This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2010 proceedings

Quantitative Analysis of the Sybil Attack and Effective Sybil Resistance in Peer-to-Peer Systems

Oliver Jetter, Jochen Dinger, Hannes Hartenstein
Institute of Telematics, Karlsruhe Institute of Technology (KIT), Germany
oliver.jetter@kit.edu, jochen.dinger@kit.edu, hannes.hartenstein@kit.edu

***Abstract*—Current peer-to-peer (P2P) systems are vulnerable to a variety of attacks due to the lack of a central authorization authority. The Sybil attack, i.e., the forging of multiple identities, is crucial as it can enable an attacker to control a substantial fraction or even the entire P2P system. However, the correlation between the resources available to an attacker and the resulting influence on the P2P system has yet not been studied in detail. The contributions of our paper are twofold: i) we present an approach for assessing the actual threats of Sybil attacks and ii) we propose a distributed approach to limit the impact of Sybil attacks effectively. Therefore, we conduct a thorough analysis of the Sybil attack w.r.t. the resource requirements to operate Sybil nodes and we investigate the quantitative influence of Sybil nodes on the overall system. Our study focuses on Kademlia, a very popular distributed hash table (DHT) which is for instance used in BitTorrent. We ran extensive Internet measurements within the BitTorrent DHT to determine the actual required resources to operate nodes. To evaluate the quantitative influence of Sybil nodes, we additionally conducted a comprehensive simulation study. The results show that upstream network bandwidth is the dominating factor concerning resources. Furthermore, we illustrate that small portions of Sybil nodes are tolerable in terms of global system stability. Finally, we propose a new approach called *RACING* to improve the resistance of DHTs against Sybil attacks. By establishing a new distributed identity registration procedure based on IP addresses, we are able to effectively limit the number of Sybil nodes.**

I. Introduction

P2P systems have become a fundamental part of today's communication. Several examples for currently widely-deployed P2P systems exist, such as the telephony system Skype [1] or the content distribution network BitTorrent [2]. One of the most important characteristics is scalability making it attractive for resource-demanding applications. Due to the distributed self-organizing concept, these systems can cope with a huge number of participants and are still able to provide sufficient performance. At this point, approaches based on central components are often limited or can only be realized with complex and therefore more expensive systems. The ability of P2P systems to scale with a growing number of participants is also complemented with a high robustness against failures of single nodes. To achieve this robustness, redundancy mechanisms (i.e., data replication and alternative routing paths) are employed.

Although the decentralized characteristics of P2P systems offer several advantages, the missing central instance comes along with a certain vulnerability to attacks as central instances typically act as authorization authorities. A very severe one is the Sybil attack which was first formulated by Douceur [3]. An attacker who performs a Sybil attack tries to forge multiple identities to subvert existing redundancy mechanisms. If he succeeds, he is in the position to use these identities to control a certain fraction of the P2P network. Assuming that the attacker is able to occupy sufficient identities, he ultimately can influence and control the entire system. Therefore, the Sybil attack poses a serious threat to P2P systems.

Douceur shows with his abstract model that distributed approaches to defend the Sybil attack cannot be completely secure. He proposes a central authority certifying all identities as the only effective solution. However, even "secure" solutions based on central authorities suffer from the problem of distinguishing different entities. For instance, if credit cards are used as distinction criteria, an attacker can easily obtain multiple cards. In addition, his model assumptions do not comprise the resources an attacker needs. Hence, the vulnerability is unclear in case of a more specific model that reflects current P2P systems operated throughout the Internet. Sybil attacks require a non-negligible amount of resources as the operation of each node needs its own resources such as CPU cycles, main memory, and network bandwidth. Thus, there is a chance to prevent Sybil attacks with a distributed approach that is appropriately secure considering the available resources of an attacker and the huge number of participants in current P2P systems. An important question is how many resources are necessary to perform a Sybil attack successfully. Despite several existing papers related to the topic of Sybil attacks, a quantitative analysis of the actual resource consumptions is missing. Furthermore, most papers only state that Sybil attacks are either possible or not, but they do not present a metric for the influence gained with Sybil nodes. As a result, it is not possible to assess the actual threat of Sybil attacks.

To tackle the above mentioned issues, we define a metric to be able to assess the success of a Sybil attack. Based on this metric, we conduct a quantitative resource-based analysis in two steps to evaluate the potential influence of Sybil attacks. First, we perform several Internet measurements to determine the required resources to operate nodes. In a second step, a simulation study depicts the quantitative influence of Sybil nodes with respect to the overall system. As a conclusion, we present a new distributed identity registration procedure that improves the resistance of DHTs against Sybil attacks by effectively limiting the number of Sybil nodes per IP address.

The paper is organized as follows. In Section II we survey

978-1-4244-6404-3/10/$26.00 ©2010 IEEE

the most relevant related work. In Section III we pick up Douceur's work and extend the underlying model in the context of P2P systems. Section IV presents our results of a resource-based analysis concerning the quantitative influence of Sybil nodes. Based on these insights, we present our distributed approach to limit the influence of Sybil nodes in Section V. Section VI concludes the paper.

II. Related Work

There are three different surveys that cover research work related to the Sybil attack: Levine et al. [4], Risson and Moors [5] with a more specific view on robustness of DHTs, and Urdaneta et al. [6] who provide a very detailed survey about DHT attacks in general. Papers related to the Sybil attack and their approaches for defending against it, can be categorized as follows: Certification by trusted authorities, utilization of social networks, and examination of node resources.

A central authority certifying identities was already proposed in Douceur's work [3] as an approach to prevent Sybil attacks. Several papers follow Douceur and present similar solutions, e.g. [7] and [8]. A major disadvantage of this approach is the implementation of a single point of failure that can result in scalability problems. The central authority additionally forms a potential target for attackers. Furthermore, the essential question which characteristics to use for distinguishing entities also remains open. Even identifiers like credit or identity cards are not completely secure within this context as a participant can obtain multiple ones. Moreover, it is unclear who could be responsible and operate such an authority. In this context, IP addresses play a special role. They can be viewed as identities due to their unique assignment. Hence, the trusted authority is represented by the IANA (Internet Assigned Numbers Authority). Stoica et al. leverage IP addresses to create and verify node identifiers by hashing the corresponding IP address [9]. However, some issues remain open, e.g. the usage with NAT routers (i.e., multiple nodes share *one* IP address) or with IPv6 (each participant occupies numerous IP addresses as he can choose the last 64 bits arbitrarily, see [10]). We will resume this concept in Section V and present the use of IP addresses as identification objects to develop a new method that can prevent Sybil attacks.

Yu et al. [11] and Hota et al. [12] propose the use of social networks instead of a third-party authority to establish trust relations between the participants. The trust relations have to be established by an external exchange of keys. In a similar fashion, Danezis et al. [13] map their concept of a *bootstrap graph* to social relations. In both cases, it is questionable if such a social network generally exists. Participants of P2P systems are often anonymous and do not know each other. Furthermore, the external exchange of keys is not specified and might not be convenient for most use cases either.

Approaches defeating Sybil attacks by examining the resources of a node, such as CPU cycles or network bandwidth, are based on the assumption that an attacker can only occupy a limited number of Sybil nodes due to his finite resources. Thus, the attacker has to prove that he actually holds these resources. Otherwise, he is rejected and cannot participate anymore. Existing papers deal with the resource *computing capacity*. Rowaihy et al. [14] and Borisov [15] use *cryptographic puzzles* to detect possible attackers. Baumgart et al. [16] apply crypto puzzles to limit the generation of node identifiers. The papers remain on an abstract level regarding the modeling of resources and they do not focus on other resources than CPU cycles, such as network bandwidth or memory consumption. Hence, a complete evaluation of an attacker's potential influence is not possible.

III. The Sybil Attack "revisited"

According to Douceur's definition, a Sybil attack is performed in case an entity manages to establish more than *one* identity. Thus, two identities can already be sufficient. For example, using a second identity to influence Internet auctions can increase the revenue as a seller. However, two identities are normally not sufficient to control an entire P2P system.

Analyzing the Sybil attack shows that there is not *the* Sybil attack but a *class* of Sybil attacks. Actually, any attack can become a Sybil attack by using multiple Sybil nodes (called *Sybils*). However, the number of required Sybils varies considerably depending on the specific P2P system and the attacker's objectives. For instance, DHTs without any redundancy mechanisms are already vulnerable to Eclipse attacks [8] using only a few Sybils. However, it is necessary to control a substantial fraction of nodes in DHTs with well-chosen redundancy mechanisms.

In this paper we investigate the Sybil attack in the context of Kademlia, a very popular DHT. Our focus is primarily on Sybil attacks that try to establish as many identities as possible in order to achieve the objective of an actual attack, like controlling the content of a content distribution network.

The key indicator to assess the influence of an attacker is the fraction of connections from well-behaving nodes to Sybils. The more Sybil connections a regular node occupies, the higher is the chance that it will contact these Sybils. From a technical point of view, connections between nodes correspond to routing table entries. Thus, the success of a Sybil attack can be evaluated by using the following metric: *fraction of malicious routing table entries*. Malicious routing table entries are entries that point to Sybils. We only evaluate routing tables of well-behaving nodes as performance indicator because malicious nodes are already controlled by the attacker himself. Thereby, we are able to quantify the success of a Sybil attack. An attacker will achieve his actual objective more likely if the fraction of malicious routing table entries is higher. Any other chosen metric, such as the lookup failure rate, is scenario specific and thus cannot cover the entire class of Sybil attacks. Moreover, others metrics can be deduced from our metric w.r.t. the Sybil attack.

IV. Assessment on the Influence of Sybils

Two steps are required to perform a resource-based analysis that builds the foundation to assess the actual threat which arises from Sybils: the measurement of required resources to

operate *one* node and a quantitative analysis of the influence of Sybils. The results will provide a basis to evaluate the influence of an attacker depending on his available resources.

As already mentioned in Section III, the influence of Sybils mainly depends on the underlying P2P system. Thus, a resource-based analysis can only be conducted using a specific P2P network. In our paper we use *Kademlia* [17] due to its popularity. For all Internet measurements we consider the BitTorrent DHT [18] because it is based on Kademlia and is used by millions of users (see [19]). However, the assessment methodology and the Sybil limitation approach presented in Section V can be transferred to other DHTs as well.

A. Resource-based Analysis

We conducted measurements to analyze the consumed resources per node: bandwidth related to incoming and outgoing traffic, CPU load and memory consumption. Existing measurements [19], [20], [21] focus on the characterization and optimization of P2P systems by measuring churn rate, control overhead, and lookup performance etc., but do not measure the necessary node resources with the required level of detail.

Our test nodes participated in the BitTorrent DHT but did not search for content actively. We used a slightly modified version of *libtorrent* [22] (based on v.0.13) as node software. Our test machine was connected to the Internet by our campus network with a symmetric link of 100 Mbit/s. All network traffic was captured with *Wireshark* [23]. For this purpose, we developed a *Dissector*, a Wireshark plugin to analyze BitTorrent DHT traffic by differentiating packet types, such as Kademlia Ping or Find Node.

We conducted measurement campaigns for 10 and 31 days. Each time, we operated 5 nodes simultaneously. Due to space requirements, we only present summarized results of those measurements in Table I. More details can be found in [24]. Our measurements show that the decisive factor concerning resources is bandwidth, in particular required upstream bandwidth. On average, a node produced about 25 kbit/s of outgoing traffic. However, only a negligible fraction was used for routing table maintenance whereas the dominant part was caused by requests of other nodes (exemplary for two days in Fig. 1). CPU load and memory consumption stayed constant and were insignificant throughout all measurements. There are possibilities to optimize CPU load and memory usage. However, it is almost impossible to reduce the required network bandwidth. Nodes that ignore incoming requests will be dropped from other routing tables due to a timeout mechanism. Hence, these nodes loose connectivity, cannot influence the DHT anymore, and will be useless for Sybil attacks.

B. Quantitative Influence of Sybils

To evaluate the quantitative influence of Sybils, we performed a simulation study. As expected, operating Sybils in a regular way has a linear influence on the P2P system, i.e., operating a total of 20% Sybils without modifying the node behavior results in a fraction of 20% malicious routing table entries (see Fig. 2, w/o RTP). However, modifying the node behavior can increase an attacker's influence.

TABLE I
AVERAGE CONSUMPTION OF RESOURCES PER DHT NODE

	Traffic Downstream	Traffic Upstream	CPU Load	Memory Consumption
Run 1	12 kbit/s	22 kbit/s	—	—
Run 2	11 kbit/s	20 kbit/s	0.13 %	4.4 MByte
Run 3	18 kbit/s	28 kbit/s	0.13 %	2.4 MByte
Run 4	16 kbit/s	30 kbit/s	0.28 %	3.0 MByte

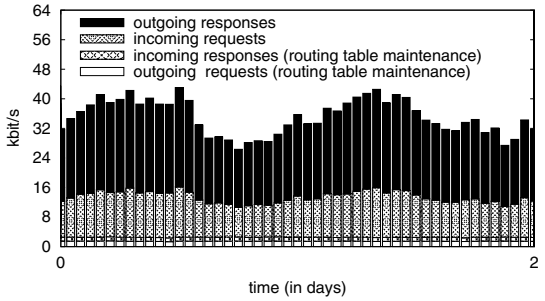

Fig. 1. Traffic analysis of a BitTorrent DHT node exemplary for two days

By means of *Routing-Table-Poisoning (RTP)*, attackers try to achieve a disproportionate high influence of their Sybils. For this purpose, malicious nodes systematically distribute forged routing information by sending node information about Sybils exclusively. The goal is to achieve a higher fraction of Sybil addresses in the routing tables of well-behaving nodes. As the following results show, RTP can increase an attacker's influence very well. The reason for this behavior is that well-behaving nodes also start to spread forged information unwittingly and thus contribute to the phenomenon. In the following we show the impact of Sybils in combination with RTP. As a result, it is possible to determine the point where an attacker occupies a critical fraction of Sybils enabling him to control a substantial part of the system.

As simulator we used *OMNeT++* (v. 3.4b2) [25] in combination with *OverSim* (v. 2008/09/19) [26] which is a collection of overlay protocols. As underlay model we used the model *SimpleUnderlay* that offers a simplified IP stack but provides a sufficient level of detail. If not mentioned otherwise we used the following parameters: a total of 1000 well-behaving nodes, a DHT refresh interval of 15 minutes (according to specification [18]), and an average node lifetime of 3 hours (Pareto distribution according to Yao et al. [27]). We simulated a time of 166 hours and evaluated our system afterwards. The following figures always have 95% confidence intervals.

As illustrated in Fig. 3 small fractions are already sufficient to control major parts of the DHT. For instance, 10% Sybils can already accumulate up to 60% of routing table entries within 48 hours. This situation is even more drastic if the attacker is able to operate his Sybils for a long time. Thereby, he is in the position to control up to 80% of all nodes. However, Fig. 3 also shows that a certain fraction of Sybils is necessary to benefit from Routing-Table-Poisoning.

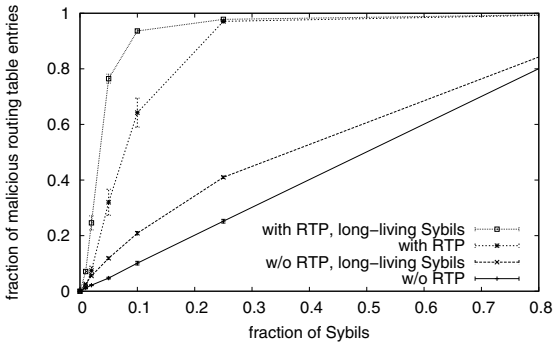

Fig. 2. Relation between fraction of malicious nodes and fraction of malicious routing table entries with and without Routing-Table-Poisoning (RTP)

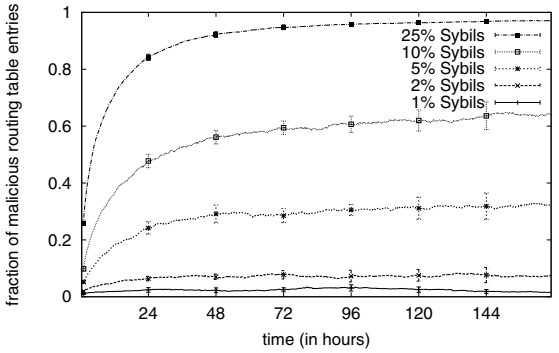

Fig. 3. Sybils with Routing-Table-Poisoning over time

C. Summary

Altogether, Sybil attacks form a serious threat for DHTs. Nevertheless, an attacker has to provide sufficient resources, in particular in DHTs with huge numbers of participants. Although it is not possible to calculate the required resources to perform a Sybil attack in general, we are able to estimate the influence for specific scenarios. For instance, if we assume a DHT with 1,000,000 participants and an attacker wants to control 10% of the network, he has to insert and operate 111,111 Sybils (10% of 1,111,111 nodes in total) simultaneously. In the case of the BitTorrent DHT, the attacker already has to provide an upstream bandwidth of about 2.7 Gbit/s.

Section IV-B also illustrated the impact of Routing-Table-Poisoning. Picking up the above example of 111,111 Sybils, an attacker can already reach a fraction of 60% malicious routing table entries when using Routing-Table-Poisoning. However, as six times more requests are directed to the Sybils which therefore have to send six times more responses, the attacker requires about six times more upstream bandwidth. This fact has to be kept in mind as Section IV-A depicted that the decisive factor concerning resources is bandwidth, in particular upstream bandwidth. Moreover, a certain fraction of Sybils is necessary to benefit from Routing-Table-Poisoning. Its impact is only marginal if used in conjunction with less than 5% Sybils.

V. Identification-based Limitation

The preceding quantitative resource-based analysis showed that bandwidth is the most decisive resource regarding consumption. Hence, this can already be a barrier for attackers. This limitation could be further increased by artificially raising the periodic traffic per node which is necessary to maintain the routing tables. However, such an approach would waste plenty of resources of regular nodes and therefore be inefficient. Additionally, users with limited bandwidth might not be able to participate in the P2P system anymore.

Today, botnets offer a convenient way to acquire an enormous accumulated bandwidth. Consequently, we decided to enforce an additional limitation of an attacker's influence. In this case, IP addresses and IP address ranges in IPv6 respectively can be used as an additional resource. Some authors criticize that IP addresses can be spoofed (amongst others [3] and [11]). However, communication, in particular receiving packets, with spoofed IP addresses on a global scale is not feasible without further effort.

The central idea behind our approach is to limit an attacker's influence, but at the same time not to handicap well-behaving participants in using the P2P system. Typical P2P users merely occupy one IP address and operate one node and a few nodes if they are using a NAT router, respectively. In contrast to that, Sybil attackers are tempted to run as many nodes as possible per machine to maximize their resource efficiency. Therefore, the objective of our approach is to effectively limit the number of Sybils per IP address and IP address range respectively. Clearly, botnets also occupy numerous IP addresses. Nevertheless, we will show in the evaluation that our approach is secure enough even in case of botnets.

A. RACING (Routing-ACtive-INactive-Grouping)

Our approach partitions the DHT network in *routing-active* and *routing-inactive* node groups (*RANs* and *RINs*). Both node types operate like stated in the Kademlia protocol specification except that addresses of RINs are not used in any routing table. Furthermore, key-value pairs are not stored on RINs. Thus, RINs are not involved in the routing process and cannot influence the DHT. Nevertheless, they are able to use all services of the DHT, i.e., lookup and store operations.

Consequently, to limit an attacker's influence within the DHT, it is essential to prevent that he can establish too many routing-active Sybils. Therefore, we restrict the number of routing-active Sybils effectively per IP address and IP address range respectively by applying a self-registration procedure.

We use a refined version of the self-registration approach in [28] to differentiate between RANs and RINs per IP address and IP address range respectively. The idea is that a node registers *itself* in the DHT by calculating its own node identifier. At the same time, other nodes are able to verify correct node identifiers easily. As IP addresses are centrally assigned by the IANA, we do not need an additional central authority and therefore can preserve the totally decentralized character of the P2P system.

To join the DHT, a node has to choose an appropriate registration identifier $regId$. Based on its IP address $ipAddr$ the corresponding node identifier $nodeId$ is calculated through a consistent hash function h such as SHA-1 according to the following expression (string concatenation is denoted by $\oplus$):

$$nodeId := h(ipAddr \oplus regId) \tag{1}$$

To detect an unused registration identifier, a node calculates potential node identifiers according to (1). Each time the node probes whether or not the calculated node identifier is already in use by looking up the corresponding key. As soon as an unused registration identifier is determined, the node adds this information to all packets and henceforward can be identified as a RAN. If all registration identifiers are already used, a node remains routing-inactive. Although RINs do not necessarily need a node identifier, they should calculate one with any registration identifier greater than $regId_{max}$ to determine their positions within the P2P network. Thus, all RINs can be distributed uniformly within the DHT to balance the additional load for RANs.

For RANs applies $regId \in \{1, \ldots, regId_{max}\}$. As a result, the maximum number of Sybils per IP address, that are routing-active, can be set by $regId_{max}$. A RAN adds its registration identifier as well as the resulting node identifier to all P2P packets. This mechanism enables the differentiation between RANs and RINs. A receiving node is able to verify the registration process by applying (1) (source IP address and $regId$ can be extracted from any packet).

To verify a correct node registration, one has to check the IP address and the node identifier. The verification of the IP address can be done by sending a challenge, i.e., a random number, to the source node. Receiving a response within a few seconds shows that the node actually occupies the IP address. The node identifier can be checked via (1) as the registration identifier is included in any packet received from the node. A node with an invalid node identifier or a registration identifier greater than $regId_{max}$ is considered as routing-inactive and thus is not involved in the routing process.

Instead of limiting the number of RANs per IP address, it is also possible to enforce a limitation per IP address range. For this purpose, we have to replace $ipAddr$ in (1) by the IP address prefix of the corresponding address range. This is especially useful in combination with IPv6 as an attacker can choose the last 64 bits independently (see [10]). More information on this issue can be found in [24].

The choice of $regId_{max}$ cannot be answered in general. It depends on the specific P2P system (e.g. the average number of participants) as well as the average number of regular nodes per IP address (due to operation behind NAT routers). The latter is very difficult to determine because of the vast number of participants and the essential requirement to determine all IP addresses of all running nodes simultaneously. Due to that, we can only recommend an estimation. Generally, $regId_{max} \leq 4$ should be sufficient to retain the scalability of a P2P system and limit the influence of an attacker at the same time. As we show in the next section, a too small value only affects the overhead for RANs but never compromises the functionality of the P2P system for any node.

Current versions of the popular *Azureus* [29] client also limit the number of nodes per IP address (see [30]). However, they do not fully enforce the limitation and furthermore tolerate 2000 nodes per IP address.

B. Evaluation

We evaluate our approach on the basis of two criteria: *Sybil resistance* and *load balancing*.

1) Sybil resistance: The assignment process of node identifiers has to be verifiable to make a P2P protocol preferably resistant against Sybil attacks (cf. principles in [31]). Moreover, the number of nodes per entity has to be as small as possible. The verification of node identifiers can be performed easily via (1) due to the proposed self-registration procedure. The calculation is independent of the fraction of Sybils within the P2P system. Furthermore, this calculation cannot be influenced by an attacker, as it is executed locally and without information from any other node. A limitation of nodes per participant is enforced by $regId_{max}$. This bound limits the number of Sybils per IP address range effectively.

In case, two different nodes use the same registration identifier, this can result in an inconsistent routing behavior. Nevertheless, this does not pose a serious problem due to the existing redundancy and the fact that node identifiers are not included multiple times in routing tables of other nodes. Actually, this also happens in several current P2P systems as node identifiers are chosen randomly. Regarding the aspect of Sybil attacks, an attacker cannot benefit from repeated use of registration identifiers as nodes with the same node identifier would be treated as *one* node.

Additionally, the limited number of potential registration identifiers means that a node has almost no influence on the choice of its node identifier. Hence, this approach can also prevent against other attacks such as the Eclipse attack, because an attacker is not able to completely occupy certain key ranges of a DHT (cf. [8]).

Our approach does not prevent an attacker from acquiring multiple IP addresses. However, it increases the Sybil resistance even in case of attackers with lots of IP addresses (e.g. using a botnet) as every machine can only run $regId_{max}$ Sybil nodes instead of hundreds or even thousands. For example, assuming $regId_{max} = 4$ and a botnet of 50,000 nodes, we can limit the number of routing-active Sybils to 200,000. Applying this example to the real-world BitTorrent DHT with its approximately 4 million users means that the attacker can only control about 5% of the P2P system. Without our mechanism, the attacker could at least run 2,000,000 Sybils assuming an upstream bandwidth of 1 Mbit/s per node and therefore control the majority of the P2P system. Furthermore, our approach directly influences an attacker's costs as the "rent" of a botnet is based on the number of bots (currently average costs are $0.50 per bot, see [32]). Despite the fact that a few attackers might be able to buy such botnets, it requires a substantial financial background. That is why we consider our

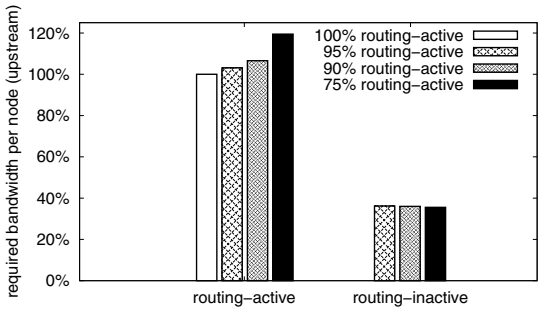

Fig. 4. Average outgoing network traffic of routing-active nodes (RANs) and routing-inactive nodes (RINs) with varying fraction of RANs

approach as appropriately secure for most Internet scenarios. Finally, our approach provides a methodology to assess the risk of an attacker controlling a P2P system.

2) Load balancing: Due to the partitioning of RANs and RINs, RANs experience an additional load because they have to accomplish the routing process for RINs as well. However, it is reasonable to assume that the fraction of RINs will be comparatively small in case $regId_{max}$ is chosen appropriately. Our simulations showed that the additional required upstream bandwidth for RANs is acceptable. The overhead remains marginal for small fractions of RINs. Even in case of 25% RINs, the additional load is less than 20% for RANs (see Fig. 4). Furthermore, there is only a negligible overhead regarding the required downstream bandwidth that results from smaller packet sizes of requests compared to corresponding responses. Overall, the overhead is reasonable since we only expect a small fraction of RINs when using $regId_{max} = 4$.

VI. Conclusion

Previous work does not allow a complete assessment of the actual threat arising from Sybil attacks. Therefore, we followed a resource-based and cost-oriented approach to analyze the quantitative influence of Sybils concerning the overall system. On the one hand, we presented the resources currently necessary to successfully perform a Sybil attack. On the other hand, we depicted a methodology to assess these resources and costs for the future, respectively. Altogether, the assessment of the influence of Sybils provides an evaluation model for the actual risk of Sybil attacks. Furthermore, we proposed a distributed approach to limit the number of Sybils per IP address and showed the effectiveness of the limitation.

References

[1] Skype, http://www.skype.com/, 2009.
[2] BitTorrent, http://www.bittorrent.com/, 2009.
[3] J. R. Douceur, "The sybil attack," in *IPTPS '02: The 1st Int'l Workshop on Peer-to-Peer Systems*. Springer, 2002, pp. 251–260.
[4] B. Levine, C. Shields, and N. Margolin, "A survey of solutions to the Sybil attack," University of Massachusetts Amherst, TR 2006-052, 2006.
[5] J. Risson and T. Moors, "Survey of research towards robust peer-to-peer networks: Search methods," *Computer Networks*, vol. 50, no. 17, pp. 3485–3521, 2006.
[6] G. Urdaneta, G. Pierre, and M. van Steen, "A survey of DHT security techniques," *To appear in ACM Computing Surveys*, 2009.
[7] K. Hildrum and J. Kubiatowicz, "Asymptotically efficient approaches to fault-tolerance in peer-to-peer networks," in *17th Int'l Symp. on Distributed Computing (DISC)*, 2003, pp. 321–336.
[8] M. Castro, P. Druschel, A. Ganesh, A. Rowstron, and D. S. Wallach, "Secure routing for structured peer-to-peer overlay networks," in *OSDI'02: Proc. of the 5th Symp. on Operating Systems Design and Implementation*. ACM, 2002, pp. 299–314.
[9] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for internet applications," in *SIGCOMM '01: Proc. of the 2001 Conf. on Applications, Technologies, Architectures, and Protocols for Computer Communications*, vol. 31, 2001, pp. 149–160.
[10] R. Hinden and S. Deering, "RFC 4291: IP version 6 addressing architecture," IETF, http://www.ietf.org/rfc/rfc4291.txt, 2006.
[11] H. Yu, M. Kaminsky, P. B. Gibbons, and A. Flaxman, "Sybilguard: Defending against Sybil attacks via social networks," in *Proc. of the 2006 Conf. on Applications, Technologies, Architectures, and Protocols for Computer Communications*. ACM, 2006, pp. 267–278.
[12] C. Hota, J. Lindqvist, K. Karvonen, A. Ylä-Jääski, and M. C.K.J, "Safeguarding against Sybil attacks via social networks and multipath routing," in *NAS 2007: Int'l Conf. on Networking, Architecture, and Storage*. IEEE, 2007, pp. 122–132.
[13] G. Danezis, C. Lesniewski-laas, M. F. Kaashoek, and R. Anderson, "Sybil-resistant DHT routing," in *ESORICS 2005: 10th European Symp. on Research in Computer Security*. Springer, 2005, pp. 305–318.
[14] H. Rowaihy, W. Enck, P. McDaniel, and T. L. Porta, "Limiting Sybil attacks in structured P2P networks," in *INFOCOM 2007: 26th IEEE Int'l Conf. on Computer Communications*. IEEE, 2007, pp. 2596–2600.
[15] N. Borisov, "Computational puzzles as Sybil defenses," in *P2P '06: Proc. of the 6th IEEE Int'l Conf. on Peer-to-Peer Computing*. IEEE, 2006, pp. 171–176.
[16] I. Baumgart and S. Mies, "S/Kademlia: A practicable approach towards secure key-based routing," in *ICPADS '07: Proc. of the 13th Int'l Conf. on Parallel and Distributed Systems*. IEEE, 2007, pp. 1–8.
[17] P. Maymounkov and D. Mazieres, "Kademlia: A peer-to-peer information system based on the XOR metric," in *IPTPS '02: The 1st Int'l Workshop on Peer-to-Peer Systems*. Springer, 2002.
[18] A. Loewenstern, "BitTorrent protocol spec., bep 5: DHT protocol (draft, v. 11031)," http://www.bittorrent.org/beps/bep_0005.html, 2008.
[19] S. A. Crosby and D. S. Wallach, "An analysis of BitTorrent's two Kademlia-based DHTs," Rice University, Tech. Rep., 2007.
[20] J. Falkner, M. Piatek, J. P. John, A. Krishnamurthy, and T. Anderson, "Profiling a million user DHT," in *IMC '07: Proc. of the 7th ACM SIGCOMM Conf. on Internet Measurement*. ACM, 2007, pp. 129–134.
[21] Y. Qiao and F. E. Bustamante, "Structured and unstructured overlays under the microscope: a measurement-based view of two P2P systems that people use," in *ATEC '06: Proc. of the Annual Conf. on USENIX '06 Annual Technical Conf.* USENIX Association, 2006, pp. 31–31.
[22] libtorrent, http://www.rasterbar.com/products/libtorrent/, 2009.
[23] Wireshark, http://www.wireshark.org/, 2009.
[24] J. Dinger, "The potential of peer-to-peer networks and systems: Architectures, robustness and legal classification (in German)," Ph.D. dissertation, Universität Karlsruhe (TH), ISBN 978-3-86644-327-3, 2009.
[25] OMNeT++, http://www.omnetpp.org/, 2009.
[26] I. Baumgart, B. Heep, and S. Krause, "Oversim: A flexible overlay network simulation framework," in *GI '07: Proc. of 10th IEEE Global Internet Symp.*, 2007, pp. 79–84.
[27] Z. Yao, D. Leonard, X. Wang, and D. Loguinov, "Modeling heterogeneous user churn and local resilience of unstructured P2P networks," in *ICNP '06: Proc. of 2006 IEEE Int'l Conf. on Network Protocols*. IEEE, 2006, pp. 32–41.
[28] J. Dinger and H. Hartenstein, "Defending the Sybil attack in P2P networks: Taxonomy, challenges, and a proposal for self-registration," in *ARES '06: Proc. of the 1st Int'l Conf. on Availability, Reliability and Security*, 2006, pp. 756–763.
[29] Azureus Client, http://azureus.sourceforge.net/, 2009.
[30] M. Steiner and E. W. Biersack, "Crawling Azureus," Institut Eurecom, TR RR-08-223, 2008.
[31] E. Sit and R. Morris, "Security considerations for peer-to-peer distributed hash tables," in *IPTPS '02: The 1st Int'l Workshop on Peer-to-Peer Systems*. Springer, 2002, pp. 261–269.
[32] Y. Namestnikov, "The economics of botnets," Kaspersky Labs, http://www.viruslist.com/analysis?pubid=204792068, 2009.