects both OSN operators and OSN users from illegal exploitation. Existing social-graph-based defense schemes effectively bound the accepted Sybils to the total number of social connections between Sybils and non-Sybil users. However, Sybils may still evade the defenses by soliciting many social connections to real users. We propose SybilFence, a system that improves over social-graph-based Sybil defenses to further thwart Sybils. SybilFence is based on the observation that even well-maintained fake accounts inevitably receive a significant number of user negative feedback, such as the rejections to their friend requests. Our key idea is to discount the social edges on users that have received negative feedback, thereby limiting the impact of Sybils' social edges. The preliminary simulation results show that our proposal is more resilient to attacks where fake accounts continuously solicit social connections over time.

1. INTRODUCTION

The popularity of online social networking (OSN) services such as Facebook and LinkedIn has attracted attacks and exploitation. In particular, OSNs are vulnerable to Sybil attacks, where attackers create many fake accounts, called *Sybils*, to send spam [10], manipulate online voting [17], crawl users' personal information [6], etc.

There has been several proposals that leverage the underlying social graph to defend against Sybils [7, 8, 17, 18, 20, 21]. The *social-graph-based Sybil defenses* are proactive approaches, as Sybils can be uncovered before they interact with real users. Those proposals have been extensively discussed in the research community due to their simplicity and reliability. The social-graph-based Sybil defenses rely on an assumption that the social edges connecting Sybils and non-Sybil users, called *attack edges*, are strictly limited. Most of them bound the undetectable Sybils, called *accepted Sybils*, to the number of attack edges [7, 20], i.e. $O(\log n)$ Sybils per attack edge.

Although social-graph-based Sybil defenses are able to provide theoretical guarantees on accepted Sybils, the upper bound of accepted Sybils still depends on the total number of attack edges. Therefore, Sybils have the incentive to solicit for social connections from real users to increase the attack edges and evade the detection. Furthermore, *well-maintained Sybils* can choose to continuously solicit social edges, at a speed similar to real users. As a result, they may be able to accumulate many attack edges from *promiscuous real users*, who are open to befriending even strangers. With current social-graph-based Sybil defenses, the fake accounts behind those well-maintained Sybils are indistinguishable from non-Sybil users, because this entire set of Sybils has adequate connectivity to non-Sybil users.

Fortunately, we observe that the attack edges from Sybils are usually accompanied by the negative feedback from *cautious real users*, who are resistant to abusive communication. A *negative feedback* can be a rejection to a friend request or a report on receiving unwanted communication. We have been conducting a study on live fake Facebook accounts in the wild (§2.1), and find a significant number of negative feedback (pending friend requests) on well-maintained fake accounts that are purchased in black market, although those accounts may manage to connect to real users.

Our understanding of this observation is that the controllers behind the fake accounts have limited knowledge about the users' security awareness. OSN users have varying levels of security awareness of the potential exploitation from fake accounts. Promiscuous users have high tolerance of abusive activities and unwanted communication, while cautious users are more resistant to fake accounts. The controller cannot distinguish these types of OSN users, and thus cannot correctly target promiscuous users. As a result, fake accounts are likely to receive negative feedback from a set of cautious users, although they may be able to interact with some promiscuous users.

To leverage this observation, we propose SybilFence, which improves over social-graph-based Sybil defenses by incorporating user negative feedback. Our key idea is to discount the social edges on users who have received negative feedback. By penalizing the social edges that are accompanied by negative feedback, we are able to mitigate the impact of Sybils' attack edges, and construct a *defense graph* with reduced weights on attack edges. In this paper, we use SybilRank, a state-of-the-art social-graph-based Sybil defense scheme [7], as a proof of concept, and adapt it to the weighted defense graph.

As shown in simulations (§4), with the defense graph, SybilFence improves over SybilRank by 10%∼20% in term of the probability of ranking non-Sybil users higher than Sybils. SybilFence is shown to be more resilient to attacks where well-maintained Sybils keep soliciting for social connections over time. We conjecture that SybilFence can also improve other Sybil detection schemes such as SybilLimit and Sybil tolerance schemes such as Bazaar [16].

2. SYBIL ACTIVITIES AND NEGATIVE FEEDBACK

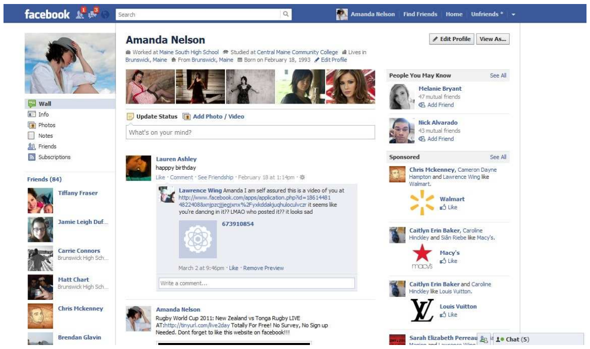
At a high level, the user negative feedback must be triggered by abusive activities in OSNs, and reflect the user distrust to the executor of the abuse. In practice, the user negative feedback includes rejections to friend requests, flags on inappropriate incoming communication such as spam, phishing, pornography and extreme violence, etc. We observe that such negative feedback has already existed in OSNs. OSNs like Facebook have collected and stored such user negative feedback, although some of them may be currently underutilized or ignored by OSN operators. Therefore, we do not introduce any change to current OSNs' use model, and our proposal is completely transparent to users.

We next take the negative feedback triggered by abusively befriending as an example and demonstrate why the negative feedback occurs and how the negative feedback associates to Sybil activities.

2.1 Negative feedback during befriending

Befriending real users is the first step for fake accounts to infiltrate an OSN after their creation. During this stage, fake accounts attempt to establish many social connections to real users. However, aggressively befriending strangers may trigger negative feedback from resistant users. For instance, a bilateral social connection requires reciprocal agreements from both users. The negative feedback during befriending can be a rejected or an ignored friend request.

Study on fake Facebook accounts in black market. To better understand the rejection to fake accounts' friend requests in real world, we have been conducting a study on fake Facebook accounts in black market. The fake accounts in black market is an example of the well-maintained live fake accounts in the wild. Those fake accounts are priced based on the ages, the number of friends, the number of pictures, etc. According to our purchase experience, a fake account with 50 ∼ 100 friends costs \$2 ∼ \$6 at different vendors. Those fake accounts look real and their friends also have rich content in profiles, walls, etc. We have purchased accounts from different vendors via Freelancer [2] and BlackHatWorld [1]. Apart from pictures, emails, security Q/A sets, we explicitly require in our purchases that the accounts should have ">50 real US friends". The vendors always ask a relatively long time period to deliver accounts after we order, e.g., one week or one month. Our work is still in progress for a large-scale study on live fake accounts in black market.



Amanda Nelson

Worked at Maine South High School · Studied at Central Maine Community College · Lives in Brunswick, Maine · From Brunswick, Maine · Born on February 18, 1993 · Edit Profile

Lauren Ashley
happpy birthday

Lawrence Wing Amanda I am self assured this is a video of you at http://www.facebook.com/apps/application.php?id=186144814822408&xnjpzcjjjegjxnx%2Fyxkddakjuqhuloculvczr it seems like you're dancing in it?? LMAO who posted it?? it looks sad

Amanda Nelson
Rugby World Cup 2011: New Zealand vs Tonga Rugby LIVE AT:http://tinyurl.com/live2day Totally For Free! No Survey, No Sign up Needed. Dont forget to like this website on facebook!!!

Chris Mckenney, Cameron Dayne Hampton and Lawrence Wing like Walmart.

Caitlyn Erin Baker, Caroline Hinckley and Siân Riebe like Macy's.

Caitlyn Erin Baker and Caroline Hinckley like Louis Vuitton.

Figure 1: A sample of purchased accounts.

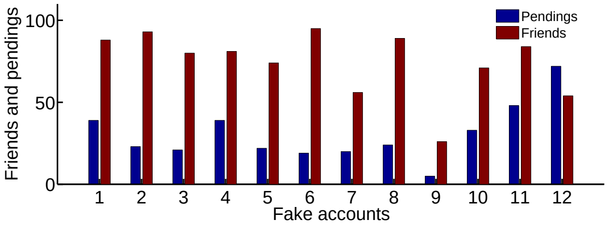


Figure 2: Friends and pending friend requests on purchased accounts.

Fake accounts look real. In this study, we use 12 fake accounts with 837 total friends. Those accounts are purchased at different vendors. All of them are at least one year old. Figure 1 is the profile of a sample account. The profile is crafted as a college student with pictures, and with posts on the wall. This account has 84 friends and has interacted with some friends, such as messages and comments.

Fake accounts receive rejections. In Facebook, a pending request implies a rejection, as Facebook does not provide the rejection option. Therefore, we examine the pending friend requests on each account. Facebook does not provide this statistic to users directly, but provides APIs to access the pending friend requests. Figure 2 shows the number of friends and pending requests on each account. As we can see, although those well-maintained accounts have many social connections to users that may be real, each of them still has a significant number of pending requests.

This result also indicates that the friend requests from those accounts have an acceptance rate >50%. It is reported that randomly flooding friend requests only yields an acceptance rate less than 30% [6, 15]. We speculate that the vendors might exploit some befriending strategies to improve the acceptance rate, such as sending requests to the friends of the users that they have already befriended (triadic closure principle) [6].

Fake accounts can befriend real users. To estimate how many real users the fake accounts have befriended, we choose 2 accounts and send a message to each of their friends. In the message, we inform the friends that the account is fake, suggest them to disconnect the social connection, and send us a message back if they established the connection by mistake. As a result, we have sent out the message to 174 friends. Within 48 hours, we received 6 messages to clarify that the connection is mistakenly established. Also, we observed a significant decrease in the friends of the test accounts. In total, 16 friends out of 174 disconnect the connections to the fake accounts. This number is a conservative estimation of the real users that those fake accounts have connected to, because some real users may have simply ignored our messages.

2.2 Discussion

Non-manipulability of negative feedback. In our proposal, a user is able to signify a negative feedback only if she received unwanted communication such as an unexpected friend request, a spam message, or a spam post on her wall. This means that a negative feedback can be generated only if the user has been directly annoyed or harmed. This is in contrast to the negative ratings in online services such as YouTube and Flickr, where users are granted to rate arbitrarily based on their preference. In our proposal, a real user will not receive negative feedback if she never sends out unwanted communication. We choose to use the abuse-triggered negative feedback because it is non-manipulable under collusion. Without the trigger of abusive activities, a group of malicious users cannot collude to render arbitrarily negative feedback to a victim.

Why not use negative feedback to directly detect Sybils? The negative feedback can be used to directly detect Sybils with machine-learning (ML) techniques. However, ML-based techniques [19] require extensive calibration efforts due to the abundance of possible legitimate and malicious behaviors in OSNs. More importantly, these approaches are based on individual user features, and the resulting alarms or alerts are only applicable to individual users. Therefore, such techniques may miss the Sybils behind the active entrance Sybils or currently silent entrance Sybils. Instead, SybilFence employs social-graph-based schemes and considers Sybils as groups. By aggregating negative feedback, SybilFence is able to leverage the aggressive behaviors of the entrance Sybils to uncover a much larger set of Sybils behind them.

3. SYSTEM DESIGN

In the previous section, we discussed what is the user negative feedback and how would user negative feedback occur. We now discuss how to incorporate user negative feedback into social-graph-based Sybil defenses. We first introduce our system model and threat model.

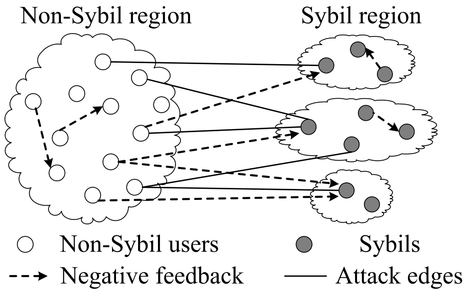


Figure 3: The social graph and the negative feedback graph in an OSN.

System model. We model an OSN with two graphs: 1) the underlying social graph as an undirected graph $G^+ = (V, E^+)$, where V is the user set in the OSN, and E^+ represents the social relationships among users; and 2) the negative feedback graph as a directed graph $G^- = (V, E^-)$, where V is the same user set as in G^+, E^- is the directed edge set that includes the negative feedback between users. Given a direction between each pair of users, we consider no more than one negative feedback edge. Figure 3 shows both the social graph and the negative feedback graph sharing the same user set in an OSN. A node v has a *social degree* of $deg^+(v)$ in G^+ and an in-degree of $deg^-(v)$ in G^-.

Threat model. Malicious users may launch Sybil attacks by creating many fake accounts. We divide the user set V into two disjoint subsets: non-Sybil users and Sybils, as shown in Figure 3. The *non-Sybil region* is the collection of the non-Sybil users, and the social edges and negative feedback links among them. Similarly, we define the *Sybil region* with respect to the Sybils. We refer to the social edges between the non-Sybil region and the Sybil region as *attack edges*. We refer to the Sybils adjacent to the attack edges as *entrance Sybils*, and the other Sybils as *latent Sybils*.

3.1 Overview

SybilFence aims to improve the social-graph-based Sybil defenses using the negative feedback. Our observation is that the Sybils' attack edges are always accompanied by negative feedback. If we discount the trustworthiness of the social edges that come with the negative feedback, we can limit the impact of the excessive attack edges on well-maintained entrance Sybils, and enable social-graph-based Sybil defenses to uncover the Sybils behind them.

SybilFence comprises of two major modules: 1) a *negative feedback combiner*, which incorporates the negative feedback graph into the social graph, and generates a *defense graph* with discounted social edges on the users that have received negative feedbacks; and 2) an *adopted social-graph-based defense scheme* that detects Sybils on the defense graph with improved accuracy.

3.2 Incorporating negative feedback

The social-graph-based Sybil defenses bound the accepted Sybil to the number of attack edges [7,20], regardless of how much negative feedback that Sybils have received. SybilFence improves over social-graph-based Sybil defenses by reducing the impact of attack edge through the collected user negative feedback. In principle, we aim to build a weighted *defense graph* based on the social graph and the negative feedback graph. We reduce the weights of social edges on the users that have received negative feedback, such that the aggregate weight on attack edges is substantially limited.

Defense graph. Our key idea is to discount a user's social relationships with the negative feedback that the user has received. By cancelling out the social edges that come along with negative feedback from other users, we are able to mitigate the impact of the entrance Sybils' attack edges. We define the *net social degree* of a node v as $net(v) = deg^{+}(v) - \alpha \times deg^{-}(v)$ (we require $net(v) \geq 0$), where the offset factor α is a positive parameter. A large α indicates a substantial penalty of an incoming negative feedback edge. Thus, a node whose social edges accompanied by negative feedback has a net social degree $net(v)$ smaller than its social degree $deg^{+}(v)$.

We define the weight of a node v as its net social degree divided by its social degree: $w(v) = \frac{net(v)}{deg^{+}(v)}$. The node weight is essentially the discount rate of a node. With this definition, a real user that never receives negative feedback has a weight of value 1, while a user with received negative feedback is assigned a discounted weight due to the discounted social degree. A node weight can be translated to the extent to which the node can be trusted.

We derive the weight of a social edge based on the weights of its adjacent nodes: $w(u, v) = \min(w(u), w(v))$. The weight of an edge is determined by the lowest weight of its adjacent nodes. This is because any of the adjacent nodes that has triggered negative feedback should discount the edge quality. Therefore, an edge weight is always no more than 1, and low weighted social edges are always adjacent to incoming negative feedback links.

Therefore, we can build a weighted and undirected *defense graph* $G = (V, E^{+}, w)$. We weight each edge (u, v) with $w(u, v)$ that discounts the quality of social edges. As compared to the social graph, where every social edge is treated equally, our defense graph enforces strictly limited aggregate weight on attack edges using negative feedback.

3.3 Detecting Sybils

In our initial design, we take SybilRank [7] as a proof of concept, and adapt it to the weighted defense graph. SybilRank comprises of three steps: trust propagation, trust normalization, and ranking. We adapt SybilRank to use the defense graph as below.

In the first stage, SybilRank propagates trust from the trust seeds via $O(\log |V|)$ power iterations. In each iteration, the distributed trust on each edge is proportional to the edge weight. $T^{(i)}(v)$ is the amount of trust on node v in the i^{th} iteration. We assume that the non-Sybil region part of the defense graph is well connected after social edges discounting. We empirically validate it with simulations (§4), and leave a formal study in future work. In the second stage, SybilRank normalizes the total trust of every node with its social degree. Since a node's social degree is always no less than the aggregate weight of its adjacent edges due to the discounted edge weights, this normalization further penalizes Sybils that are likely to have substantially discounted social edges. The last stage outputs a ranked list according to the degree-normalized trust with non-Sybil users on top.

Algorithm $Adapted\_SybilRank(G(V, E^{+}, W), seedSet\ S)$

```
    Stage I: $O(\log |V|)$-step trust propagation
    In initialization, seed trust evenly in trust seeds;
    In a step $i$ ($0 < i \leq h$, $h = O(log|V|)$), node $u$
    updates its trust as below:
    $T^{(i)}(u) = \sum_{(u,v) \in E^{+}} T^{(i-1)}(v) \frac{w(u,v)}{\sum_{(k,v) \in E^{+}} w(k,v)}$
    Stage II: Normalize node trust by social degree
                $\hat{T}_u = \frac{T_u^{(h)}}{deg^{+}(u)}$
    Stage III: Rank users based on their
    degree-normalized trust $\hat{T}$
    Return ranked list $L$
```

4. SIMULATION STUDY

To gain a better understanding on how our initial design improves the detection accuracy, we evaluate SybilFence in comparison to the original SybilRank. We simulate user friend requests on social graphs, and use the request rejections as negative feedback.

4.1 Simulation setup

We simulate Sybil attacks in four social graphs (Table 1). The Facebook graph is sampled via the "forest fire" sampling method [14]. The synthetic graph is generated based on the scale-free model [4]. We connect a Sybil region that consists of 5,000 Sybils to each social graph. We simulate the social connections among Sybils by establishing social edges from each Sybils to another 5 random Sybils upon its arrival.

Social Network	**Nodes**	**Edges**	**Clustering Coefficient**	**Diameter**
Facebook	$10,000$	$40,013$	0.2332	17
ca-AstroPh [3]	$18,772$	$198,080$	0.3158	14
ca-HepTh [3]	$9,877$	$25,985$	0.2734	18
Synthetic	$10,000$	$39,399$	0.0018	7

Table 1: Social graphs used in our simulation.

Similar to [7] and [18], we use the metric *the area under the Receiver Operating Characteristic (ROC) curve* [12] to compare the quality of the ranking that social-graph-based schemes use to uncover Sybils. The area under the ROC curve measures the probability that a Sybil user is ranked lower than a random non-Sybil user. It ranges from 0 to 1.

4.2 Simulating negative feedback

Users can send friend requests to others. A request acceptance yields a social edge, while a rejection produces a negative feedback edge.

Rejections to Sybil users. We simulate the process that the entrance Sybils solicit social edges from non-Sybil users. In the Sybil region, we designate 200 nodes as entrance Sybils, and the rest 4800 nodes as latent Sybils. The entrance Sybils represent well-maintained fake accounts, which continuously send friend requests and have lower rejection rates than latent Sybils due to the better maintenance. By default we set the rejection rate of entrance Sybils by non-Sybil users to 60%, and the rejection rate of latent Sybils to 98%.

Rejections to non-Sybil users. We simulate rejections to non-Sybil users based on the social graph. In particular, given a rejection rate to non-Sybil users and the number of friends that a non-Sybil user has in the social graph, we can infer the number of rejections on this user. we then add this number of rejections to the non-Sybil user by randomly selecting non-friend users and simulating a rejection from each of them. We set the rejection rate of non-Sybil users to 1%. We study how SybilFence's performance varies with the change of the rejection rates in §4.3.

4.3 Simulation results

We now present the simulation results in the Facebook graph. The results on other graphs are similar (see Appendix).

Impact of the negative-feedback offset factor. The offset factor α (§3.2) is a penalty factor to the nodes that have received rejections, including both Sybils and non-Sybil users. To investigate its impact, we vary the value of the penalty factor from 0 to 4. Since we set the rejection rate of entrance Sybils' requests to 60%, an offset factor of value $\frac{2}{3}$ can leverage the rejections to cancel out the attack edges on entrance Sybils. However, the entrance Sybils also have social edges from Sybils. Therefore, a larger offset factor can yield further improvement. Figure 4 shows that the improvement keep increases until the offset factor reaches a sufficient large value, i.e., 3.0. With this value, SybilFence is able to cancel out most of the entrance Sybils' social edges from both non-Sybil users and Sybils.

Resilience to Sybils' flooding requests. The fake accounts can solicit social edges by flooding friend requests. As a result, the attack edges keep increasing. We study the SybilFence's resilience to request flooding. We set the offset factor to 1 and vary the number of requests that each entrance Sybil sends from 4 to 36. Each latent Sybil is set to send 2 requests to random non-Sybil users. Consequently, the attack edges increase from ∼500 to ∼3000. Figure 5 shows that SybilFence has only small performance degradation. In contrast, SybilRank's performance decreases sharply, because its security guarantee only relies on the number of attack edges.

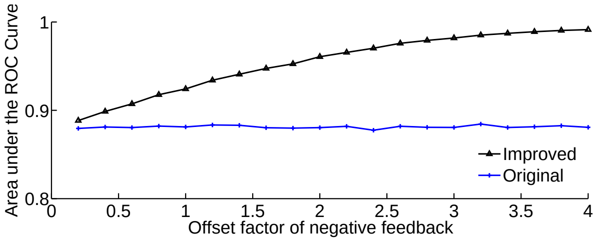


Figure 4: Impact of the offset factor of negative feedback.

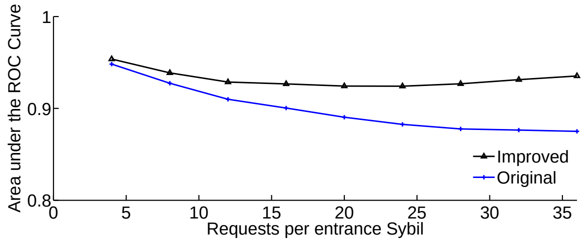


Figure 5: Resilience to the number of requests from entrance Sybils.

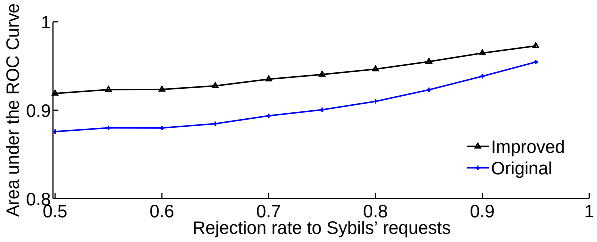


Figure 6: Detection accuracy as a function of the rejection rate to the Sybils' requests.

Impact of the rejection rate to Sybils' requests. At a high level, both SybilFence and social-graph-based schemes rely on non-Sybil users to defense against Sybils. We investigate the impact of the rejection rate to Sybils' requests. In this simulation, each entrance Sybil sends 25 requests to random non-Sybil users. We vary the rejection rate to these requests from 0.5 to 0.95. The number of attack edges decreases accordingly from ∼2700 to ∼450. An increased rejection rate to Sybils' requests improves both SybilFence and SybilRank as shown in Figure 6, because this further limits the attack edges and signifies more negative feedback. SybilFence achieves higher accuracy due to its advantage from the consideration of negative feedback.

Rejections among non-Sybil users. SybilFence is based

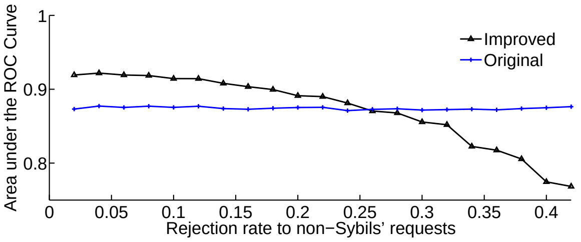


Figure 7: Detection accuracy as a function of the rejection rate to non-Sybil users' requests.

on an assumption that non-Sybil users are less likely to receive negative back from others than Sybils. We study SybilFence's performance with a varying rejection rate to non-Sybils' requests. We simulate the rejections among non-Sybil users as decribed in §4.2. The rejection rate increases from 0.05 to 0.45. The offset factor is set to 1 and the rejection rate to entrance Sybils' requests is set to 0.4. As shown in figure 7, the improvement that SybilFence makes over SybilRank decreases as the rejection rate to non-Sybil users' requests increases. This is because more rejections to non-Sybil bring more penalty to non-Sybil users, and thus increase the chance that a non-Sybil user ranks low in SybilFence. SybilFence performs even worse than SybilRank when the rejection rate to non-Sybil users reaches 0.25. The reason is that beyond this rejection rate, the non-Sybil users are more likely to get rejections than Sybils. This threshold is smaller than the entrance Sybils' rejection rate 0.4, because the entrance Sybils have 0 rejection rate to establish social edges among themselves. This results indicate that SybilFence cannot improves SybilRank if non-Sybil users are more likely to be rejected. We suspect that this does not happen in real world, because real users always send friend requests to acquaintances. We leave a further study to our future work.

5. RELATED WORK

This work is mainly related to social-graph-based Sybil defenses [7, 8, 17, 18, 20]. These proposals rely on social graph properties to distinguish Sybils from non-Sybil users, i.e., the attack edges are strictly limited. Existing schemes bound the accepted Sybils to the number of attack edges. Thus, fake accounts benefit from soliciting social connections. SybilFence improves over existing Sybil defenses by leveraging negative feedback from users. It can be used to further uncover the fake accounts that have obtained social edges to real users but inevitably received negative feedback from resistant users.

There have been proposals to propagate distrust in social graphs [5, 9, 11, 13, 22]. However, those approaches are proposed as general techniques for social network analysis, but not targeting Sybil defense. Furthermore, the distrust in those proposals is not securely defined, which can include arbitrarily negative information and is not resilient to user collusion.

6. CONCLUSION AND FUTURE WORK

The detection of fake accounts in OSNs has been increasingly urgent as both OSN operators and users have been suffering from illegal exploitation. We observe that even well-maintained fake accounts inevitably receive negative feedback from others, as the controllers only have limited knowledge on users' security awareness. Thus, user negative feedback can be used to strengthen existing Sybil defenses. We propose SybilFence, which incorporates user negative feedback into social-graph-based Sybil defenses. Fake accounts can evade SybilFence only if they can connect to real users, and meanwhile receive little negative feedback from others. Therefore, SybilFence advances the Sybil defenses by raising the cost for Sybils to evade detection.

In future work, we plan to continue our study on the live fake accounts in black market to further quantify the negative feedback they receive. With this study, we plan to complete and extend the our preliminary SybilFence design. We can then implement the complete SybilFence system and probably deploy it in real OSNs.

7. REFERENCES

[1] BlackHatWorld. `http://www.blackhatworld.com/`.

[2] Freelancer. `http://www.freelancer.com/`.

[3] Stanford large network dataset collection. `http://snap.stanford.edu/data/index.html`.

[4] A.-L. Bárabási and R. Albert. Emergence of Scaling in Random Networks. *Science*, 286:509–512, 1999.

[5] C. Borgs, J. Chayes, A. T. Kalai, A. Malekian, and M. Tennenholtz. A Novel Approach to Propagating Distrust. In *WINE*, 2010.

[6] Y. Boshmaf, I. Muslukhov, K. Beznosov, and M. Ripeanu. The Socialbot Network: When Bots Socialize for Fame and Money. In *ACSAC*, 2011.

[7] Q. Cao, M. Sirivianos, X. Yang, and T. Pregueiro. Aiding the Detection of Fake Accounts in Large Scale Social Online Services. In *NSDI*, 2012.

[8] G. Danezis and P. Mittal. SybilInfer: Detecting Sybil Nodes using Social Networks. In *NDSS*, 2009.

[9] C. de Kerchove and P. van Dooren. The PageTrust Algorithm: How to Rank Web Pages When Negative Links are Allowed? In *SDM*. SIAM, 2008.

[10] H. Gao, J. Hu, C. Wilson, Z. Li, Y. Chen, and B. Y. Zhao. Detecting and Characterizing Social Spam Campaigns. In *IMC*, 2010.

[11] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins. Propagation of Trust and Distrust. In *WWW*, 2004.

[12] J. A. Hanley and B. J. McNeil. The Meaning and Use of the Area under a Receiver Operating Characteristic (ROC) Curve. *Radiology*, 143, 1982.

[13] J. Kunegis, A. Lommatzsch, and C. Bauckhage. The Slashdot Zoo: Mining a Social Network with Negative Edges. WWW, 2009.

[14] J. Leskovec and C. Faloutsos. Sampling from Large Graphs. In *SIGKDD*, 2006.

[15] S. Panagiotopoulos, Q. Cao, M. Sirivianos, A. Stavrou, C. Liang, and X. Yang. Quantifying the Cost of Sybil Attacks in Online Social Networks. `http://tinyurl.com/quantifying-sybil-cost`, 2011.

[16] A. Post, V. Shah, and A. Mislove. Bazaar: Strengthening user reputations in online marketplaces. In *NSDI*, 2011.

[17] D. N. Tran, B. Min, J. Li, and L. Subramanian. Sybil-Resilient Online Content Rating. In *NSDI*, 2009.

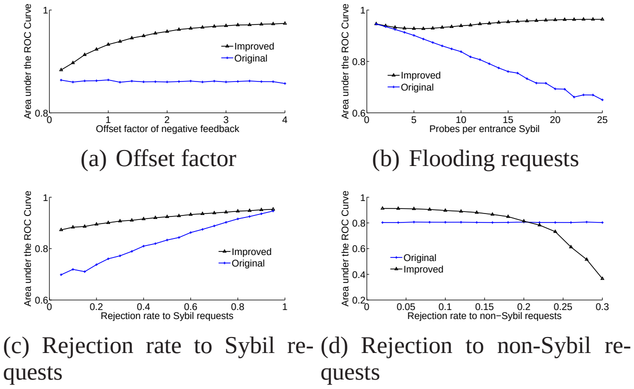
[18] B. Viswanath, A. Post, K. P. Gummadi, and A. Mislove. An Analysis of Social Network-based Sybil Defenses. In *ACM SIGCOMM*, 2010.

[19] Z. Yang, C. Wilson, X. Wang, T. Gao, B. Y. Zhao, and Y. Dai. Uncovering Social Network Sybils in the Wild. In *IMC*, 2011.

[20] H. Yu, P. Gibbons, M. Kaminsky, and F. Xiao. SybilLimit: A Near-Optimal Social Network Defense Against Sybil Attacks. In *IEEE S&P*, 2008.

[21] H. Yu, M. Kaminsky, P. B. Gibbons, and A. Flaxman. SybilGuard: Defending Against Sybil Attacks via Social Networks. In *SIGCOMM*, 2006.

[22] C.-N. Ziegler and G. Lausen. Propagation Models for Trust and Distrust in Social Networks. *Information Systems Frontiers*, 7, 2005.



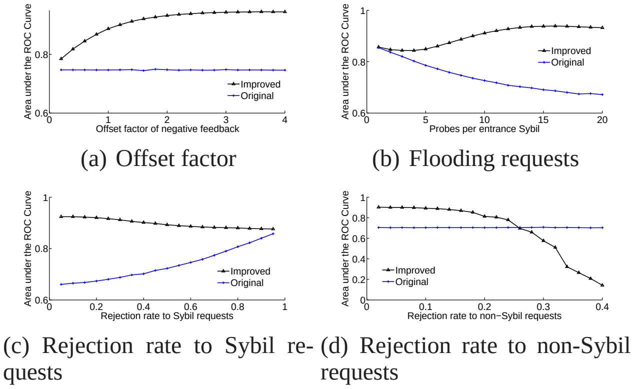
(a) Offset factor

(b) Flooding requests

(c) Rejection rate to Sybil requests

(d) Rejection to non-Sybil requests

Figure 8: Simulation results in ca-AstroPh.



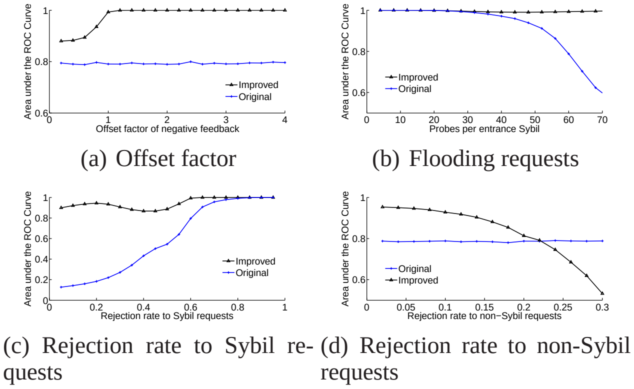
(a) Offset factor

(b) Flooding requests

(c) Rejection rate to Sybil requests

(d) Rejection rate to non-Sybil requests

Figure 9: Simulation results in ca-HepTh.



(a) Offset factor

(b) Flooding requests

(c) Rejection rate to Sybil requests

(d) Rejection rate to non-Sybil requests

Figure 10: Simulation results in the synthetic graph.

APPENDIX

Figure 8, Figure 9, and Figure 10 show the simulation results in the graphs ca-AstroPh and ca-HepTh, and the synthetic graph (Table 1). In these graphs, SybilFence achieves similar improvement over SybilRank as described in §4.3.