A Survey on Sybil Attacks and Defenses

Xinghuang Xu
EECS Department
Wichita State University
Email: xxxu3@wichita.edu

***Abstract*—Sybil attack has been a thread to most peer to peer network systems. If new identities can be created without control in a distributed system, the system is susceptible to Sybil attacks. For example in IMDB, sybil accounts can be created to boost the score of a new movie in order to attract potential audiences to watch the movie in theatres. The survey paper is a guideline for open distributed system designers who want to introduce defense mechanisms into their systems to protect against Sybil attacks. We first define various Sybil attacks under different domains with different goals then we presents three categories of defenses against Sybil attacks. The three categories include the traditional approach, the social network based approach and the domain specific approach. We also analyse the differences among the three categories and the advantage/disadvantage of methods within each category. Readers will have a deeper understanding of how to protect their distributed systems against Sybil attacks after reading this survey.**

I. Introduction

Sybil is book written by Flora Rheta Schreiber about the treatment of Sybil Dorsett for dissociative identity disorder. She is believed to have manifested sixteen different personalities according her doctor Cornelia B Wilbur.[1]

This survey is about a specific system security attack name Sybil Attack. Sybil attack takes place when an adversary in a system acts as if he is multiple users with different identities in order to disrupt the proper functionality of the underlying system and benefit himself. Sybil attack has proven to be harmful in many systems.

Sybil attack was not possible when distributed systems were closed in nature. A good example is the flight reservation system used by airlines. The distributed airline reservation systems are closed in nature because they have a static set of nodes and there is a strict process involving human administrators to add a new node to the system. The users of the system are bound to real human beings. Things are quite different today due to the advancement of Internet and other peer-to-peer technologies, open access systems are becoming the norm in many domains. The benefit of open access system is that it would enable the penetration of a specific technology. Imaging if there is a strict process for creating accounts that involve human inspection on Facebook, it would take a lot longer for it to reach today's 1 billion users.

While open distributed systems were the original target of Sybil attack, centralized systems are vulnerable to it as well. For example, in a centralized book recommender system like GoodRead. The popularity of a book depends on the number of people who have liked it. In such a system, the goal is to find books that is likely to be of interest to users based on others' recommendations. An attacker can create many fake accounts and out vote legitimate users on the books he/she wants to promote or demote. The success of the attack is almost guaranteed given the fact that most legitimate users are too lazy to vote in a recommender system. Sybil attacks influence many systems, and we try to classify sybil attack based on the domains they target because attacks in the same domain tend to have similar attack methods and attack goals. It's impossible to enumerate all the domains susceptible to Sybil attack in this survey, so we have selected some typical domains that are well studied in section III. If your domain is not listed in III, that doesn't mean your domain is free from Sybil attacks. You should ask yourself if it's possible to create fake identities in your system, if your answer is yes then you should keep an eye open for Sybil attacks.

The following is a roadmap of the rest of the paper. In section II, we define some common properties of sybil. In section III, a list of some major domains that are susceptible to Sybil attacks are provided. We define the goal of the defense and the general and domain specific defense mechanisms in section IV. There are two kind of general sybil defenses: traditional defense and social network based defense. We provide descriptions on how the defense approaches work and point our their advantage/disadvantage in terms of their efficiency, false positive/negative rate, deployability and more. In the last section VII, we conclude this survey. After reading this survey readers will have a deeper understanding of Sybil attacks and should be equipped with many techniques to defense against Sybil attacks.

II. Attack Model

The context of the attack can be vastly different. Sybil attacks can take place in many different domains such as Wireless Sensor Network, Online Social Network, Reputation System, Ad hoc Mobile Network and etc. The goal of the attacker can vary too. The goal of an attacker can either be to control the system for self benefit or to subvert the normal functionality of the system. Attackers with the goal to manipulate the system will create sybil nodes that camouflage themselves as honest nodes and acts like honest nodes. For example, in an Online Social Network, the attacker can gradually create sybils and make them looks like real users by using other people's online profile and daily posts etc. Some attackers would go further to ensure their fake accounts act like regular users with regular logins, friend requests, friend request acceptances and real user like click streams. After the

sybils have made enough connection with honest users, they are start spreading news or malware to disrupt the targeted online social network. If the goal of the attacker is to subvert the targeted system, he/she will usually inject as many bad players into the system as possible. Sybils are like bombs that are being hide in the system and when they explode at once, the system could be destroyed.

In sum, attackers can create sybils quickly or gradually. Sybils can have a short or long camouflage periods when they act as honest players. Sybils can launch the attack all at once or they can misbehave one at a time. Sybils can have no connection within themselves or they can form relationship between each other to form a group.

III. Sybil Attacks

A. Routing System

Routing is an essential part of all distributed systems. Many P2P systems use DHT to store routing tables. There are two major strategies to perform routing table attacks in such P2P systems named horizontal attack and vertical attack according to [2]. Assume a setting of Kademlia-based system[3] where the system has N nodes, each node maintains a k-bucket routing table and the average number of hops in routing a message in the system is $O(log(N))$.

A horizontal attack aims at polluting as many routing tables as possible through spreading sybils widely across the whole system. Based on the k-bucket mechanism, an attacker need only to control at least one sybil among a node's neighbours to effectively intercept messages. To launch a successful horizontal attack, an attacker would need to roughly inject $\frac{N}{min(k,O(log(N)))}$ sybils into the system to hijack the whole system. The next question concern the attack is how much resource is needed to perform this attack. A straightforward way to do it is to run a sybil instance in one machine but this is highly inefficient. By modifying the DHT client, attackers can run many sybil instances simultaneously on a machine. Moreover, by exploiting the hopping technique in [4], sybils can change their id periodically and jump to a new location in the DHT. By jumping into new locations periodically, a sybil instance can cover a lot more area than a static sybil.

On the other hand, vertical attack tries to insert as many sybils as possible into one specific routing table. Using vertical attack, a specific content ID can be targeted. This attack would be made more difficult with a very large DHT and if the DHT protocol assign random ids to newly join nodes but this is not the case in Mainline DHT(MLDHT)[2]. MLDHT allows nodes to pick their own IDs and this security weakness has been around for over a decade and no one care enough to fix it. Given a target Id, sybils can generate ids that will locate them close to the target and 'isolate' the target from the others.

The authors in [2] also mention a hybrid approach in which attacks would first launch a horizontal attack to take control over the whole system then target individual nodes with vertical attack. In sum, a hybrid attack would lead to the attacker controlling the whole system.

There are other routing protocols that are vulnerable to Sybil attack. Geographical routing protocol requires nodes to exchange coordinate data with their neighbours to efficiently address packets. By using Sybil attack, an attacker can create multiple identities in different geographical locations thus making him available in multiple places at once which violates the fundamental assumption of the geographical routing protocol[5]. Sybil attack poses a thread to the seemingly robust multipath routing protocol too. For more detail please see [5].

B. Content Rating System

Sybil attack is a fundamental thread to many user-based content rating system such as GoodReads, YouTube and IMDB. There are huge incentives in this kind of attacks because attackers can promote low-quality content to a wide audience. It has been studied that many people check the IMDB score before going to the movie theatre. A high IMDB score will attract more audiences thus making the movie more profitable. This is not hypothetical. There are successful real world cases. For example, the famous Slashdot poll on the best computer science school has caused students to write automatic scripts to vote for their schools repeatedly. Moreover, some underground companies made money through assisting clients in promoting their YouTube video's view counts by using a large number of Sybil accounts.[6]

C. Online Market Place

Sybil attack has posted a significant challenge for building reputation systems in online market place. In a reputation system, an adversary can create a large number of identities and maliciously increase the reputation of one or more master identities by giving false recommendations to them. Sybils can also promote their own reputations and falsely accuse well-behaved players in the system to hurt their reputation. For example, in eBay.com reputation is calculated as the sum of (+1,0.-1) of all the transaction ratings no matter how big the transaction is. Sybils can be create to make small transactions with a seller and automatically give them good reviews to boost their reputation. Afterwards, the seller can use that reputation on a dishonest transaction of high value. By using Sybil attack, a dishonest seller can hide the fact he frequently misbehaves at a certain rate.

Moreover, in networks that use reputation scheme to find misbehaving nodes/sybils, nodes with good reputation can report nodes they believe to be misbehaving in its neighbours. But this scheme can backfire. For example, users can collude to artificially boost the reputation values of one or more friends, or falsely accuse well-behaved users of misbehaviour. When adversaries control enough sybil nodes and decide to repeatedly report honest nodes. The outcome is that most of the honest nodes will be considered malicious and be removed from the networks, the malicious nodes will take full control of the whole system and use it for their own benefits. Detecting such collusion attacks is yet an unsolved problem that severely limits the impact of existing reputation systems.[7][8]

D. Resource Sharing System

Sybil attack can be used to gain a disproportional share of resources in P2P network. In a P2P system, people can share their resources such as bandwidth, memory, computation power and file. An adversary can create sybils to claim an unfair and disproportionate share of the resources that were intended to be distributed amongst all nodes in the system. For example, in a public cloud infrastructure like Amazon Web Services, Google Cloud, each user is eligible for a free 15GB of data storage. An attacker can create 100 or more sybil accounts and claim more than 1500GB of free storage. Let's consider a distributed file sharing system, the download speed depends on how much credit a user. To obtain credits, a user needs to make contribution and upload files to others. This seems like a fair file sharing system until an attacker creates many sybil accounts and generate credits from download/upload files between themselves. With Sybil attack, the attacker can generate unlimited credits and use them to quickly download files from others.

E. Distributed Storage System

In distributed storage system, nodes are required to score a fragment of a file and each fragment is duplicated into multiple machines to prevent data lost and increase the performance of file download speed. Sybils can cause data lost by being selfish and not storing the fragment of data that they are asked to store. Sybils can also degrade the performance of the distributed file system by not responding to file request or provide the wrong file segment. What's more, because some distributed file systems replicate data to neighbouring nodes, sybils can be used to crawl the entire file system through frequently hopping to different locations in the network and obtain data fragments from all its neighbours.[8][4]

F. Online Social Network

Online Social Networks(OSN) like Facebook and Twitter are vulnerable to Sybil attack as well. One goal of the attacker can be to crawl the OSN's user's personal data. Personal data includes name, phone number, age, address etc. In order to obtain those personal information attackers need to be friends with the actual users to be able to see those information. An attacker in this case will create a lot of fake accounts that camouflage themselves as real users. New sybil accounts can be created by copying the information of some of the victims who befriended a sybil account and have their personal information stolen. New sybils will continue to send friend requests to other people and some will even befriend themselves. After using sybils to crawl users' account information, attackers can then make a profit from selling those information. Another goal of attacking the OSN can be to spread spam. More and more people these days use facebook and twitter as their news channel and read posts on the Social network as they read news paper. After successfully befriend many users and their friends, a sybil account can more effectively spread spam or even malicious files.

G. Collaborative Mobil Application

People these days have spent more time on portable devices such as their smart phone or tablet than on their computers. Researchers have recently proposed general infrastructures for portable devices within proximity of each other to trade various services in a scalable and decentralize way without going through any Internet server. Collaborating devices can synchronize their times, run localization algorithms that increase the precision of street map software, borrow each other's bandwidth, or even share cached web content. The problem with this model is that it can easily be disrupted by uncooperative and malicious sybils. Those who only want to profit from these services and not providing anything in return. They can usually create a number of sybil identities to avoid being tracked down since no identity certification authority is involved in this kind of model.[9]

IV. Sybil Defense Overview

A. Defense Goal

Before going into the details of sybil defenses, the defense goal should be defined. It should be obvious that the ideal goal is to eliminate Sybil attack but this goal might not be realistic deal to the fact that we don't want to enforce a strict rule whenever people join the our system. Without a centralized trusted certification based scheme, the best we can do is to restrict the effects of sybils. False positive rate and false negative rate can serve as good metrics when evaluating a defense approach. False positive happens when a honest node is identified as sybil using the defense mechanism and false negative happens when a sybil node is not detected and treated as a honest user. So the goal in the following defense mechanisms is to minimize the false positive and false negative rates as much as possible.

B. Classification of Defenses

There are general defense mechanisms that work in most domains and there are domain specific approaches that target a specific domain and works better in that domain by levering some domain specific features. We further divide the general approach into traditional approach and social network based approach. Traditional approach were approaches found between 1999 to 2004. They focus mainly on building identity authentication, resource testing or human assisted sybil detection mechanisms into existing systems. On the other hand, social network based approach started around 2006 tries to detect sybils by exploiting key social network structures underlying the system.

V. General Defense Approach

There are two types of general approaches. The traditional approach and the social network based approach.

A. Traditional Defense

Traditional approach is based on research starting around 1999 and end around 2004. During that period of time, researchers have focused on preventing Sybil attacks by involving secure mechanisms such as digital signatures and identity authentication. Other methods have also been found to increase the resource cost of a Sybil attacks such as resource testing and recurring cost. Moreover, the human assisted sybil detection approach has been widely deployed during that time.

1) Trusted Certification: This is the most popular solution for countering Sybil attacks, it requires a trusted certifying authority to validates the identity of a node before it joins the system. There are two variations in this approach. One is the centralized version, the other is the semi-centralized version. In the centralized version, it is assumed that there is a trusted central authority who can verify the validity of each participant. After the validation, a certificate will be given to each participant. The participant then can use the certificate to access the system. The model is very popular and has been used widely. Most authentication services use employ this sybil defense approach and it can completely eliminate sybil attack. Given the obvious advantage, this centralized defense mechanism is very susceptible to single point failure and could be the performance bottleneck to the whole system. For example, any down time of the central authentication service will bring down the whole system.

The semi-centralized approach seek to cut the cost of asymmetric cryptography used in the centralized version. It leverage a technique called partial identity verifications. This approach still need to rely on a trusted base station but reduce the involvement of a third party authority. The semi-centralized version has better performance than the centralized version given that it distribute the service workload to a group of base stations spread out in different locations of the system. The disadvantage compared to the centralized version is that it would allow a small amounts sybil entering the system due the fact that there could be a mismatch of the register list in each trusted base station. Moreover, the semi-centralized approach is harder to implement and maintain.

The common problem to the trusted certification approaches is that they rely on one or more centralized trusted authorities for credential generation, assignment and verification. However, this type of approaches sacrifice the open nature that underlies the success of many systems and increase the overhead in joining those systems.[10][11][12].

2) Resource Testing and Recurring Cost: Resource Testing is another line of solution. The idea behind resource testing is that each identity should own a fair amount of resource because it runs on a legitimate client otherwise there is a high potential that this is a sybil node. The question is how can we test that there are resource backing up a node? Some propose the testing of IP address because multiple identities sharing a single IP address is a good sign of Sybil attacks. Others test resources such as computing power, network bandwidth, MAC address. This approach in theory should work for systems that are in very low risk. Its easy to implement resource testing or recurring cost approaches but people these days can acquire a large amount of resources in a short period of time with the help of public cloud. Resource testing is mostly obsolete when an attacker can spin of hundreds of EC2 instances in a short period of time and terminate them after a few hours of attack. As of this writing, the EC2 t2.nano instance will only cost $0.0043 per hour with upfront payment.

A variation of the resource testing method is called "Recurring Costs". For example, in one solution participants are required to perform some tasks such as solving puzzles[13] periodically. The biggest disadvantage of computational puzzle is that it will prevent honest users with old computers from joining the system. Turing tests like CAPTCHA are also suggested as a recurring cost solution[14]. Using this approach, the cost of Sybil attacks have become more expensive but would the benefit still outweighs the cost? This approach is not recommended in high risk system for the following reasons. Computational puzzle can hold back entry level attacks but not advance attacks that leverage the public cloud's computing power. How about turing tests? Is it not a lot of Online Social Network sites still use this approach? With crowdsourcing services like Amazon Mechanical Turk, turing tests can be crowdsourced at a reasonably low price. A quick Google search shown that the market price to solve a 1000 CAPTCHAs is just a few dollars.

3) Human Assisted Approach: We believe the human assisted approach has been the oldest sybil defense approach ever. This approach has been used to fight against identity theft. Even today, there is still no automatic way to identify fake identity. Online social network sites have long been using this approach to find fake accounts due the the failure in automated fake account detection. This approach starts with the honest users report a potential fake accounts. Afterwards, an account police start the manual inspection involving matching profile photos to the age or address, understanding natural language in posts, examining the friends of the user, etc. This approach is time consuming. Tuenti a Spanish social network receives on average 12,000 reports regarding fake accounts per day and only about 5% of them are indeed fake. This approach can effectively offload some of the sybil detection work to its honest users but should be kept to its minimum by combining it with another automated sybil detection technique.

B. Social Network-Based Defense

Social network based sybil defense (SNSD) aim at leveraging the underlying social structure within a system to defend against Sybil attacks. Why is this a general approach not a domain specific approach? The reason is that in most systems we can modify the underlying communication protocol and enforce certain social structures that would help in preventing Sybil attacks. For example, in a file sharing P2P system, system designers can enforce some social structure between users. Connections can only be built based on trust between users. If you think it is too much trouble to implement a social network on top of your system you could offload this task to

some online social network site through asking your user to login to your system through facebook or google+ accounts.

C. Overview

Yu et al. have started a new era of sybil defense when they proposed the idea of detecting sybils using a unique structure in the social network graph in their first paper [15] in 2006. We will provide some definitions and intuitions in this overview.

Social network can mean different things in different context. Here we tries to grasp the essence of social network when it comes to sybil defenses. A distributed system D as a undirected graph G, each node in G represents an identity in D. An edge between tow nodes in G corresponds to human established trust relations between the two corresponding identities in the real world. We call a subgraph of G containing of only honest nodes a Honest Region H, and subgraphs of G containing only sybil nodes a sybil region S (Figure 1). The edges between H and S are called attack edges. By converting D into G, we convert the sybil defense problem into finding sybils in G where G is a social network. [16]

The following is three key insights for the social network based protocol proposed for sybil detection in G:

1) The number of attack edges is independent of the number of sybil identities, and is limited by the number of trust relation pairs between malicious users and honest users.
2) If the malicious users create too many sybil identities and given that the number of attack edges remains fixed, there exist a small quotient cut, a small set of edges whose removal disconnects a large number of nodes (all the sybil identities) from the rest of the graph.
3) There is known honest node used for breaking the symmetry when sybil group has similar or bigger size then the honest group.

We will use the following notations in describing the rest of the social network sybil defense techniques:

n	total number of nodes in the honest region
m	total number of edges in the honest region
g	total number of attack edges
t	mixing time of the honest region
w	length of the random walks

Assumption 1. *The honest region of G has a mixing time no large than t(n), where t(n) is a function of the size of the honest region. [16]*

Because finding small cuts in graph is a NP-hard problem, SNSD converts the problem to something relating to mixing time. Mixing time describes how fast random walks in a given graph reach its stationary distribution. Large mixing time means that it would require a longer random walk to approach the graph's stationary distribution. A connected graph having a small quotient cut is guaranteed to have a small conductance and thus a large mixing time. This means too many sybils nodes in G will increase the mixing time of G. Because of Assumption 1, the end guarantees of SNSD depend on the definition of t. With smaller t, there will less sybil nodes but more honest nodes falsely labelled as sybil nodes. With larger t, there will be more sybil nodes allowed in G and less honest nodes will be regarded as sybil nodes. [16]

D. SybilGuard and SybilLimit

SybilGuard[15] and SybilLimit[17] both designed by Yu et al. are the earliest SNSD techniques. Both approaches based on Assumption 1 and leverage random walk techniques to exploit the abnormal mixing time to bound the number of sybil groups and the size of a sybil group a malicious user can create in G. Since both approaches are very similar and SybilLimit is an enhancement on top of SybilGuard we will mainly cover SybilGuard here.

SybilGuard assumes that there is known honest node. From this original honest node, we initialized k random paths with a length of w = $O(nlog(n)))$. The reason behind the selection of the random walk length is well explained in [15]. . We call the paths originated from the honest node 'verifiers'. To decide if a suspect nodes is honest or sybil, we start k random paths from the suspect node and if a suspect path meet a verifier we believe this path is verified once. If a suspect path meet up with s verifiers where s is predefined value, we believe the path is a trusted path. The the majority of the paths coming out from a suspect node is trusted path, the suspect node will be treated as honest node otherwise it will be classified as sybil node. SybilGuard works well because most of the random paths originate from a suspect node is likely to stay in their own region, either honest region or sybil region due to the fact that there are very few bridges (attack edges) connecting the two regions. The above is a simplified explanation of SybilGuard.

SybilLimit is an enhancement on SybilGuard. Each node generates in SybilLimit generates $O(\sqrt{m})$ random paths with length w = $O(log(n))$. Instead of using node as intersection points, SybilLimit use edges. SybilLimit provides a near-optimal guarantees. SybilLimit allows $O(log(n))$ sybil nodes while SybilGuard would tolerate $O(\sqrt{n}log(n))$

E. SybilInfer

SybilInfer [18] protocol is similar to SybilLimit and SybilGuard. They all assume sybil nodes will increase the mixing time of the graph and they all use random walk. However, SybilInfer does not provide a complete decentralized design (a centralized knowledge of G is assumed) and does not have a provable end guarantee.

SybilInfer works by first converting G into a new Markov chain G', so that the node stationary distribution on G' is uniform. The transformation is done through changing the transition probability on each edge from 1/degree(A) in G to min(1/degree(A),1/degree(B)) in G'. [18] assumes without proving the if G has small mixing time than G' has small mixing time as well. For each node A , a number of random walks are initiated with a walk length of w = t' where t' is the mixing time of G'. SybilInfer defines the tail distribution of A as the distribution of the last nodes visited by all the random walks originated from A. After obtaining the tail distribution

of each node, SybilInfer computes a set H where each node in H has a tail distribution similar to a uniform distribution of H. All nodes in H will be labelled as honest and the other nodes will be regarded as sybils. Sybilnfer provides no prove on end guarantees such as false negative or false positive rates but it conduct an experiment with 1000 nodes and show that it significantly outperforms SybilLimit. The above is a conceptual explanation of how SybilInfer works and the actual implementation is quite different, please see [18].

F. SumUp and Gatekeeper

Gatekeeper [19] is an enhancement on both SumUp and SybilLimit but it requires that the honest region of G to be reasonably balanced. Gatekeeper is also a decentralized version of the centralized SumUp approach. We will first cover how Gatekeeper works than come back to SumUp.

Starting with some random roots, Gatekeeper grows each root node using special breadth first search. Each root node is initialized with $\theta(n)$ tickets and each node in G will received tickets from its parents. For a specific node A, after receiving tickets from its parents, A will consume one ticket then it will distribute the rest of the ticket evenly to its neighbors. GateKeeper will label a node honest if it receives tickets pass a threshold after the growth of all the root nodes, otherwise it will be regarded as sybil.

The upper bound for the number of sybils allowed in Gatekeeper depends on g, where g is the number of attack edges. When g is large such as $\theta(n^{1-c})$ where $0 < c < 1$, the guarantee is the same as SybilLimit $O(log(n))$. If g is close to $O(n/log(n))$, only $O(log(g)$ number of sybils will be mislabelled as honest. With a small g where g= $\theta(1)$, the bound is $O(1)$ which is asymptotically better than SybilLimit. The disadvantage of Gatekeeper as we have pointed out earlier is that it requires the honest region to be reasonably balanced.

SumUp [20] is a centralized version of GateKeeper and does not require the honest region to be reasonably balanced. It uses adaptive maximum flow on the social network to label hones nodes. It has a sybil tolerance similar to GateKeeper, but it runs a lot slower than SybilLimit and GateKeeper.

G. Summary

Assumption 1 is where all the SNSD approaches rely on. There are many more SNSD approaches that based on Assumption 1 but utilize different techniques in detecting sybils. We will not be able to cover all those approaches in detail due to the length of this survey, but we will try our best to summarize them here.

Other than random walk, community detection technique is another technique we can used to find H and S regions. The problem of community detection has been well studied for decades and people have proposed ways to find the minimum cut between H and S regions using state of the art community detection algorithms. SybilDefender[21], SybilShield[22] and VoteTrust[23] represent some of the best research efforts in utilizing community detection algorithms in sybil detection. The key drawback of this type of approach is that it does not provide any formal end guarantee on false positive and false negative rates. Moreover, community detection algorithms are not designed for adversarial cases and most of them required global knowledge which makes it harder to decentralized. We suggest that community detection approach should be considered only if community information is valuable to your system.

PageRank is another technique that could hep in finding sybils. The ranking of a page depends on the quality and quantity its referenced links reside in other pages. Instead of ranking pages, we can rank nodes in G and nodes with lower ranking have higher possibility of being sybils. The original PageRank algorithm has been proven to be vulnerable to Spam Farm attack, but there are many enhancements to the PageRank algorithm we can apply to bound the effect of Spam Farm attack. SybilRank[24] is one of the first sybil detection approaches that utilize PageRank as the underlying method but it is susceptible to attacks that used to target the PageRank algorithm. By the time of this writing, SybilRank is the only SNSD approach that has been deployed in a real world setting. After deployment, SybilRank has found 200k accounts in Tuenti, a Spanish Online Social Network. Among the 200k accounts, more than 90% of them are actually fake accounts. SybilRank claims to achieve the same end guarantee as SybilLimit but much better run time performance when applied to a large social network graph. Integro [25] is an enhancement on SybilRank by first predict victims and use the predicted victims information in the PageRank process. Integrao is also deployed in Tuenti and achieve an order of magnitude higher precision when compare with SybilRank. Both SybilRank and Integrao are designed to detect sybils in OSN, it's likely that they will also work in other domains but no prove has been given.

VI. Domain Specific Sybil Defense

Domain Specific Sybil Defense (DSSD) try to exploit the domain specific features to help prevent against sybil attacks. Since we will not be able to list all the DSSD approaches we will like to present a few that are more representative than others in the following few subsections. Some DSSD approaches have better performance than general approaches and require less implementation and maintenance overhead.

A. Recommendation Systems

DSybil[26] is an innovative approach trying to reduce the influence of sybils in recommendation system. DSybil has provable guarantees that are optimal under its setting. The evaluation result shows that DSybil would continue to provide high quality recommendation even under a potential sybil attack launched from a million node botnet. DSybil mainly exploit two domain specific features of recommendations systems. The first one is the heavy-tail distribution of the typical voting behavior of the honest identities. The second feature is trust. A recommender system should trust a person more (less), if he/she voted for good (bad) objects and objects

with more votes should probably be recommended over objects with fewer votes.

B. Ad hoc Networks

In wireless ad hoc networks, a group of sybils are usually sharing the same device and they can be detected through monitoring signals? features or the moving patterns of co-existing identities. SybilCast is a novel protocol proposed in [27] that can limit the number of fake identities in centralized multichannel wireless networks. SybilCast can ensure that each honest user gets at least a constant fraction of their fair share of the bandwidth and complete his or her data download in asymptotically optimal time.

C. Wireless Sensor Netowrk

In wireless sensor netowrk, Demirbas et al. proposed an sybil attack counter measurement by using received signal strength indicator (RSSI). The algorithm proposed in [28] claims to be light weight because it only require the colloboration of one other node apart from the receiver and accurate because it detects sybil attack cases with 100% completeness and only a few percent false positives. [28]

D. Online Social Network

Yu et al.[29] leverage some behavioral attributes of fake accounts to detect them in Renren, the biggest Online Social Network in China. The identified behavioral attributes include invitation frequency, outgoing requests accepted, incoming requests accepted and clustering coefficient. Invitation frequency is the number of friend requests a user has sent out during a fixed amount of time. Sybil accounts are believed to have sent out more friend requests than honest users. Moreover, sybil accounts tend to have a very low outgoing requests acceptance rate, because honest users are less likely to accept friend requests from a stranger. What's more, sybil users tend to accept most of the incoming friend requests while honest users only accept a portion of all incoming friend requests. Last but not least, clustering coefficient (cc) is a metric used to measures the mutual connectivity of a user's friends. Sybil accounts are likely to befriend users who don't know each other and as a result have a low cc value. All the above features are feed into a support vector machine and the result is promising. The deployment of this technique by Renren found more than 100,000 fake accounts in less than 6 months.

VII. Conclusion

This survey has showed a full picture of the effects of Sybil attacks in many different domains. We described how those attacks are being carried out. If you are maintaining a system that could potentially be the target of Sybil attacks, you should seriously consider adding protection in your system.

As a guide for Sybil defense, this survey presents two type of defense mechanisms. The general defense and the domain specific defense. The general defense consists of the traditional defenses and the newly emerged social network based defenses.

There are two types of general defense approaches. The traditional sybil defense approach (TSD) and the social network based sybil defense approach (SNSD)

TSD includes trusted authority authentication, recurring cost, resource testing and human assisted defense. Maybe you might think that traditional approach has been outdated but that it's not true, they are actually the most popular approaches in terms of deployment rate. A lot of systems today has no sybil tolerance so the only option they have is to go with the central authority certification appraoch. Resource testing especially IP address testing is also popular because it is very easy to implement and enforce in real world but this approach provides an extremely week bound on the number of sybils an attacker can create. In the era of Cloud, even casual attackers can instantly acquire large amount of resources in a small cost so resource testing should be your last option. System administrator should compare the cost of creating sybils and the benefit the attacker would acquire carefully before they decide to go with resource testing. Recurring cost is very similar to resource testing except that the recurring cost can be turing test that machine would not be able to solve efficiently. While turing test could rule out machines and raise the Sybil attacks cost on those system, one should not be too optimistic about this approach either mainly because the advent of crowdsourcing. What's more, the last traditional approach human assisted defense tries to offload the sybil detection work to some of the systems' users. Honest users in this case has mechanisms to monitor their neighbors and report them to a human judge and the judge would investigate if the reported node is a sybil or not. This approach can be very expensive due to human involvement in the process.

If none of the TSDs do not appeal to you, there is the more advance approach that could keep your system open and provide a tighter bound on the negative effects of Sybil attacks. SNSD has been the most popular automated sybil detection approach by exploiting the underlying social network structure in your systems. The effectiveness of this approach depends on how much your system follow the social network structure. Is it easy for honest node to connect with sybil nodes in your system? If yes, this approach is not for your system. Also, this approach does not stop sybils from joining the system but detect them or reduce their effects after their joining. Most of the SNSD approaches have not been deployed into the wild mainly because the implementation, runtime and maintenance overhead is too high for them. SybilRank and Integro are the only two approaches that have been successfully deployed and have proven to be effective in automatically detecting sybils in OSN. If your system does not have a social network structure but would want to use this approach, there is a workaround to this. You can authenticate your systems' users through any online social networks and obtain their social connections through their online social network accounts.

Before going for a general approach you own yourself to investigate the possibility of a better domain specific sybil defense (DSSD) approach in your domain. Sometimes you could combine general approaches with DSSDs to further

enforce your system resilience to sybil attacks.

REFERENCES

[1] Wikipedia, "Sybil (book)," 1973. [Online; accessed 02-May-2016].

[2] L. Wang and J. Kangasharju, "Real-world sybil attacks in bittorrent mainline dht," in *Global Communications Conference (GLOBECOM), 2012 IEEE*, pp. 826–832, Dec 2012.

[3] P. Maymounkov and D. Mazières, "Kademlia: A peer-to-peer information system based on the xor metric," in *Revised Papers from the First International Workshop on Peer-to-Peer Systems*, IPTPS '01, (London, UK, UK), pp. 53–65, Springer-Verlag, 2002.

[4] S. Wolchok, O. S. Hofmann, N. Heninger, E. W. Felten, J. A. Halderman, C. J. Rossbach, B. Waters, and E. Witchel, "Defeating vanish with low-cost sybil attacks against large dhts," 2009.

[5] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: Attacks and countermeasures," 2003.

[6] N. Tran, B. Min, J. Li, and L. Subramanian, "Sybil-resilient online content voting," in *Proceedings of the 6th USENIX Symposium on Networked Systems Design and Implementation*, NSDI'09, (Berkeley, CA, USA), pp. 15–28, USENIX Association, 2009.

[7] G. Swamynathan, K. C. Almeroth, and B. Y. Zhao, "The design of a reliable reputation system," vol. 10, pp. 239–270, Dec. 2010.

[8] Q. Lian, Z. Zhang, M. Yang, B. Y. Zhao, Y. Dai, and X. Li, "An empirical study of collusion behavior in the maze p2p file-sharing system," in *In ICDCS*, 2007.

[9] D. Quercia and S. Hailes, "Sybil attacks against mobile users: Friends and foes to the rescue.," in *INFOCOM*, pp. 336–340, IEEE, 2010.

[10] J. Newsome, E. Shi, D. Song, and A. Perrig, "The sybil attack in sensor networks: analysis defenses," in *Information Processing in Sensor Networks, 2004. IPSN 2004. Third International Symposium on*, pp. 259–268, April 2004.

[11] M. Castro, P. Druschel, A. Ganesh, A. Rowstron, and D. S. Wallach, "Secure routing for structured peer-to-peer overlay networks," *SIGOPS Oper. Syst. Rev.*, vol. 36, pp. 299–314, Dec. 2002.

[12] A. Adya, W. J. Bolosky, M. Castro, G. Cermak, R. Chaiken, J. R. Douceur, J. Howell, J. R. Lorch, M. Theimer, and R. P. Wattenhofer, "Farsite: Federated, available, and reliable storage for an incompletely trusted environment," *SIGOPS Oper. Syst. Rev.*, vol. 36, pp. 1–14, Dec. 2002.

[13] N. Borisov, "Computational puzzles as sybil defenses," in *Proceedings of the Sixth IEEE International Conference on Peer-to-Peer Computing*, P2P '06, (Washington, DC, USA), pp. 171–176, IEEE Computer Society, 2006.

[14] L. V. Ahn, M. Blum, N. J. Hopper, and J. Langford, "Captcha: Using hard ai problems for security," in *Proceedings of the 22Nd International Conference on Theory and Applications of Cryptographic Techniques*, EUROCRYPT'03, (Berlin, Heidelberg), pp. 294–311, Springer-Verlag, 2003.

[15] H. Yu, M. Kaminsky, P. B. Gibbons, and A. D. Flaxman, "Sybilguard: Defending against sybil attacks via social networks," *IEEE/ACM Transactions on Networking*, vol. 16, pp. 576–589, June 2008.

[16] H. Yu, "Sybil defenses via social networks: A tutorial and survey," *SIGACT News*, vol. 42, pp. 80–101, Oct. 2011.

[17] 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*, pp. 3–17, May 2008.

[18] G. Danezis and P. Mittal, "Sybilinfer: Detecting sybil nodes using social networks," Tech. Rep. MSR-TR-2009-6, January 2009.

[19] N. Tran, J. Li, L. Subramanian, and S. S. M. Chow, "Optimal sybil-resilient node admission control," in *INFOCOM, 2011 Proceedings IEEE*, pp. 3218–3226, April 2011.

[20] N. Tran, B. Min, J. Li, and L. Subramanian, "Sybil-resilient online content voting," in *Proceedings of the 6th USENIX Symposium on Networked Systems Design and Implementation*, NSDI'09, (Berkeley, CA, USA), pp. 15–28, USENIX Association, 2009.

[21] W. Wei, F. Xu, C. C. Tan, and Q. Li, "Sybildefender: A defense mechanism for sybil attacks in large social networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, pp. 2492–2502, Dec 2013.

[22] 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*, pp. 1034–1042, April 2013.

[23] J. Xue, Z. Yang, X. Yang, X. Wang, L. Chen, and Y. Dai, "Votetrust: Leveraging friend invitation graph to defend against social network sybils.," in *INFOCOM*, pp. 2400–2408, IEEE, 2013.

[24] Q. Cao, M. Sirivianos, X. Yang, and T. Pregueiro, "Aiding the detection of fake accounts in large scale social online services," in *Presented as part of the 9th USENIX Symposium on Networked Systems Design and Implementation (NSDI 12)*, (San Jose, CA), pp. 197–210, USENIX, 2012.

[25] Y. Boshmaf, D. Logothetis, G. Siganos, J. Leria, J. Lorenzo, M. Ripeanu, and K. Beznosov, "Integro: Leveraging victim prediction for robust fake account detection in osns," Mar 2015.

[26] H. Yu, C. Shi, M. Kaminsky, P. B. Gibbons, and F. Xiao, "Dsybil: Optimal sybil-resistance for recommendation systems," in *2009 30th IEEE Symposium on Security and Privacy*, pp. 283–298, May 2009.

[27] C. Zheng and D. S. Gilbert, "Thwarting sybil attacks and malicious disruption in wireless networks, http://www.comp.nus.edu.sg."

[28] M. Demirbas and Y. Song, "An rssi-based scheme for sybil attack detection in wireless sensor networks," in *World of Wireless, Mobile and Multimedia Networks, 2006. WoWMoM 2006. International Symposium on a*, pp. 5 pp.–570, 2006.

[29] Z. Yang, C. Wilson, X. Wang, T. Gao, B. Y. Zhao, and Y. Dai, "Uncovering social network sybils in the wild," in *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference*, IMC '11, (New York, NY, USA), pp. 259–268, ACM, 2011.