SybilRadar: A Graph-Structure Based Framework for Sybil Detection in On-line Social Networks

Dieudonné Mulamba, Indrajit Ray, Indrakshi Ray

SybilRadar: A Graph-Structure Based Framework for Sybil Detection in On-line Social Networks

Dieudonné Mulamba, Indrajit Ray, and Indrakshi Ray

Dept. of Computer Science, Colorado State University, Fort Collins, CO 80523
{mulamba, indrajit, iray}@cs.colostate.edu

Abstract. Online Social Networks (OSN) are increasingly becoming victims of Sybil attacks. These attacks involve creation of multiple colluding fake accounts (called Sybils) with the goal of compromising the trust underpinnings of the OSN, in turn, leading to security and the privacy violations. Existing mechanisms to detect Sybils are based either on analyzing user attributes and activities, which are often incomplete or inaccurate or raise privacy concerns, or on analyzing the topological structures of the OSN. Two major assumptions that the latter category of works make, namely, that the OSN can be partitioned into a Sybil and a non-Sybil region and that the so-called "attack edges" between Sybil nodes and non-Sybil nodes are only a handful, often do not hold in real life scenarios. Consequently, when attackers engineer Sybils to behave like real user accounts, these mechanisms perform poorly. In this work, we propose SybilRadar, a robust Sybil detection framework based on graph-based structural properties of an OSN that does not rely on the traditional non-realistic assumptions that similar structure-based frameworks make. We run SybilRadar on both synthetic as well as real-world OSN data. Our results demonstrate that SybilRadar has very high detection rate even when the network is not fast mixing and the so-called "attack edges" between Sybils and non-Sybils are in the tens of thousands.

Keywords: security of on-line social networks, sybil attacks and detection, graph structures, graph metrics

1 Introduction

The success of Online Social Networks (OSNs) [**?**] such as Facebook, Twitter, LinkedIN, and Google+, have made them a lucrative target for attackers. Owing to their open nature, they are specifically vulnerable to a new form of threat call Sybils. In a Sybil attack, an adversary creates a large number of fake identities or forges a large number of existing identities and uses those to target the trust underpinnings of the OSN [**?**]. Various types of malicious attacks can be launched this way such as, social spamming [**?**], malware distribution [**?**], and private data collection [**?**]. Therefore, it is important to provide OSN administrators a tool for detecting Sybil accounts automatically, speedily and accurately.

Although much effort has been devoted to design such a tool, existing Sybil defense approaches are efficient against a naïve attacker but can be evaded by sophisticated ones. An attacker can evade a "content-based approach" in which different features of OSN user-level attributes and activities are analyzed to discriminate them from fake account activities [?], by creating fake accounts whose features are similar to those of real accounts. On the other hand, several researches [?], [?], [?] have shown that "structure-based approaches", which model the OSN as graph with nodes and edges respectively representing user accounts and social relationships, can be evaded by an attacker who succeeds in creating a large number of edges between the fake accounts and the benign ones. This happens specially in weak-trust OSNs. Given that extracting and selecting appropriate features from users attributes and activities for content-based Sybil detection is challenging, prone to inaccuracies, and often raises privacy issues, we propose SybilRadar, a Sybil detection mechanism that is based on graph-based structural properties of OSNs. SybilRadar is able to protect OSNs with weak trust relationships against Sybil attacks. We exprimentally evaluate the accuracy of SybilRadar in detecting Sybils using real world OSN data. Our results show that SybilRadar has much better detection accuracy than the closest competitor.

The rest of the paper is organized as follows. Section 2 discusses major works in OSN Sybil defense. In Section 3 we present the system model for SybilRadar. We discuss why assumptions in existing structure-based detection mechanisms are invalid under real world settings. We end the section with a discussion on our attack model. The main design of SybilRadar is presented in Section 4. We discuss the major intuitions in our design and the different graph metrics that we used. Section 5 presents the experimental setup and evaluation of SybilRadar including comparison with SybilRank, which is the closest in design to SybilRadar. We conclude in Section 6 with a discussion of our results and pointers to future work.

2 Related Works

Several studies have shown that OSNs are very vulnerable to Sybil attacks. Facebook [?], Twitter [?], [?], and Renren [?] have each experienced significant amount of spams whose origins were Sybil attacks. Several researchers have investigated approaches to defend against Sybil attacks on Online Social Network following studies that have been conducted to assess the severity of these attacks. Two bodies of works have been proposed in order to mitigate Sybils. The first body of works that we call content-based approaches leverages user behaviors and employs machine-learning techniques to learn and classify these behaviors. OSN nodes deviating significantly from these nodes are called Sybils. The second body of works that we call structure-based approaches leverages graph-theoretic proprieties of the social network. Nodes that exhibit significantly different properties than others are identified as Sybils.

Content-based approaches aim to find Sybil accounts by using a classifier trained using machine-learning techniques. The most recent user activities are analyzed to extract some unique features that will serve as inputs on which a classifier is built. Machine-learning techniques such as clustering, support vector machines, and Bayesian networks are used to build the classifier. Some of these approaches are used for spam detection such as blacklisting, whitelisting, and URL filtering [?], [?], [?]. While many of these approaches have very high detection rates, the problem with these approaches is that they are only as good as the data that are used to train the classifiers. We believe that identifying proper features from user attributes and activities is challenging because these attributes often contain incomplete, inaccurate and sometimes purposefully misleading information. Additionally, a sophisticated attacker can create fake accounts presenting features similar to the ones one of real accounts, thus evading detection. We also believe creating such user profiles can lead to privacy breaches and are not supportive of such techniques. Consequently, We do not consider content-based approaches in our work any further.

Structure-based approaches model an OSN as a graph with user accounts and social relationships respectively represented by nodes and links. These approaches determine some graph-theoretic characteristics of nodes which are then used to discriminate Sybils from the real ones. Existing structure-approaches are based on two assumptions. The first is that the social graph will be partitioned into two distinct regions, one region with the Sybil nodes and the other one with benign nodes. The second assumption is that there will be only a small number of attack edges between the two regions, as a consequence of the strong trust relationship in the social graph. Several mechanisms use these approaches to detect Sybil communities, which are tight-knit communities that have a small quotient-cut from the honest region of the graph [**?**], [**?**], [**?**].

SybilRank [**?**] is one of the most well-known techniques. It uses graph-theoretic properties of the OSN social graph to compute the likelihood of users to be Sybils in order to perform the ranking. The detection starts with the administrator determining some known real users as initial seed node. A short random walk is run with the known seeds. At the end of the random walk, all nodes are given trust values which are the landing probabilities for the random walk. SybilRank then ranks all the nodes based on their trust value. Nodes having higher trust value will be at the top, while the nodes with lower trust values will be lowly ranked. SybilRank performs almost linearly in the size of the social graph.

However, SybilRank is based on certain assumptions that several researches [**?**], [**?**] have proven not to be true in real life. In addition to these researches, Yang et al. show that Sybils on Renren blend into the social graph rather than forming tight communities [**?**]. Mohaisen et al. show that many social graphs are not fast-mixing, which is a necessary precondition for the structure-based Sybil detector of SybilRank to be effective [**?**]. SybilRadar, on the other hand, does not make any of these assumptions.

Integro [**?**] is an approach that extends SybilRank. It is developed without the two assumptions on which SybilRank is based on. Integro is a hybrid approach. It mixes content-based approach with a structure-based approach in order to detect Sybils. Integro first determines unique features for users which are used to build a feature-vector. The feature-vectors are used to train a classifier that predicts potential victims of Sybil attacks. After finding the potential victims, the edges in the social graph are given weights based on whether they are adjacent to the potential victims or not. The ranking is then performed by a modified random walk. Integro achieved a 95% precision in detecting Sybils. Our approach produces similar detection accuracy without using any content-based techniques.

SybilFrame [?] relaxes the assumptions that the social network can be partitioned into two distinct regions – Sybil and non-Sybil – and that there exists only a small number of attack edges between the two regions. SybilFrame is also a hybrid approach that leverages the attributes of an individual node along with a measure of correlation between connected nodes in order to classify nodes among benign and Sybils. SybilFrame operates in two steps. In the first step the initial network data are fed into the framework from which node unique features are extracted in order to compute node prior information. In Step 2, the node prior information are provided to the posterior inference layer in order to compute the correlation between nodes. This nodes correlation is computed using Markov Random Field, and along with the Loopy Belief Propagation method, it provides the posterior information of nodes which is used to perform the ranking of nodes.

3 Preliminaries

We begin by presenting the system model for our work. We then introduce the notion of strong and weak trust relationships in OSNs. We explain why SybilRank does not perform well in a real-world OSN with weak trust. We end this section with a discussion of our attack model.

System Model: Trust relationship between two OSN users allows one to assess the information based upon which further information sharing can be performed or a service can be expected [?], and is the underpinning on which OSNs are built. Consider the social network topology as defined by a graph G = (V,E) comprising a set of vertices V, denoting users on the social network and E a set of edges, representing trust relationships (or friendship) between users. We assume trust relationships are mutual (bi-directional) and represent it with undirected edges between the users in the graph G. Two kind of nodes are considered here – an *honest* node and a Sybil node. A honest node that has accepted, or is susceptible to accepting a friend request from a Sybil node is considered to be a *victim* node. The subgraph of G containing all the honest nodes is considered to be the honest region of the OSN, while the Sybil region is the subgraph of G containing all sybil nodes.

We consider three kind of edges. *Attack* edges are those connecting victim nodes in an honest region and Sybil niodes. *Sybil* edges connet Sybil nodes to each other. Finally we have *honest* edges that connect honest nodes with each other.

OSNs with weak trust: In early studies [?], [?], OSNs were assumed to have strong trust relationships. OSNs with strong trust are those that possess the property of *fast-mixing*. For Sybil detection purposes, this boils down to a social network with a *small cut*, which is a set of edges whose removal will disconnect the graph into two distinct regions – the honest region and the Sybil region [?]. In other words, in a social network with strong trust we can distinguish the two distinct regions and there is a very limited number of attack edges between the regions (in the tens). OSN with weak trust, on the other hand, is a network that does not display the fast-mixing property. Indeed, it was demonstrated [?] that not many social networks are fast-mixing. In this work, we assume an OSN with weak trust, which is in contrast to SybilRank.

Attack Model: We assume that an attacker can create an unlimited number of Sybil nodes constituting a subgraph (the Sybil region) whose topology is beyond the control of the OSN provider. Attackers can create as many number of attack edges as they want, but they do not have control on how many of those attacks edges will be *successful* in establishing victims. Our Sybil defense mechanism is built around the assumption that we know at least one honest node. This assumption is reasonable since such information can be provided by the administrator of the OSN after a carefully designed process for that purpose. Same assumption is made by other works as well. In addition, we assume that the attacker does not have complete knowledge of the entire OSN topology, since this will require him to crawl the entire network. However, the attacker can acquire the knowledge about a subgraph of the OSN

4 SybilRadar System Design

SybilRadar operates in three steps. The process starts with the network dataset (set of nodes and edges) being fed to the SybilRadar framework. The first step involves the computation of similarity values between a given pair of nodes. The chosen similarity metric is the Adamic-Adar metric [**?**], which is based on the notion of common friends between any given pair of nodes. The intuition for choosing this metric is that honest nodes will have more friends in common that Sybil nodes. In the second step, the result from the first step is refined using another similarity metric which is the Within-Inter-Community metric (WIC) [**?**]. This metric leverages the underlying community structure of the given social graph. The Louvain method [**?**] is used to find the social graph community information that is fed to the WIC similarity metric computation. This step produces the prior information which is the similarity values of any given pair of nodes driven by the community they belong to. We end this step with a tuning of the nodes similarity values for those nodes with a similarity value greater than 1. We assign the resulting similarity values to the social graph edges as their weights. In the third step, we run a Modified Short Random Walk on the weighted social graph. This step produces trust values, which are the node's landing probabilities of the random walk. These values are assigned to each node as the posterior information in order to perform the ranking of nodes.

4.1 Predicting Attack Edges

Similarity metrics have been extensively used in the field of link prediction in networks. The link prediction problem consists of predicting possible future links based on observing existing links in a given network. Sybils try to maliciously create trust relationships with honest nodes by creating attack edges. Our algorithm tries to predict these bad links. The prediction of future possible links can be based on observing unique and recent features of nodes present in the network, or can be based on structural properties of nodes present in the network. In the first case, feature similarity metrics are used, while structural similarity metrics are used in the latter case. Interested readers are referred to [**?**], [**?**] for link prediction works using feature similarity metrics, and references [**?**], [**?**] for link prediction works based on structural similarity metrics. In OSNs node attributes are not always available. For example, users may not complete their profiles or provide inaccurate or misleading information to protect their sensitive information. Moreover, trying to learn user behavior, where complete information is available, may raise privacy concerns. This leads us to consider structural similarity metrics, which are based solely on the structure of the social graph induced by trust relationships between users [**?**].

We adopt the Adamic-Adar metric [**?**] to compute an initial similarity value of pairs of nodes. For a given OSN graph $G = (V, E)$, let x and y be two nodes and $\Gamma(x)$ and $\Gamma(y)$ be the sets of neighbors of x and y. The Adamic-Adar (or simply Adamic) similarity measure is given by

$$S_{x,y}^{AA} = \sum_{w \in \Gamma(x) \cap \Gamma(y)} \frac{1}{log \mid \Gamma(w) \mid} \tag{1}$$

Given the initial social graph, running the $Adamic$ similarity metric on each pair of nodes results in a weighted social graph with the weight on a link being the similarity value of nodes adjacent to that link. For a given social graph $G = (V, E)$ and for each edge $(u_1, u_2) \in E$, the similarity value $Adamic(u_1, u_2)$ becomes its weight $w(u_1, u_2)$. After computing the Adamic similarity metric we make the following observations :

1. We have three sets of edges: edges with weight $w(u_1, u_2) = 0$, those with weight $w(u_1, u_2) \in [0, 1]$, and the edges with weight $w(u_1, u_2) > 1$.
2. For the attack edges, at least 95% of them have their weight $w(u_1, u_2) = 0$, and less than 5% have their weight $w(u_1, u_2) \in [0, 1]$, while about zero to an infinitely small number of them have their weight $w(u_1, u_2) > 1$.
3. The situation for honest edges is quite different. At least 90% of them have their weight $w(u_1, u_2) > 1$, and about less than 5% have their weight $w(u_1, u_2) \in [0, 1]$, whereas those with weights $w(u_1, u_2) = 0$ are also less than 5%.

We were able to make these observation because the social graph used for simulation purpose is derived from a synthetic network whose attack edges, Sybil edges and honest edges are known beforehand. We were able to predict about 90% of existing attack edges. We made similar observation later with our real data. We observe that predicting attack edges can be very helpful. since it can reveal nodes that have potentially been victims of Sybil attacks. This can be a valuable information for a system administrator. Note, however, that not all edges that have their $w(u_1, u_2) = 0$ are all attack edges. In other words, some honest edges, as well as some Sybil edges, have their weight equal to 0. This is due to the fact that not all pairs of honest nodes or Sybil nodes have common friends, which is the criteria used in computing the similarity value using the Adamic metric.

4.2 Further Refinement of Attack Edge Detection

We next observed that there was an extreme case where our current Sybil detection algorithm completely looses its accuracy. This situation arises when the number of attack edges far exceeds the number of honest nodes.

This situation is not desirable because, at this level, any attacker that can succeed to create a huge number of attack edges compared to the number of benign accounts and get a high degree of certainty of having a significant number of his Sybil accounts evading the Sybil detection mechanism. We observe that among the attack edges that were not detected a significant number have their weights $w(u_1, u_2) \in [0, 1]$. These edges are mixed with a significant portion of other non attack edges which also have their weight $w(u_1, u_2) \in [0, 1]$. We want to filter out as many attack edges as we can in order to increase the number of detected attack edges. For this purpose, we leverage properties of communities (or clusters) in networks.

Community detection OSNs typically display clustering characteristics. The idea of using a clustering structure when designing a similarity metric was advanced by [**?**] who showed that link prediction measures based on structural similarity perform poorly for a network with a low clustering structure. This inspired [**?**] to first divide the network into communities, and use this clustering structure information in designing a similarity metric for the link prediction problem. In order to measure the quality of a community structure, Newman et al [**?**] introduced a modularity function Q. Given a social graph $G = (V, E)$, the modularity function can be expressed as follows:

$$Q = \frac{1}{2m} \sum (A_{ij} - \frac{k_i k_j}{2m}) \delta(C_i, C_j) \tag{2}$$

where k_i and k_j are respectively the degree of nodes i and j. A_{ij} represents an element of the adjacency matrix, and m is the size of E which is the set of edges of the given graph G. C_i and C_j are the respective communities to which i and j belong. The parameter δ is the Kronecker delta symbol whose value is 1 when both i and j belong to the same community, and is 0 when both nodes belong to different communities. The goal of community detection is to divide a network into communities in a manner that maximizes the value of the modularity. In our Sybil detection algorithm we use a modularity optimization method called the Louvain Method [**?**].

To identify clusters, we first collect all the edges with weight $w(u_1, u_2) \in [0, 1]$, and for each of these edges we compute the similarity value of its end nodes using the Within Inter Cluster (WIC) similarity metric [**?**]. This metric is built based on the notion of *within-cluster common neighbors* and *inter-cluster common neighbors*. For a given graph $G = (V, E)$, and nodes $u, v, w \in V$, w is said to be a within-cluster common neighbor of u and v if w belongs to the same community as them. Otherwise, w is said to be an inter-cluster common neighbor of u and v. The WIC metric is defined to be the ratio between the size of the set of within- and inter-cluster common neighbors [**?**].

Running the WIC similarity metric on edges with weight $w(u_1, u_2) \in [0, 1]$ results in this set of edges being reduced in size. Some of its edges are converted to edges with weight $w(u_1, u_2) > 1$ while the remaining are converted to edges with weight $w(u_1, u_2) = 0$, thus increasing the size of the set of attack edges. We terminate this preprocessing with a tuning that aims to scale down all weights $w(u_1, u_2) > 1$ to $w(u_1, u_2) = 1$. The benefit of this transformation is a gain in the accuracy and the stability of the detection mechanism. We are now ready to proceed to the ranking of nodes in order to declare which ones are Sybil nodes, and which ones are benign nodes.

4.3 Trust Propagation

To rank the nodes, each node in the OSN is assigned a *degree-normalized landing probability* of a modified short random walk. The walk starts from a known non-Sybil node. Using this node, we compute the probability of a modified random walk to land on each node u_i after k steps. This landing probability is analogous to the strength of the trust relationship between the nodes, and each step of the walk's probability distribution is considered as a trust propagation process [**?**].

Early terminated walk: The modified random walk used by SybilRadar is called a short walk because it is an early terminated walk [**?**]. A random walk that is run long enough will end up with with all the nodes in the social graph having an uniform trust value. The uniform trust value is called the convergence value of the random walk [**?**]. The number of steps k required for a random walk to converge is called the mixing time of the social graph. Several researches [**?**], [**?**], [**?**] have shown that for various social networks, the mixing time is larger than $O(\log n)$ with n being the number of nodes in the social graph. To compute the trust values, SybilRadar adapts the Power Iteration method [**?**]. In SybilRadar the modified power iteration is terminated after $O(logn)$ iterations.

Our modified power iteration method takes as input the transition matrix of social graph, where each element of the matrix is the probability of the random walk to transition from one node to another. The method is executed as a succession of transition matrix multiplications, and at each step the iteration computes the trust distribution over nodes. It works as follows. We define the trust value on a node v after i iterations as $T(v)$, and the total trust as the value $T \geq 1$. Given $s_1, \ldots, s_k$ the selected trust seeds, we initialize the power iteration by distributing the total trust among the trust seeds as follows:

$$T^{(0)}(v) = \begin{cases} T/k & \text{if } v \text{ is a trusted seed} \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

After the initialization step, each node v_i is assigned a trust value $T(v_i)$. The process then proceeds with each node v_i evenly distributing its trust value $T(v_i)$ to each of its neighbor v_j during each round of power iteration. Each node v_i then updates its trust value in accordance with the trust values received from its neighbors. The trust distribution is done proportionally to $w(v_i, v_j) \div deg(v_j)$ which is the ratio of the weight on the edge between the node v_i and its neighbor v_j over the degree of the neighbor node v_j. The use of the weight ensures that a big fraction of the total trust will be distributed to benign accounts rather to Sybil accounts. This results in benign accounts having higher trust value than Sybil accounts. The entire process is summarized in equation 4.

$$T^{(k)}(v_i) = \sum_{(v_i,v_j)\in E} T^{(k-1)}(v_j)\frac{w(v_i,v_j)}{deg(v_j)} \tag{4}$$

After $O(logn)$ iterations, the resulting trust value $T(v_i)$ assigned to each node v_i is normalized according to v_i degree. The normalization process involves dividing each node trust value by its degree. This transformation is motivated by the fact that trust propagation is influenced by the node degree, and that this results in the trust propagation being biased toward node with higher degree when the number of iterations grows larger. The normalization ensures that benign nodes get trust values that are close in value [**?**]. This is influential in identifying Sybil nodes after the ranking.

5 System Evaluation

We first evaluate SybilRadar using both a synthetic network and a real dataset collected from Facebook. For both evaluations we employ procedures that other researchers have used in this line of work. We compare SybilRadar against SybilRank which takes the same structure-based approach that is also based on the use of the power iteration method albeit on an unweighted graph unlike SybilRadar which uses a weighted graph.

Comparing SibilRadar to SybilRank will help highlight the role played by similarity metrics in detecting Sybil accounts. In addition, SybilRank has been demonstrated to outperform other previous structure-based methods [**?**]. Although Integro outperforms SybilRank, it is not a pure structure-based approach since it leverages account's feature information collected from recent users activities. We have indicated earlier our reservations for using user attributes or activities in Sybil detection. For this reason, we are not including it in our comparison.

Evaluation metric: To express SybilRadar's performance, we use the Area Under the Receiver Operating Characteristic Curve (AUC). AUC for our purpose is defined as the probability to have a randomly selected benign node ranked higher than a randomly selected Sybil node. The AUC is a tradeoff between the False Positive Rate and the True Positive Rate of the classifier. A perfect classifier has an AUC of 1 while a random classifier has an AUC of 0.5. Therefore, we expect our classifier to perform better than a random classifier, and to have an AUC as close as possible to 1.

5.1 Evaluation on Synthetic Networks

The synthetic network is generated using known social network models. First, the honest and the Sybil regions are generated by providing relevant parameters to the network model, like the number of nodes, and the average degree of nodes. Then, the attack edges are generated following the scenario chosen in the experiment. They can be randomly generated or generated in a way to target some specific honest nodes.

Initial Evaluation: We generate the honest region and the Sybil region using the Powerlaw model. The honest region has a size 4000 nodes while the Sybil region has

400 nodes. Both regions have an average degree of 10. The attack scenario chosen simulates an attacker randomly generating 2000 attack edges. The weights on the edges are set to be the values resulting from the two similarity metrics previously described in this section 4.2. For this experiment, we select 20 trust seeds from the honest region. These are supposed to be some nodes that the OSN system administrator is absolutely certain to be honest nodes.

Results: Comparing the ranking quality of both SybilRank and SybilRadar under the chosen scenario, the results show that SybilRadar outperforms SybilRank. SybilRadar resulted in an AUC which is always greater than 0.95, an AUC that is higher than SybilRank's AUC of 0.90.

Varying the number of attack edges : In the next experiment, we keep the honest and the Sybil regions as set up in the previous Basic Evaluation. In order to stress-test the platforms being compared, we decide to successively vary the number of attack edges from 1000 to 10000. We want to investigate how the increase in number of attack edges affects the performance of both platforms.

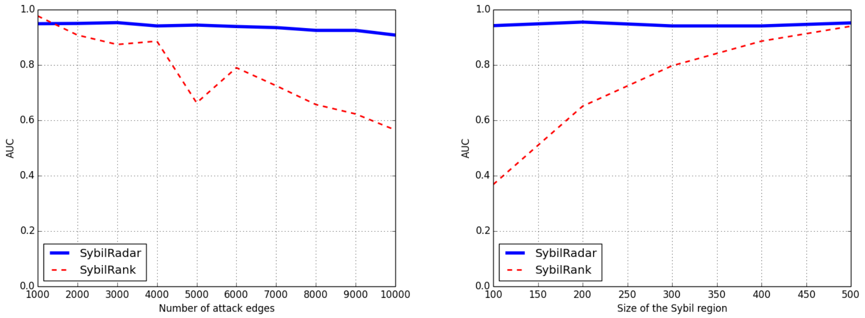
Results: This result can be seen in Figure 1(a). As the number of attack edges increases, we notice that SybilRank is unable to keep its initial performance, with its AUC dropping from 0.97 to less than 0.6. Meanwhile, the increase in the number of attack edges affects the performance of SybilRadar only marginally. Its AUC still stays above 0.90. This highlights the effectiveness of using similarity metrics in detecting Sybil nodes in the case of social graphs with weak trust.

Varying the size of the Sybil region : In this experiment, we explore how the increase in the size of the Sybil region affects the performance of both platforms. For this purpose, we design a honest region with 4000 nodes, and an average degree of 10. The attacker is able to create randomly 4000 attack edges. We vary the size of the Sybil region from 100 to 500 nodes each with an average degree of 10.

Results: The experiment results (see figure 1(b)) show that SybilRadar and SybilRank react differently to the increase in the size of the Sybil region. When the size of the Sybil region is relatively small compared to the size of the honest region, SybilRank performs poorly. SybilRank performance improves when the size of the Sybil region get relatively bigger. However, as illustrated in figure 1b, SybilRadar displays a stable performance that is less sensitive to the size of the Sybil region.

5.2 Evaluation on Real-world Twitter Network

To study if our choice of data in the previous experiments biased our results, we also evaluated the performance of SybilRadar under larger datasets from a different OSN, namely, the Twitter network. The dataset we used is a combination of four datasets: The FakeProject dataset, the Elezioni2013 dataset, the TWT dataset, and the INT dataset [**?**]. The FakeProject dataset contained profiles of real users who received friend requests from @TheFakeProject, an account created for The FakeProject that was initiated in 2012 at IIT-CNR, in Pisa-Italy. The Elezioni2013 dataset was generated in 2013 for a sociological research undertaken by the University of Perugia and University of Rome, La Sapienza. The TWT dataset and the INT dataset



(a) Varying number of attack edges

(b) Varying size of the Sybil region

Fig. 1: Performance on synthetic data

were a set of fake accounts purchased respectively from the fake accounts providers `http://twittertechnology.com` and `http://intertwitter.com`. The first two datasets mentioned provided the honest nodes while the last two datasets provided the fake nodes [**?**].

Pre-processing: Since the Twitter network is directed, we considered only the set of bidirectional edges. This provided us with an initial network of 469,506 nodes and 2,153,427 edges. We further refined this network by removing all nodes with degree less than 1. The resulting twitter network then comprised 135,942 nodes and 1,819,864 edges. The honest region comprised 100,276 nodes and 634,127 edges while the Sybil region was constituted of 35,666 nodes and 1,086,352 edges. The two regions were connected by 99,385 attack edges.

Results: We ran SybilRadar several times using the Twitter dataset described above. SybilRadar resulted in an AUC which was always greater than 0.95 as shown in figure 2.

6 Conclusion

In this paper, we presented a new framework for detecting Sybil attacks in an Online Social Network. In a Sybil attack, an adversary creates a large number of fake identities in an OSN or forges existing identities. The adversary then uses these fake identities to influence the underlying trust basis of the OSN and perform malicious activities such as social spamming, malware distribution and private data collection. Sybils are a significant threat to the OSN. While they cannot be prevented in most OSNs because of their open nature, this work provides a solution by which the OSN operator can automatically, speedily and accurately detect such Sybils.

SybilRadar belongs to the class of Sybil detection techniques that rely on the graph structure of the OSN. This is in contrast to the alternate group of detection mechanisms that rely of identifying features related to user attributes and activities. We believe that while the second class of detection algorithms may provide good detection results on carefully cleaned up OSN data, in real life such data is difficult to

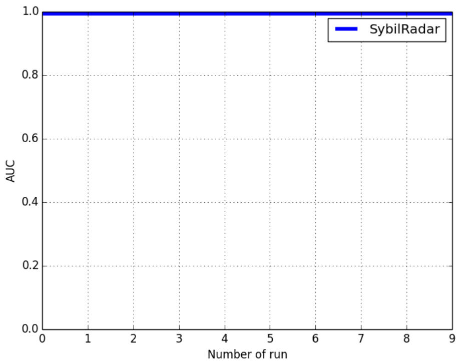


Fig. 2: Performance on Twitter dataset

obtain since OSN users frequently leave their profiles incomplete or use misleading information purposefully. Moreover, trying to obtain user activity related data may raise serious privacy concerns. As a result, SybilRadar relies on just the structural properties of the OSN graph. We used a variety of OSN test data – both synthetic as well as real-world – to evaluate the detection accuracy of SybilRadar. Our experimental results show that SybilRadar performs very well – much better than the most well known similar technique – even for OSNs that have the weak trust model and which have a very large number of attack edges between Sybil nodes and honest nodes.

For future work, we plan to add a temporal dimension to our detection framework. Sybil behavior will most likely not be static but change with time. We expect to see major differences in how structural properties of honest nodes change over time and how that of Sybil nodes change. We would like to investigate how this can be modeled to detect Sybils. Also, although we are not a big supporter of using user attributes and activities in Sybil detection, we admit that these techniques can provide somewhat better results. We would like to investigate if and how these techniques can be integrated with SybilRadar so as to improve it but in a manner that does not raise any privacy issues related to OSN users.

References

1. L. Adamic and E. Adar. Friends and Neighbors on the Web. *Social Network*, 25:211–230, 2001.
2. E. Behrends. *Introduction to Markov Chains with Special Emphasis on Rapid Mixing. 2000*. Advanced Lectures in Mathematics. Springer, 2000.
3. V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast Unfolding of Communities in Large Networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008, 2008.

4. Y. Boshmaf, D. Logothetis, G. Siganos, J. Lería, J. Lorenzo, M. Ripeanu, and K. Beznosov. Integro: Leveraging Victim Prediction for Robust Fake Account Detection in Osns. In *Proc. of NDSS*, 2015.
5. Y. Boshmaf, I. Muslukhov, K. Beznosov, and M. Ripeanu. The Socialbot Network: When Bots Socialize for Fame and Money. In *Proceedings of the 27th Annual Computer Security Applications Conference*, pages 93–102. ACM, 2011.
6. Q. Cao, M. Sirivianos, X. Yang, and T. Pregueiro. Aiding the Detection of Fake Accounts in Large Scale Social Online Services. In *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation*, pages 15–15. USENIX Association, 2012.
7. W. Chang and J. Wu. A Survey of Sybil Attacks in Networks. Technical report, Department of Computer and Information Science, Temple University.
8. S. Cresci, R. Di Pietro, M. Petrocchi, A. Spognardi, and M. Tesconi. Fame for Sale: Efficient Detection of Fake Twitter Followers. *Decision Support Systems*, 80:56–71, 2015.
9. G. Danezis and P. Mittal. Sybilinfer: Detecting Sybil Nodes Using Social Networks. In *NDSS*. San Diego, CA, 2009.
10. M. Dellamico and Y. Roudier. A Measurement of Mixing Time in Social Networks. In *Proceedings of the 5th International Workshop on Security and Trust Management*, Saint Malo, France, September 2009.
11. N. B. Ellison et al. Social Network Sites: Definition, History, and Scholarship. *Journal of Computer-Mediated Communication*, 13(1):210–230, 2007.
12. X. Feng, J. Zhao, and K. Xu. Link Prediction in Complex Networks: A Clustering Perspective. *The European Physical Journal B*, 85(1):1–9, 2012.
13. M. Fire, L. Tenenboim-Chekina, R. Puzis, O. Lesser, L. Rokach, and Y. Elovici. Computationally Efficient Link Prediction in a Variety of Social Networks. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 5(1):10, 2013.
14. H. Gao, J. Hu, C. Wilson, Z. Li, Y. Chen, and B. Y. Zhao. Detecting and Characterizing Social Spam Campaigns. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*, pages 35–47. ACM, 2010.
15. P. Gao, N. Z. Gong, S. Kulkarni, K. Thomas, and P. Mittal. Sybilframe: A Defense-in-Depth Framework for Structure-Based Sybil Detection. *arXiv preprint arXiv:1503.02985*, 2015.
16. S. Geisser. *Predictive Inference*, volume 55. CRC Press, 1993.
17. S. Ghosh, B. Viswanath, F. Kooti, N. K. Sharma, G. Korlam, F. Benevenuto, N. Ganguly, and K. P. Gummadi. Understanding and Combating Link Farming in the Twitter Social Network. In *Proceedings of the 21st international conference on World Wide Web*, pages 61–70. ACM, 2012.
18. J. Golbeck. Trust and Nuanced Profile Similarity in Online Social Networks. *ACM Transactions on the Web (TWEB)*, 3(4):12, 2009.
19. G. H. Golub and H. A. Van der Vorst. Eigenvalue Computation in the 20th Century. *Journal of Computational and Applied Mathematics*, 123(1):35–65, 2000.
20. C. Grier, K. Thomas, V. Paxson, and M. Zhang. @ Spam: The Underground on 140 Characters or Less. In *Proceedings of the 17th ACM conference on Computer and communications security*, pages 27–37. ACM, 2010.
21. I. Kahanda and J. Neville. Using Transactional Information to Predict Link Strength in Online Social Networks. *ICWSM*, 9:74–81, 2009.
22. J. Leskovec, K. J. Lang, A. Dasgupta, and M. W. Mahoney. Community Structure in Large Networks: Natural Cluster Sizes and the Absence of Large Well-Defined Clusters. *Internet Mathematics*, 6(1):29–123, 2009.
23. R. N. Lichtenwalter, J. T. Lussier, and N. V. Chawla. New Perspectives and Methods in Link Prediction. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 243–252. ACM, 2010.

24. A. Mohaisen, A. Yun, and Y. Kim. Measuring the Mixing Time of Social Graphs. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*, pages 383–389. ACM, 2010.
25. S. Nagaraja. Anonymity in the Wild: Mixes on Unstructured Networks. In *Privacy Enhancing Technologies*, pages 254–271. Springer, 2007.
26. M. E. Newman and M. Girvan. Finding and Evaluating Community Structure in Networks. *Physical review E*, 69(2):026113, 2004.
27. L. Shi, S. Yu, W. Lou, and Y. T. Hou. Sybilshield: An Agent-Aided Social Network-Based Sybil Defense among Multiple Communities. In *INFOCOM, 2013 Proceedings IEEE*, pages 1034–1042. IEEE, 2013.
28. T. Stein, E. Chen, and K. Mangla. Facebook Immune System. In *Proceedings of the 4th Workshop on Social Network Systems*, page 8. ACM, 2011.
29. G. Stringhini, C. Kruegel, and G. Vigna. Detecting Spammers on Social Networks. In *Proceedings of the 26th Annual Computer Security Applications Conference*, pages 1–9. ACM, 2010.
30. K. Thomas, C. Grier, D. Song, and V. Paxson. Suspended Accounts in Retrospect: An Analysis of Twitter Spam. In *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, pages 243–258. ACM, 2011.
31. J. C. Valverde-Rebaza and A. de Andrade Lopes. Link Prediction in Complex Networks Based on Cluster Information. In *Advances in Artificial Intelligence-SBIA 2012*, pages 92–101. Springer, 2012.
32. A. H. Wang. Don'T Follow Me: Spam Detection in Twitter. In *Security and Cryptography (SECRYPT), Proceedings of the 2010 International Conference on*, pages 1–10. IEEE, 2010.
33. G. Yan, G. Chen, S. Eidenbenz, and N. Li. Malware Propagation in Online Social Networks: Nature, Dynamics, and Defense Implications. In *Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security*, pages 196–206. ACM, 2011.
34. Z. Yang, C. Wilson, X. Wang, T. Gao, B. Y. Zhao, and Y. Dai. Uncovering Social Network Sybils in the Wild. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 8(1):2, 2014.
35. F. Yao and L. Chen. Similarity Propagation Based Link Prediction in Bipartite Networks. In *Network Security and Communication Engineering: Proceedings of the 2014 International Conference on Network Security and Communication Engineering (NSCE 2014), Hong Kong, December 25–26, 2014*, page 295. CRC Press, 2015.
36. H. Yu. Sybil Defenses Via Social Networks: A Tutorial and Survey. *ACM SIGACT News*, 42(3):80–101, 2011.
37. H. Yu, P. B. Gibbons, M. Kaminsky, and F. Xiao. Sybillimit: A Near-Optimal Social Network Defense against Sybil Attacks. In *Security and Privacy, 2008. SP 2008. IEEE Symposium on*, pages 3–17. IEEE, 2008.
38. H. Yu, M. Kaminsky, P. B. Gibbons, and A. D. Flaxman. Sybilguard: Defending against Sybil Attacks Via Social Networks. *Networking, IEEE/ACM Transactions on*, 16(3):576–589, 2008.
39. X. Zhao, L. Li, and G. Xue. Authenticating Strangers in Fast Mixing Online Social Networks. In *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*, pages 1–5. IEEE, 2011.