Stalk Me if You Can – The Anatomy of Sybil Attacks in Opportunistic Networks

Sascha Trifunovic, Andreea Hossmann-Picu
Communication Systems Group
ETH Zurich, Switzerland
{trifunovic, picu}@tik.ee.ethz.ch

ABSTRACT

Opportunistic Networks are envisioned to complement infrastructure-based communication in overloaded cellular settings, in remote areas, during and immediately after large scale disasters. On account of their highly distributed and dynamic nature, as well as of their dependence on the honest cooperation of nodes, Opportunistic Networks are particularly vulnerable to sybil attacks. In a sybil attack, a node assumes multiple identities and attempts to form many links to the rest of the network, with the aim of gaining access to resources, influencing the network, circumventing detection of misbehavior ("spread the blame"), etc.

Sybil attacks have been studied extensively in the context of distributed systems and online social networks. However, the Opportunistic Networking setting brings new challenges, specific to the network conditions: forming links may require significant resources from the attacker (e.g. time, speed, multiple devices, etc), and each link is ephemeral. In this paper, we study the types and effectiveness of sybil attacks that are possible in Opportunistic Networks, under various resource constraints on the attacker. We evaluate each attack based on the influence the attacker can gain through it. We find that sybil attacks, even with relatively unconstrained resources, are much harder to implement in the Opportunistic Networking setting, due to the link establishment mechanisms using mobility. We believe this is a very important first step towards understanding and defending against sybil attacks in such networks.

1. INTRODUCTION

In Opportunistic Networks (OppNets), mobile phone users may cooperate to complement existing wireless communication services (cellular, Wi-Fi) and to enable communication in case of failure or lack of infrastructure (disaster, censorship, remote areas). Wireless peers communicate when they are in proximity (in *contact*), forming an impromptu network, whose connectivity graph is highly dynamic and only partly connected. Using redundancy (e.g., coding, replication) and smart mobility prediction schemes, data can be transported over a sequence of such contacts, despite the lack of end-to-end paths.

The feasibility of communication over an OppNet highly depends on the *honest* cooperation of the nodes, by contributing resource to run the network instead of only receiving the communication service passively from the system. The issues of benign selfishness (non-cooperation) and of motivating users to cooperate have already received a fair amount of attention from researchers: there exist many studies on the effects of selfishness [1, 2], as well as a number of incentive systems to ensure participation in the network [3, 4]. However, the possibility of malicious users has hardly been considered so far.

One of the most powerful and most versatile ways to disrupt the incentive and/or security mechanisms of cooperative systems is the *sybil attack*, in which the adversary creates many fake identities (sybils) and uses them to disrupt the system's normal operation. Sybils can be used to obtain a large share of resources from resource allocation algorithms, to bias recommendation or voting systems, to intercept seemingly disjoint routing paths, to circumvent the detection of misbehavior by "spreading the blame", etc.

Sybil attacks have been studied extensively in the context of distributed systems and online social networks [5, 6, 7, 8, 9, 10]. However, the OppNet setting brings both new possibilities as well as new challenges, specific to the network conditions. On the one hand, the highly distributed and dynamic nature of OppNets makes them easy targets for sybil attacks (as it is practically impossible to rely on a single centralized authority to certify legitimate users). On the other hand, a sybil attack requires the establishment of a number of connections (known as "attack edges") between the adversary's identities and the rest of the network; in OppNets, where links are by-products of node mobility, forming intentional links may require significant effort, as the two connected peers must be physically co-located for a significant amount of time.

Our goal in this paper is to explore and dissect the *procedure of carrying out sybil attacks in OppNets*. We identify the two main elements of a sybil attack (link creation, ID fabrication) and quantify the amount of effort and resources an adversary needs to spend on each of them. We also review the various tools at an adversary's disposal in an OppNet (interconnecting the fake IDs, linking via lying etc) and their impact on the strength of the attack. We show that, despite the mostly disconnected state of an OppNet, sybil attacks are much harder to implement here, even under generous resource allowances. This is mainly due to the link formation mechanisms via mobility, which demand continuous effort from an attacker.

The rest of the paper is organized as follows: In Sec. 2, we present the state of the art in sybil defense mechanisms and discuss how and whether they apply to OppNets. Next, in Sec. 3, we review the possible variations on how to implement each of the two main elements (link creation, ID fabrication) of a sybil attack in the OppNet context. Then, we show our methodology for assessing and comparing the effect of each of these variations on the attack strength in Sec. 4. In Sec. 5, we use two real traces and a mobility model to quantify the strength of sybil attacks on OppNets, depending on used resources and on how the attack is implemented. Finally, we briefly discuss our results and conclude in Sec. 6 and 7.

2. BACKGROUND AND RELATED WORK

Carrying out a (simple) sybil attack is generally quite straight-

CHANTS'14, September 07, 2014, Maui, HI, USA

http://dx.doi.org/10.1145/2645672.2645673.

forward in traditional distributed systems as opposed to OppNets. Therefore, research on the sybil attack focuses on devising effective detection/defense mechanisms, with little attention on how to best implement the attack. The goal of sybil detection is to accurately identify sybil identities (i.e. accept all legitimate identities, but no counterfeit ones). For our purpose of dissecting and comparing sybil attacks and their effectiveness in OppNets, algorithms for sybil detection can be very useful in assessing the cost-benefit tradeoff of each element of an OppNet sybil attack.

The state of the art in sybil detection consists in leveraging the social network underlying the distributed system under consideration. Assuming the sybil identities can create only a limited number of connections to honest nodes (i.e. attack edges), the resulting graph will then consist of two regions, loosely connected to each other: the honest nodes and the sybils. Depending on the structure of the honest region, defense mechanisms do one of the following:

i) Identify and exclude all sybils (universal sybil defense). If the honest region of the graph is relatively well, but flatly connected internally and fast mixing, then the sybils are easily detectable via community detection or similar algorithms. Many sybil defense solutions are based on this idea [5, 6, 7, 8, 9, 10].

ii) Enable honest nodes to white-list a set of nodes of any given size, ranked according to their trustworthiness. Since it has been shown that, in practice, the honest region often has internal structure (e.g. "communities" of tightly-knit nodes, relatively loosely connected with one another), the goal of sybil defense has shifted accordingly [11].

OppNets have been shown to have a very strong social component already at the network layer [12] (due to the nodes/phones being near perfect proxies for the human users), therefore the above-mentioned sybil defense solutions should be directly applicable to the OppNet setting. However, this requires the representation of the network as a static graph. In OppNets, communication occurs between two wireless peers whenever their mobility brings them within radio range of each other. Deriving a graph from such contacts is not straightforward: the contacts are shortlived and the timing information must be stored for each edge. This makes the resulting time-varying graph very cumbersome and unintuitive. A more practical and widely accepted representation can be obtained by aggregating contacts into a (static) weighted graph, with weights derived from contact statistics (e.g. frequency, duration, age of last contact, or combinations thereof). It is *this* graph that was shown to also reflect to high extent the social relationships among the users [13], and on which existing sybil defense schemes are readily applicable. To assess the effectiveness of OppNet sybil attacks, we will thus apply the latest sybil defense algorithm [11], from the second category to the weighted OppNet graph. We provide a more detailed description of how this algorithm operates in Sec. 4.

Some literature on sybil detection exists also for network environments similar to OppNets. For example, adaptations of the above algorithms have been proposed for small mobile networks with very sparse social ties, as resulting from secure pairing [14]. However, the feasibility of such a pairing-based network is questionable, as it requires incentives for the users to actually pair their devices. In the context of mobile ad hoc networks (MANETs), sybil detection schemes are mostly based on peculiarities of this environment (some of which are also exhibited by OppNets). However, such solutions usually require specialized hardware [15, 16, 17] or are based on strong, unrealistic assumptions [18].

3. INGREDIENTS OF A SYBIL ATTACK

In order to perform a sybil attack, an adversary must do two types of operations: create links to honest nodes and create fake identities. The OppNet environment raises issues with both operations. In the following, we discuss in detail each of these two operations and the options an adversary has for implementing them.

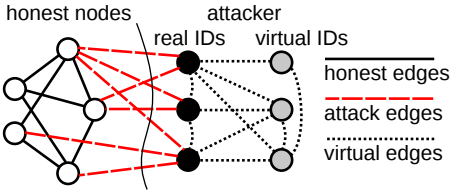


Figure 1: Real vs. Virtual IDs in the contact graph

3.1 Fabricating Node Identities

The core of a sybil attack consists of creating a number of identities, all controlled by the same entity: the adversary. To the rest of the network, each of these identities looks like just another node. In OppNets, there are two options for fabricating identities:

Real IDs: An attacker with a single physical device can create multiple real IDs, each of which will broadcast its existence to the rest of the nodes, either at different times or simultaneously. Such sybils are harder to create, as they require more resources (e.g. battery power) as well as specialized skills (e.g. knowledge of the broadcast protocol). However, real ID sybils are also much more powerful, as they are more likely to be perceived as honest nodes: firstly, because they do not necessarily need to interact among each other, and secondly, on account of their direct interactions with honest nodes.

Virtual IDs: An easier way to create sybils is to introduce virtual IDs. In contrast to a real node who broadcasts her ID to be discovered, a virtual ID only interacts with the attacker's real IDs: it is thus only an imaginary neighbor, rater of content, producer of spam, etc. Virtual IDs do not interact with other nodes, as they do not broadcast their existence to the world. Both real and virtual IDs are shown in Fig. 1.

In our practical evaluation of sybil attacks in Sec. 5, we will only show the effect of the number of real identities on the strength of the attack. Our preliminary experiments with virtual sybils show that they are easily weeded out by social sybil defences; we thus omit them here and defer a deeper look to future work.

3.2 Creating Links in the OppNet Graph

As pointed out above, representing an OppNet with a static weighted graph opens the door for applying state of the art sybil defense schemes to OppNets. However, this also means that the choice of edge weights becomes crucial. Depending on how weights are derived from contact statistics, the cost to an attacker of establishing a link in the OppNet graph can vary significantly.

Like the underlying contacts, legitimate links in the OppNet graph come "for free", since they are simple by-products of node mobility. In contrast, to establish a link to an honest node, an adversary must move *at least once* in physical proximity of the targeted node; if the link weights are chosen wisely, the adversary may even have to constantly follow a target node to establish and maintain a link.

Below we detail the different options for deriving link weights from contacts and discuss the cost they impose on a sybil adversary

Contact Frequency. A frequently occuring contact often has a higher probability of occuring again soon, reflecting a strong social link, which is what the OppNet graph attempts to capture. However, an adversary who has managed to get in range of an honest node may easily enhance the link's frequency, by constantly connecting and disconnecting.

Age of Last Contact. In many scenarios, it is reasonable to assume that an older contact has less predictive power than more recent ones. In this case, a sybil adversary must regularly go "visit" its targeted peers in order to refresh and maintain the links.

Total Contact Duration. The total contact duration, i.e., the total time a peer has been detected in proximity is a link weight which also reflects a strong social relationship between the

peers. To create strong links with this type of weight, a sybil adversary must constantly follow its target.

Frequency-Duration Combined. Link weights can also be obtained by some combination of contact frequency and duration (e.g., linear, principal component analysis etc). Depending on how the two are combined, this kind of weight might work to the advantage of the adversary, as in the case of pure frequency weights.

All in all, most reasonable types of edge weights come at a high cost to a sybil adversary: the attacker must not only establish links to honest nodes by making an initial contact, but she must also maintain these links by continuously making recurrent contacts. An adversary can do this, for example, by regularly following specific nodes or by hiding at popular locations.

This is in very stark contrast to the link creation process in the distributed systems in which the sybil attack is usually studied, such as online social networks. In that case, an adversary can initiate a virtually unlimited number of links (i.e., friendship requests) with almost no effort. In addition, once some of these links are accepted, no further maintainance is required.

Naturally, a sybil adversary also has the option of *lying about past contacts with third nodes*. Due to the distributed nature of OppNets, no single node has a global view of the contact graph. Nodes need to approximate the contact graph by exchanging information on the neighbors they have seen and for how long upon every encounter. An attacker can obviously report any encounter it sees fit. Lying about encounters not only allows for the introduction of virtual sybil IDs, but can also be done for other real nodes. This allows the attacker to manipulate all her outgoing edges in the contact graph, except the one to the node it is reporting to.

4. ASSESSING THE EFFECTIVENESS OF SYBIL ATTACKS ON OPPNETS

Having identified the two main elements of a sybil attack and their variations in the OppNet context, we now need a methodology for assessing and comparing these variations. To do so, we use the latest sybil defense algorithm [11], which enables honest nodes to white-list a set of nodes of any given size, ranked according to their trustworthiness. Then, based on this algorithm, we define some metrics to measure the strength and effectiveness of a sybil attack. We use these metrics in our experiments in Sec. 5.

4.1 Social Sybil Defense on the OppNet Graph

The state of the art sybil defense algorithm, referred to as ACL (for its creators: Andersen, Chung and Lang), was first proposed in [19] as a graph partitioning algorithm; Alvisi et al. [11] analyzed in great detail its application to sybil defense and derived theoretical guarantees on its performance.

The ACL algorithm is based on social network theory: it assumes the network is formed of one or several tightly-knit communities, which are only loosely connected to one another. Since relationships are strong inside a community, it is reasonable to assume that those nodes know and trust one another. Thus, for a given honest node, its community neighbors are very unlikely to be sybils, while the nodes outside its local community are more questionable.

The rough idea of ACL is to associate a score with each node and to identify as part of a community all nodes whose score exceeds a certain threshold. To determine the score of a node, ACL originates from the honest user many truncated random walks, whose lengths are geometrically distributed. Then, a node's score is given by the frequency with which it is visited, normalized by its degree. By interpreting the score of a node i as a measure of the trust that the seed node puts in i, ACL can be employed for sybil detection.

Since the length of the random walks is geometrically distributed, long walks are rare and short walks in the neighborhood of the honest node are common. In this way, the nodes in the community of the seed node are visited more frequently and thus receive a higher score. Sorting according to the scores in descending order produces a (white-)list of nodes ranked from the most trustworthy to the least trustworthy, from the seed node's viewpoint.

The ACL algorithm is designed for unweighted, undirected graphs. However, the OppNet graph has weighted edges, where the weight of an edge is derived from contact statistics of the corresponding node pair. Moreover, if the adversary uses the option of lying about past contacts with third nodes (as discussed in Sec. 3.2), the graph is also directed: indeed, a pair of one sybil and one honest node will report different weights for the edge between them, effectively resulting in two directed edges, with each node controlling the weight of its outgoing edge. Thus, to use ACL on the OppNet graph, we must make a small adjustment as follows. At each step of a random walk, an edge is traversed with probability proportional to its weight. That is, if the walker is currently at node i, the probability of moving to neighbor j is: $\frac{w_{ij}}{d_{\text{out}}(i)}$, where w_{ij} is the weight of edge (i, j) and $d_{\text{out}}(i) = \sum_k w_{ik}$ is the out-degree of node i.

4.2 Measuring Sybil Effectiveness in OppNets

In order to evaluate the effectiveness of a sybil attack, every node i calculates rankings of the other nodes in the OppNet, using the adapted ACL algorithm described above. That is, every node i assigns a rank, $t_i(j)$, to every other node j, where a smaller rank is better and 0 is the top rank. The better the sybil nodes are ranked, the more successful the attack.

To quantify the success of a sybil attack, we use the following three metrics. Since the rankings are intimately related to community structure, community size is an integral part of the first two metrics. The third metric captures the impact across communities.

The normalized rank measures how well an attacker is integrated in a community. It is a value between 0 and 1, where 1 means perfectly integrated and 0 means not at all. For a given honest node i and a given sybil k, the normalized rank is: $R = \max\left(1 - \frac{t_i(k)}{|C_i|}, 0\right)$. To obtain the normalized rank assigned by node i to the adversary, we take the best value across all identities. Finally, the normalized rank assigned by a community C_i to an adversary is obtained by averaging over all honest members.

The influence is the percentage of sybils in the top $|C_i|$ spots of the ranking of an honest node i. It shows the potential of the adversary to outweigh honest nodes in a given community.

The total influence is the total number of influential sybils, that is, the sum of all per-community influences, weighted by each community's size.

In the next section, we use these metrics to assess the effectiveness of a sybil attack, in function of the resources consumed by the adversary to perform the attack.

5. THE SYBIL ATTACKER'S COST-BENEFIT TRADEOFFS IN OPPNETS

To evaluate the cost-benefit tradeoffs of sybil attacks in OppNets we use two real world traces (WiFi access point associations from the campuses of Dartmouth [20] and ETH Zurich) and one synthetic trace generated by the TVCM mobility model [21]. The main properties of the traces are summarized in Table 1.

For the two WiFi traces, only users who have an AP association for at least five days a week are considered. In addition, short disconnections (less than 60 seconds) attributed to interference and the well known ping-pong effect (where devices jump back and forth between different APs in less than 60 seconds) are filtered out. Two nodes are considered in contact while associated to the same AP.

For the TVCM trace, we place 505 nodes in an area of 900 × 900 m, divided into a 9 × 9 grid of cells. Each node is assigned

	TVCM	ETH	DART
# Nodes	505	294	1045
Time Period	72 hours	14.6 weeks	16.9 weeks
Type	Coordinates	AP assoc.	AP assoc.
# Contacts Total	3'822'531	101'805	5'177'521
# Contacts/Node	7'569	346	4'954
# Communities	34	26	65
Modularity	0.89	0.77	0.61

Table 1: Mobility traces used in evaluation. (Modularity measures the strength of the community structure [23].)

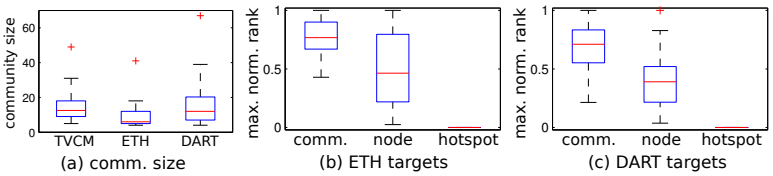


Figure 2

one of 34 home cells, based on a skewed distribution, resulting in a skewed community size distribution. Each node moves within the home cell for 80% of the time. The nodes visit one of 10 hotspot cells for 10% of the time, and the remaining 10% is spent roaming the whole area. We generate one trace with a timespan of 72 hours. Nodes are considered in contact if they are less than 30 m apart.

We build the OppNet graphs of our three traces, using total contact duration as the edge weight. Since community size is an important component of our metrics from Sec. 4, we extract the community structure from the OppNet graphs, using the Louvain community detection algorithm [22]. The community size distributions are shown in Fig. 2(a) for all three traces.

In the following we discuss and evaluate the different tradeoffs facing a sybil adversary in an OppNet. We exclusively use real sybil IDs as virtual IDs are hardly useful (see Sec. 3.1).

5.1 Finding a Target: Communities vs. Users vs. Hotspots

The first step of a sybil attack is finding a target. An adversary has several options: (i) position oneself at a hotspot, that is often frequented by many nodes, (ii) follow a random node, or (iii) follow a community, by following that community's highest degree node (this requires knowledge of the degrees). Fig. 2(b) and 2(c) show the benefit of these attacks in terms of the maximal normalized rank an attacker receives in a community.

For the hotspot approach, we positioned the attacker at the most frequented AP in the ETH and DART traces, and in the center of a hotspot cell in the TVCM trace. While this is the most convenient approach for an attacker (as she does not have to move), Fig. 2(b) and 2(c) clearly show that it is also useless. The adversary has numerous contacts with many different nodes, but their cumulative duration is too short to have an impact.

Following a random target node puts the attacker very high in that specific node's rankings. However, the attacker's rank by other nodes depends on whether the target is well connected in its community or not. This results in a highly variable distribution of the normalized rank, as seen in Fig. 2(b) and 2(c). If the attacker only wants to influence the specific target, this approach is fine. However, if she wants to influence as many nodes as possible, the attacker must specifically target a whole community, by following the highest degree node of that community. Performing such a directed attack, the adversary must identify the most central node of a community. While this is certainly not an easy task, it does produce the highest normalized rank within the community, as shown in Fig. 2(b) and 2(c).

Conclusion: An attacker must make an effort to gain trust, positioning itself at a hotspot although easy is futile. To achieve the greatest impact an attacker should follow the most central node of a target community.

5.2 Creating Links to Honest Nodes

As we have seen in Sec. 3, one of the main costs of a sybil attack is the time the attacker has to spend following its target. This is necessary, since it is the only way to create a link to the target. Here we show the relationship between amount of following time and gained influence. Fig. 3 clearly demonstrated that a sybil attack on an OppNet is at least one full time job. While in the more homogeneus scenario of the TVCM trace, 8 continuous hours are enough to successfully infiltrate a community, in the ETH and DART traces there is a day-night cycle, which affects the attack's efficiency. Since different nodes are active at different times of the day, following less than 24 hours a day has a notable effect (not all nodes in the community are active at the same time).

Conclusion: Efficient sybil infiltration is a fulltime stalking job.

5.3 Following Multiple Targets

The community structure of OppNet nodes confines the reach of a sybil attacker to the community of the target. However, an attacker may want to influence users beyond community borders. To do so, she must follow multiple targets from different communities. This not only requires more knowledge about the existing communities and their central nodes, but also requires an effort to spatially switch from following one target to another. Our simulations take the switching distance into account, as the attacker needs some time to move from the current target's AP to the next target's AP.

When following multiple communities, the influence in each community will be reduced as shown in Fig. 4. Note that this influence reduction is actually more significant than when varying the following time, as shown in Fig. 3. The perceptive reader will notice that the time spent in each community is, in fact, the same as in the variable following time scenario.

Having a reduced influence in each community is not necessarily a bad thing, as more communities are reached. To account for this effect, we show the total number of influential sybils in Fig. 5. For the TVCM model the reduction in influence per community is actually more severe than the benefit of influencing multiple communities. For the ETH and DART traces, following multiple targets is beneficial, peaking at 10 and 3 targets respectively. For the ETH trace following more than 10 targets is difficult to assess as the trace is too small.

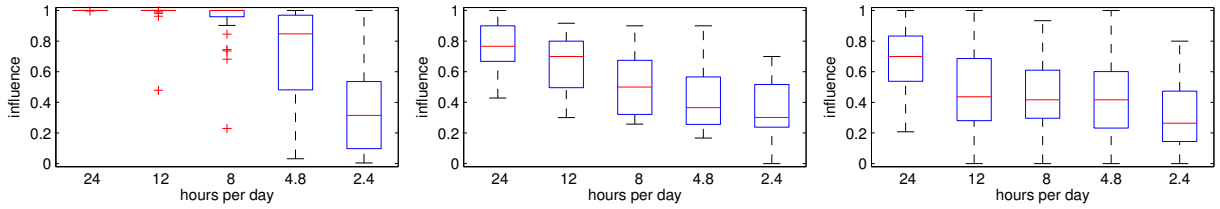
Conclusion: If an attacker wants to extend her influence beyond the border of a community, she must follow multiple targets. However, as the influence per followed community decreases, the total influence is still bounded.

5.4 How Many Sybils Are Enough?

Each real sybil ID needs to broadcast its existence and interacts with honest nodes. This naturally incurrs a cost in terms of battery consumption (depending on the communication protocol there is also a natural limit), so a natural question that arises is how many sybil identities should an attacker create. As shown in Fig. 6 the influence of an attacker increases as the number of sybils increases.

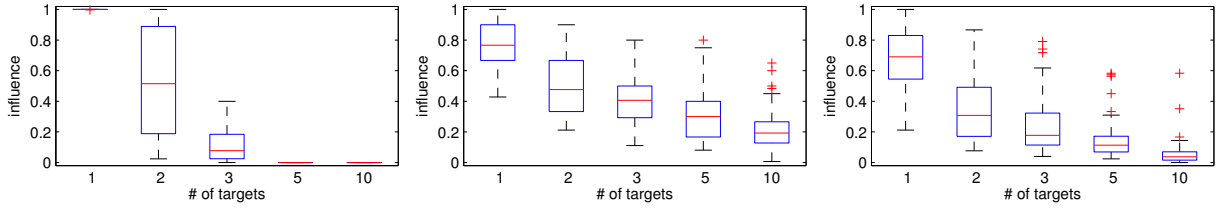
If the number of created sybils is too low, the attacker's influence is diminishing as she is outnumbered by the honest nodes in the community. However, when enough sybils are created to outnumber the whole community (for our traces this happens between 20 and 40 sybils), the influence is only limited by the attackers rank. Interestingly, increasing the number of sybils also increases the rank and thus the influence of the attacker. The reason why the rank would increase with an increasing amount of sybils is not nessarily intuitive as a particular sybil will not be visited more often by a random walk just because there are more of her kind. However, an increasing number of sybils increases the degree of honest nodes resulting in a more severe normalization.

While for the TVCM model and the ETH traces a good influence



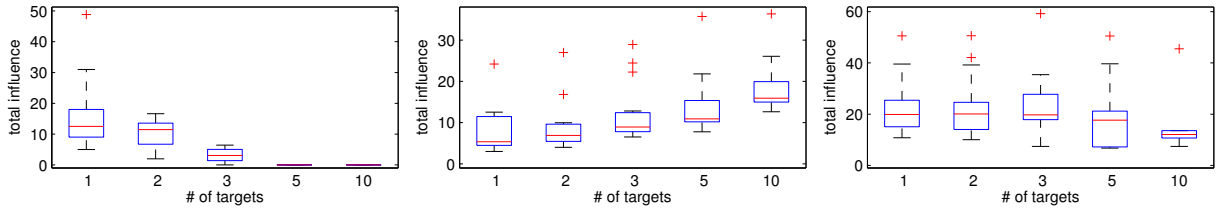
(a) TVCM (b) ETH (c) DART

Figure 3: Impact of the stalking time in the % of infiltrated by Sybils. Real Sybils, not interconnected, following 1 target node.



(a) TVCM (b) ETH (c) DART

Figure 4: Impact of following multiple targets on the % of target communities infiltrated by Sybils. 80 real Sybils, not interconnected.



(a) TVCM (b) ETH (c) DART

Figure 5: Impact of following multiple targets on the total number of influential Sybils. 80 real Sybils, not interconnected.

can be achieved with a reasonable amount of Sybils, in the DART traces the impact increases slower. This is because the communities in the DART traces are less tightly knit and it is thus hard for an attacker to cover the whole community.

Figure 6 also shows the rank and the influence in the second most infiltrated community. We can see that for the TVCM model this is practically 0, as the community structure is very strong, but the DART traces have weaker community structure and there is thus a slight influence in the second community. There is practically no influence in a third community. This actually shows the benefit of communities: while one community can be infiltrated, the others remain unharmed.

Conclusion: The more Sybils an attacker can create, the better, while they come at a cost of energy consumption. Also, with a stronger community structure the attack is more focused on the specific community.

5.5 To Build or Not to Build a Sybil Network

When an attacker creates many real sybil identities (here we use 80), these will all be connected to the target and its community in the same way. However, since all sybils are on the same device, the attacker may decide on how to interconnect them. There is a natural tradeoff when considering the interconnection of sybils. On the one hand, a random walk that enters the sybil region may spend more time among sybils if they are interconnected. On the other hand, having interconnections increases the degree of each sybil identity and decreases its trust via the degree normalization. As shown in Fig. 7, the second effect clearly dominates in all traces.

Conclusion: An attacker has a choice on how it apparently interconnects its sybils and the best strategy is to pretend they never see each other.

6. DISCUSSION AND FUTURE WORK

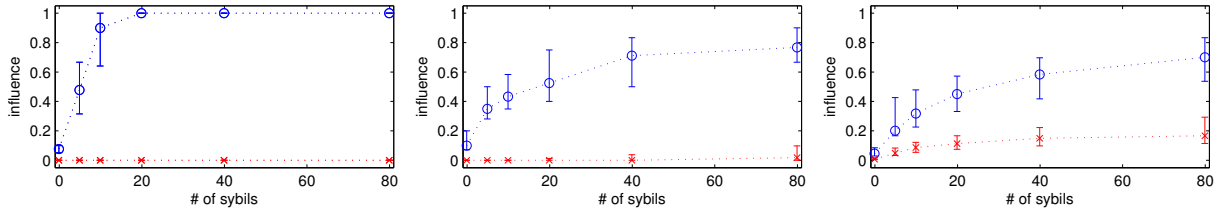
Here we discuss limitations and possible extensions to our work.

The Benefit of Lying: As explained in Sec. 3.2, an attacker might lie about its encounters and change the perceived contact graph. While we analyze the effect of lying about sybil interconnections in Sec. 5, lying about encounters with honest nodes is a complex topic. Our preliminary experiments and calculations show that it often has a counterintuitive result as the edge weights not only influence the random walk but also the normalization in the end and always in opposing ways. Additionally, lying needs to be done in a smart way in order not to be exposed. We leave the thorough analysis of this topic to future work.

Colluding Attacker: Multiple attackers may of course collude to reach more communities and increase their influence. We did not study this attack as its effect is straightforward. When attackers collude they may actually also lie and pretent to be in contact. This is however a bad idea as it increases the attackers' degree by which its trust will be normalized.

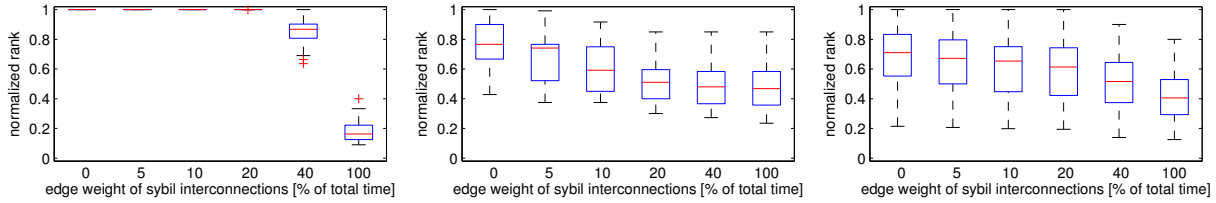
Beyond Social Network-Based Defences: We have shown that an attacker who is willing to pay the price of creating edges can easily infiltrate a community. On the bright side, this spares all the other communities; however, we may also want to protect the attacked community. Considering a defence in depth approach, we can add more layers of security [24], such as a complementary social network approach (see Sec. 2), based on a network of secure pairing [25], imported form an online social network, or based on communication behavior such as phone calls and text messages. However, such networks may be unavailable, too sparse, or also infiltrated by an attacker. Anyway, if the attacker can infiltrate the local community, any social sybil defence is helpless.

A different approach is to find patterns in the attackers behavior



(a) TVCM (b) ETH (c) DART

Figure 6: Impact of # of sybils on the % of infiltrated Sybils. Real Sybils, not interconnected, following 1 target node.



(a) TVCM (b) ETH (c) DART

Figure 7: Impact of Sybil interconnections on the trust rank of the attacker. 80 real Sybils, following 1 target node.

or that result from the introduction of sybils. Piro et al. [15] tries to detect sybils based on the fact that they are always seen at the same time. Looking at the introduction of sybils into the contact graph, there are also patterns that emerge. Generally, nodes that are in contact for a long time are also more similar, as they see many of the same neighbors. However, an attacker who creates many sybils and does not interconnect them will have long contacts to its target, but show little similarity as the honest node sees all sybils, but each sybil does not see the others. While this sounds promising as a detection scheme, especially as the similarity decreases with an increasing number of sybils, such an approach is easily fooled and will naturally have false positives. One might weigh the contacts by the similarity to penalize the attacker but such an approach is difficult to implement and not very effective as preliminary experiments have shown.

Other patterns may be discovered and used for detection, especially if an attacker resorts to misreporting contacts to honest nodes. However, patterns can always be faked if the attacker knows about them. Nonetheless, knowledge does not come for free and such machine learning approaches should be further explored and applied.

7. CONCLUSION

In this paper we studied the anatomy of the sybil attack in an OppNet. While sybil attacks have been studied for many distributed systems, OppNets come with additional challenges as well as useful properties. We show that infiltrating the underlying contact graph of OppNets comes at a significant cost in terms of time, as nodes need to be actively followed. Furthermore, the reach of an attack is limited by the targets community. While in this paper we show the limits of the sybil attack in terms of general trust establishment, which is an important first step, there is much more to explore, such as sybils' impact on routing specific metrics.

8. REFERENCES

[1] Sermpezis P and Spyropoulos T. *Understanding the effects of social selfishness on the performance of heterogeneous opportunistic networks*. Elsevir ComCom, 48, 2014.

[2] Li Y, Hui P et al. *Evaluating the impact of social selfishness on the epidemic routing in delay tolerant networks*. IEEE Communications Letters, 14(11), 2010.

[3] Chen BB and Chan MC. *MobiCent: a Credit-Based Incentive System for Disruption Tolerant Network*. INFOCOM. 2010.

[4] Krifa A, Barakat C et al. *Mobitrade: trading content in disruption tolerant networks*. Chants. 2011.

[5] Yu H, Kaminsky M et al. *SybilGuard: Defending Against Sybil Attacks via Social Networks*. IEEE/ACM ToN, 16(3), 2008.

[6] Yu H, Gibbons PB et al. *SybilLimit: A Near-Optimal Social Network Defense Against Sybil Attacks*. IEEE/ACM ToN, 18(3), 2010.

[7] Danezis G and Mittal P. *Sybilinfer: Detecting sybil nodes using social networks*. NDSS. 2009.

[8] Wei W, Xu F et al. *SybilDefender : A Defense Mechanism for Sybil Attacks in Large Social Networks*. IEEE Trans on Parallel and Distributed Systems, 24(12), 2013.

[9] Cao Q, Sirivianos M et al. *Aiding the detection of fake accounts in large scale social online services*. NSDI, 2012.

[10] Viswanath B, Post A et al. *An analysis of social network-based Sybil defenses*. SIGCOMM Computer Communication Rev, 40(4), 2010.

[11] Alvisi L, Clement A et al. *SoK: The Evolution of Sybil Defense via Social Networks*. 2013 IEEE Symposium on Security and Privacy, (2):382–396, 2013.

[12] Hui P, Crowcroft J et al. *BUBBLE Rap: Social-based Forwarding in Delay Tolerant Networks*. MobiHoc. 2011.

[13] Hossmann T, Spyropoulos T et al. *Know Thy Neighbor: Towards Optimal Mapping of Contacts to Social Graphs for DTN Routing*. INFOCOM. 2010.

[14] Quercia D and Hailes S. *Sybil Attacks Against Mobile Users: Friends and Foes to the Rescue*. INFOCOM. 2010.

[15] Piro C, Shields C et al. *Detecting the Sybil Attack in Mobile Ad hoc Networks*. Securecomm and Workshops. 2006.

[16] Tangpong A, Kesidis G et al. *Robust Sybil Detection for MANETs*. ICCCN. 2009.

[17] Ureten O and Serinken N. *Wireless security through RF fingerprinting*. IEEE Canadian Journal of Electrical and Computer Engineering, 32(1), 2007.

[18] Abbas S, Merabti M et al. *Lightweight Sybil Attack Detection in MANETs*. IEEE Systems Journal, 2012. ISSN 1932-8184.

[19] Andersen R, Chung F et al. *Local Graph Partitioning using PageRank Vectors*. 2006.

[20] Henderson T, Kotz D et al. *The changing usage of a mature campus-wide wireless network*. Computer Networks, 52(14), 2008.

[21] Hsu WJ, Spyropoulos T et al. *Modeling Spatial and Temporal Dependencies of User Mobility in Wireless Mobile Networks*. IEEE/ACM ToN, 17(5), 2009.

[22] Blondel VD, Guillaume JL et al. *Fast unfolding of communities in large networks*. Journal of Statistical Mechanics: Theory and Experiment, (10), 2008.

[23] Newman MEJ. *Analysis of weighted networks*. Physical Review E, 70(5), 2004.

[24] Stytz MR. *Considering defense in depth for software applications*. Security & Privacy, IEEE, 2004.

[25] Čapkun S, Hubaux JP et al. *Mobility helps peer-to-peer security*. IEEE TMC, 5(1), 2006.