"Super Nodes" in Tor: Existence and Security Implication

*ChenglongLi[1,3], Yibo Xue[1,2], Yingfei Dong[4] and Dongsheng Wang[1,2]
[1]Tsinghua National Laboratory for Information Science and Technology (TNList), Beijing, 100084, China.
[2]Research Institute of Information Technology (RIIT), Tsinghua University, Beijing, 100084, China.
[3]Department of Computer Science & Technology, Tsinghua University, Beijing, 100084, China.
[4]Department of Electrical Engineering, University of Hawaii, Honolulu, HI 96822, USA.

ABSTRACT
Tor (the second generation onion routing) is arguably the most popular low-latency anonymous communication system now. In this paper, we reexamine the anonymity of Tor based on our observation of "super nodes". These nodes are more available and reliable than other nodes and provide high bandwidth for assisting the system in both performance and stability. We first confirm their existence by analyzing the life cycles of node IP addresses and node bandwidth contributions via two correlation approaches, on a set of self-collected data and a set of real data from the Tor official collection. We then analyze the effect of super nodes on the anonymity of Tor, discuss attacks that exploit such knowledge, and verify our analysis with real data to show potential damages. Furthermore, we investigate new attacks that exploit the knowledge of super nodes. Our simulation results show that these attacks can greatly damage the anonymity of Tor.

Categories and Subject Descriptors
D.6.5 [**Security and Protection**]

General Terms
Security

Keywords
Anonymous communication, Anonymity, Tor

1. INTRODUCTION
Tor is the realization of the second generation onion routing for low-latency anonymous communications. Three types of nodes exist in the Tor network: *directory servers*, *relays* and *bridge relays*. Directory servers are controlled by independent groups[3] and provide global information of Tor relays. Relays are run by volunteers, and they are listed at directory servers and used to build paths for anonymous communications. Bridge relays are not listed in directory servers, and are used as dynamic entry nodes to get around the blocking of public relays by some ISPs. They are chosen by directory servers according to mean uptime and bandwidth[6]. Users learn bridge relays via special emails or sites. The main difference between relays and bridge relays is publically listed or not, and they are both used for building anonymous communication paths. As a result, we consider them the same in our anonymity analysis. For ease of discussion, we use the terms of 'relay' and 'node' alternatively in the following.

ACSAC'11 Dec. 5-9, 2011, Orlando, Florida USA

According to the Tor official report[4], the daily average number of relay nodes is over 2000, and the daily maximum number of directly connecting Tor users is more than 25000 in February 2011. Tor has a broad user base, including various governments, military agencies, enterprises, and individuals. They mainly use Tor for anonymity. Despite its popularity, Tor still has security issues, and several effective attacks have been developed to compromise its anonymity[18-21]. Tor is sometime mis-used for confidentiality, e.g., the founder of Wikileaks claimed that some documents were intercepted at malicious Tor exit nodes[11, 15].

As Tor becomes more popular in recent years, many volunteers make their relays online as long as possible and contribute more bandwidth. As a result, these nodes contribute more in both performance and reliability. We consider these relay/bridge nodes as "*super nodes*", different from other common users who are less reliable and contribute less bandwidth. In this paper, we investigate how these super nodes affect the anonymity of Tor.

To the best of our knowledge, the existence of super nodes in Tor has not been confirmed before. In this paper, we first verify the existence of super nodes by observing the life cycles of node IP addresses and node bandwidth contributions. Our intuition is that super nodes tend to be more available and more likely to be chosen in building communication paths due to their high bandwidth contribution. However, we found that some of these nodes may associated with different IP addresses over time, due to:(1) some (dial-up) users do not have static IP addresses, e.g., a cable modem user may get a slightly different IP address after rebooting; (2) users may change their IP addresses to avoid being detected and blocked; (3) the network condition of some users may change over time.

To deal with the slight changes of node IP addresses, we use two correlation methods (based on C-Class IP address and BGP prefix) to find the IP addresses that are more likely related to certain relay nodes: long lived in a narrow address range and with high bandwidth contribution. Based on experimental results on our self-collected dataset and a Tor official dataset, we have confirmed that super nodes do exist. While they help the system in both performance and reliability, they may cause new security issues. We will analyze these issues in this paper. The main contributions of this paper are:

- We have confirmed the existence of super nodes in Tor based on experimental data.

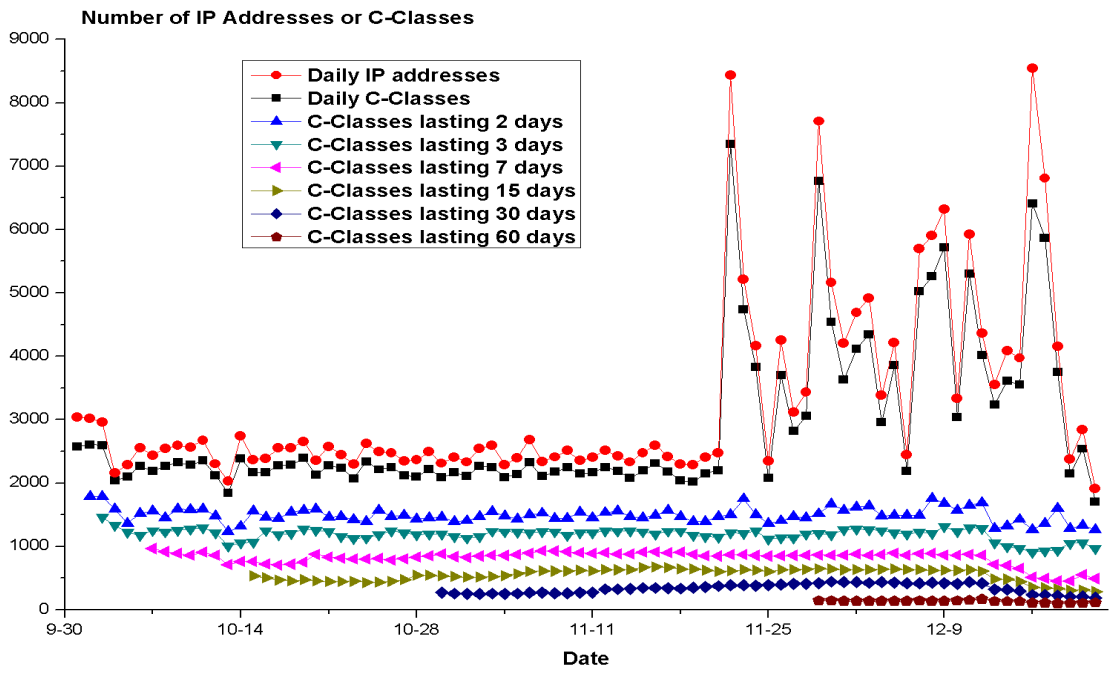


Figure 1.Results of C-Class Correlation on a dataset collected in the US.

- We have analyzed the effect of super nodes on anonymity measure, proposed a new anonymity analysis by considering the effect of super nodes, and discussed the related issues.
- We have investigated potential attacks that exploit the knowledge of super nodes to improve the effectiveness of existing attacks.
- In particular, we have proposed a new theoretical attack which exploits super nodes to cause serious damages.

The remainder of the paper is organized as follows. We will discuss the related work in Sec.2, and introduce the evidences of super nodes in Sec.3. We will then present a new anonymity analysis and discuss its implication in Sec.4. We will discuss several attacks that exploit the knowledge of super nodes in Sec.5, and conclude the paper and discuss our future work in Sec.6.

2. RELATED WORK

Anonymity is one of critical challenges in security systems. Anonymous communication was mostly a pure research topic until 1996: the onion routing[22] was deployed, which is a low-latency anonymous communication system based on Chaum's mix cascades[14]. Tor is the second generation of onion routing, deployed in 2002[3].

The most widely adopted anonymity measure for anonymous communications is the information entropy method [7, 8]. First, entropy is calculated based on the size of effective anonymity set [7], and then the entropy is normalized between 0 and 1 based on the maximum entropy given by the system [8]. Furthermore, the measure of anonymity is defined as a probability (*1-p*) [9], where p is the probability that a sender is identified by an adversary. Moreover, the degree of anonymity is assessed as $A = \log_2 N$ [10], where N is the number of users in an anonymous network. However, the anonymity measures in [9, 10] cannot appropriately represent anonymity in complex anonymous systems. Therefore, a specification framework for information hiding properties was proposed [12] based on the concept of function view: a concise representation of the attacker's partial knowledge about a function. It describes system behavior as a set of functions, and formalizes different information hiding properties in terms of the views of these functions. Hordes anonymous communication protocol provides similar anonymity as Crowds [9] or onion routing, but with some advantages such as making use of anonymity inherent in multicast routing [13]. Recently, a structural highly-distributed anonymous communication system was proposed to handle scalability issues of Tor and other peer-to-peer systems[2]. Attacks to Tor bridges were proposed, and a membership-concealing overlay network was designed [1]. However, all these existing schemes have not considered the effect of "super nodes" and related security issues in Tor or other anonymous communication systems. In the following, we will first confirm the existence of super nodes, analyze their effect on anonymity, and investigate potential attacks exploiting them.

3. FINDING SUPER NODES IN TOR

By the nature of super nodes, they provide high bandwidth and are long-lived, acting as the "backbone" of Tor for reliability and performance. The Tor path selection algorithm determines their existence. For example, when selecting relays to build a path, it requires that the first 16 bits of the IP addresses of relays are different. As a result, the relays on a path are more likely scattered in different domains. This motivates us to classify relays based on their IP address classes. We then extend the classification with BGP prefixes for more generic cases on the Internet.

We use two correlation methods to analyze the life cycles of C-Class addresses and BGP prefixes of relays, in order to identify the existence of super nodes. The main arguments behind our methods are as follows.

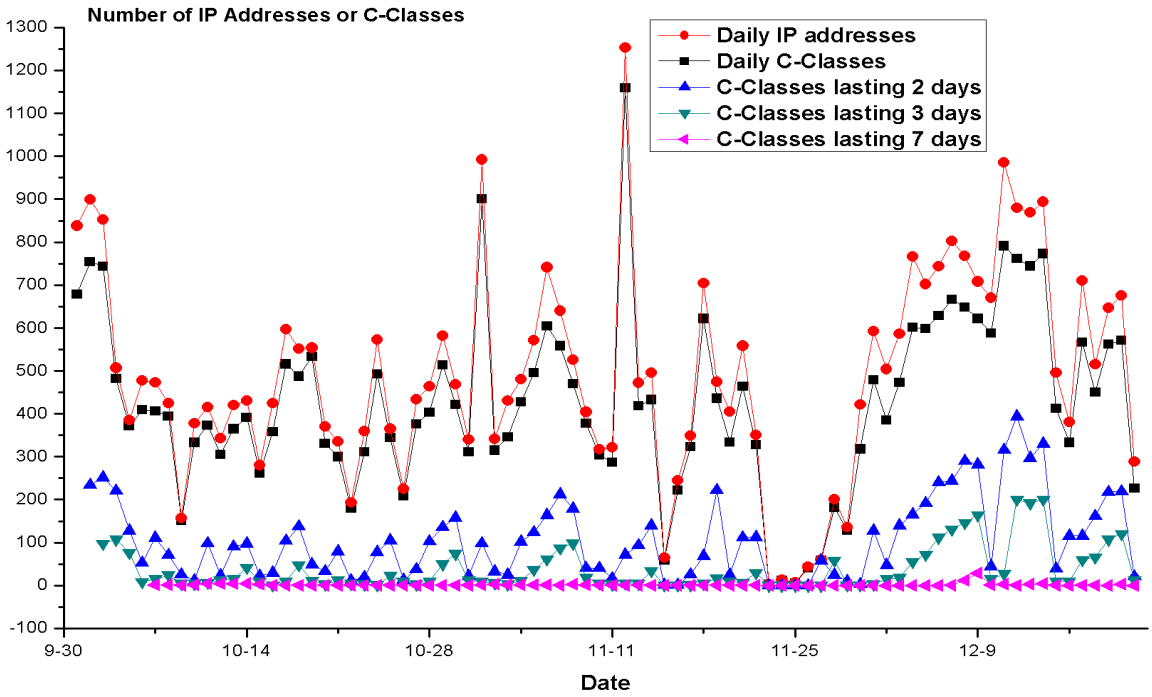


Figure 2.Results of C-Class Correlation in a self-collect dataset in China.

- *One C-Class address is most likely corresponding to one relay node.* Our data reported in the following shows that relays are scattered fairly evenly into different C Classes. (The path selection algorithm builds a path with relays from different C Classes.) The total number of daily IP addresses that we see in the data is very close to the total number of C-Class addresses, i.e., each C Class is likely corresponding to only one relay node.
- *The IP address of a super node is limited to the same C-Class address or BGP prefix.* Although the IP address of a relay may be changed over time, the range of such change is fairly limited in the current IPv4 setting, mostly within the same C Class address or BGP prefix.
- *Tor needs to have a certain number of stable nodes to achieve a proper level of anonymity.* As any anonymous communication systems, the total number of nodes in the system determines the anonymity measure. We try to find the set of reliable nodes by analyzing the life cycles of node IP addresses over different time intervals.
- *Tor path selection algorithm somehow determines which types of nodes become super nodes,* because the bandwidth and flags (including capacity and uptime) are the main factors in the current path selection algorithm. Our analysis in the following confirms that super nodes are *long-lived* and usually have *high bandwidth contribution*.
- *We can easily filter out most normal relays by examining their life cycles and bandwidth contributions.* A normal volunteer usually runs a Tor relay with limited resources for relative short periods and leaves based on its own needs. Because it is less reliable and less capable, it is less likely to be selected in a Tor path. Its IP address (and its associated C-Class address/BGP prefix) is only shown up shortly in the data set.

We obtained two datasets: one is self-collected, and the other is from the Tor official collection.

- Self-collected Datasets: We collected the IP addresses, bandwidth, and other information of Tor relay nodes in both US and China from Sept. 30th, 2010 to Dec. 21st, 2010. In China, we run a Tor relay node without exit permission in the CERNET (Chinese education network), and collected information of Tor nodes connected to it. In the US, we use the same method on a VPS (Virtual Private Server) of *Linode*[16].
- Official Dataset: This dataset is acquired from the Tor official metrics site [4]. We analyzed official consensus data on the site from Jan. 1st, 2010 to Dec. 31st, 2010.

3.1 C-Class Correlation

As supper nodes tend to be online for long periods, we examine the life cycles of C Classes of the *daily* relay IP addresses in each dataset over continuous time intervals of **i** consecutive days, where **i**= 2, 3, 7, 15, 30 or 60.(It represents the uptime of C Classes.) The results of the self-collected data in the US and China are shown in Fig.1 and Fig.2, respectively.

First, the *daily* number of C Classes of relays is just a little smaller than that of *daily* IP addresses of relays in both figures. This means that the IP addresses of relays are fairly evenly distributed among C Classes, both in the US and China. In other words, in most cases, one relay IP address is corresponding to one C-Class address.

Second, as we increase the interval length, the number of *persistent* C Classes in Tor changes very little. By *persistent*, we mean that these C Classes show up every day during the interval. As we see, the large daily fluctuation of normal-relay joins/leaves does not affect the Tor "backbone", consisting of long-lived nodes. In our self-collected datasets, we have seen x=2824 *daily* C Classes occupied by Tor nodes. Without loss of generality, we

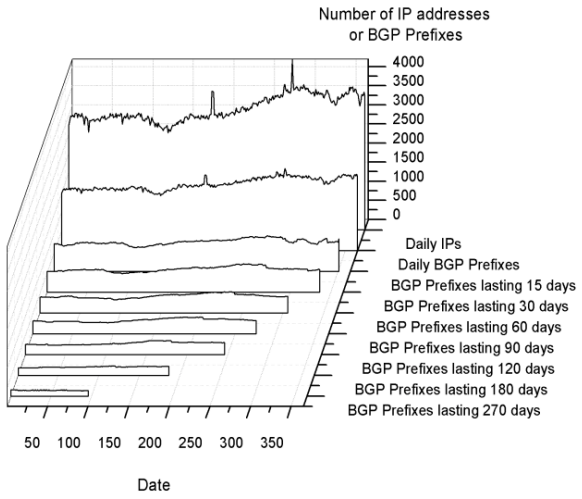


Figure 3.Results of BGP Prefix Correlation.

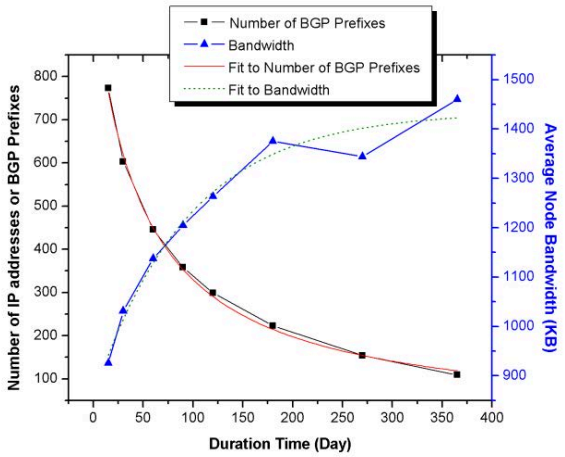


Figure 4. Average Prefix Number and Bandwidth of super nodes.

assume that a node join is independent to the current node distribution. For a node join, the probability that the new node belongs to the existing $\boldsymbol{x}$ persistent C Classes is very small ($\boldsymbol{x/y} \approx 0.02\%$), where $\boldsymbol{y}$ is the total number of C Classes on the Internet, and $\boldsymbol{y}$=2^{24}. The main reason behind the small change of the number of C Classes is that an existing node likely changes its IP address in the same C Class, or the same Classless Inter-domain Routing (CRID) block, corresponding to a BGP prefix.

Third, in Fig.1 (*data collected in the US*), the number of *daily IP addresses* (the top curve) is quite smooth from the beginning to Nov. 21st (with some fluctuations for the remainder of the period). The number of *daily C Classes* (the second curve from the top) behaves similarly. However, the number of persistent C Classes over several days (the six curves at the bottom) is relatively steady all the time for all cases. This fact indicates that the basic structure of Tor network is mainly stable without dramatic changes in spite of the fluctuation of daily relay IP addresses. About $\frac{1}{3}$ of C Classes are still active after seven days. Over 100 C Classes are active more than60consecutive days during the 83-day data collection period. We believe that these nodes are likely the foundation of Tor, supporting its basic anonymity and service quality. They are very likely the "super nodes" that we are looking for. In Fig.2 (*data collected in China*), the results are quite different due to network censorship in China. The number of *persistent* C Classes over several days (the bottom three curves) is very low, compared with that in the US. The number of persistent C Classes drops very fast and often reaches zero after seven days. It is most likely due to active filtering and blocking.

In summary, the data collected in the US reflects common cases on the Internet. The number of persistent C Classes tells us that super nodes are quite stable in spite of a large number of normal nodes join/leave. The number of C Classes over seven (or more) days has almost no change, i.e., super nodes are long lived.

3.2 BGP Prefix Correlation

To further verify the existence of super nodes, we also analyze the BGP-prefixes associated with relays in the Tor official dataset. We chose one-year official consensus data and calculated the daily relay IP addresses in the same BGP prefixes over continuous time intervals of **i** consecutive days, where **i**= 15, 30, 60, 90, 120, 180, 270, and 365. The BGP table was obtained from [5] in Jan., 2011. Fig.3 and Fig.4 show the results. More than 100 *persistent* BGP prefixes were active across the whole year as shown in Fig.3. The number of persistent prefixes (the left Y-axis in Fig.4) drops slowly as we increase the interval length. (One year may be too long to be justified as a super node, because it may switch to another ISP or replaced by a new node for various reasons. We just use it as an example to show the behavior.)

There were$\boldsymbol{y'} \approx 344{,}000$ different BGP prefixes in the BGP table [5]. The peak number of *daily* BGP prefixes of Tor relays is $\boldsymbol{x'} = 2152$. Assuming Tor nodes have no social relations in the real world, a new relay IP address appears in the data due to two cases: a new node joins, or an existing node changes its address. Our results show that a Tor node is likely distributed in a unique (BGP-prefix) address space, while a new node join is only adding a new *daily* BGP prefix into the dataset. The probability that the new address belongs to $\boldsymbol{x'}$existing BGP prefixes is independent to the current node distribution, which equals to $\boldsymbol{x'/y'} \approx 0.06\%$. When an existing relay changes its IP address, it is likely to under the same BGP prefix. The results are similar to those seen in the above C-Class address analysis.

The number of super nodes we see depends on how we define the correlation interval. As shown in Fig.4, it varies from about one thousand to several hundred on the current Tor based on different views. We also calculate the average node bandwidth (the right Y-axis in Fig.4) associated with these persistent BGP prefixes in different time intervals. We firstly calculate the average bandwidth of nodes belonging to the same prefix respectively, and then obtain the final average bandwidth of all persistent BGP prefixes in an interval. As shown in Fig. 4, the average bandwidth of these nodes increases with the interval length. The number of persistent prefixes fits the reciprocal function $y = 1/(a + b * x)$ well, where a=0.001 represents the adjusting parameter, and b=0.00002 represents the attenuation factor. The average bandwidth of these nodes fits the exponential function $y = y_0 + A^{r*x}$ well, where y_0=1435.06, A=-578.27, and r=-0.0106.

3.3 More Discussions of Super Nodes

The Tor path selection algorithm clearly separates super nodes from normal nodes based on reliability and bandwidth (or throughput) contribution. Combining these two parameters together, we can determine the set of super nodes. From the Tor official metrics data [4], the advertised average daily node

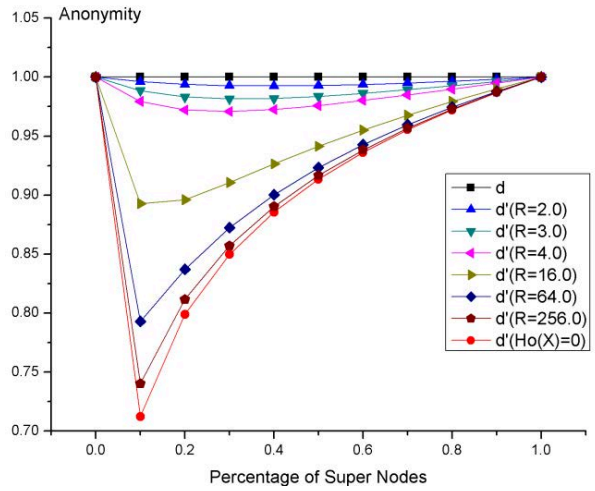


Figure 5.Comparison of d and d′.

bandwidth in 2010 is 354 KB; the real traffic read-and-written by a relay (recorded from April 29th, 2010) has a daily average of 264 KB. As the latest Tor path selection algorithm considers node reliability and bandwidth as critical parameters, it is clear that the uptime and the bandwidth of super nodes are very consistent in the data. The nodes with a long life cycle (even changing their IP addresses slightly) provide more bandwidth for anonymous communications. Because Tor tends to select nodes with more bandwidth, we define the *selection ratio of super nodes over normal nodes* as R, which is the average bandwidth of super nodes over the average bandwidth of normal nodes. It is about 2.6 based on the 15-day correlation interval and about 4 based on the one-year correlation interval. It becomes larger as the interval length increases. Clearly super nodes have significant influence on both system stability and performance.

In summary, we have used two correlation methods to confirm the existence of super nodes over a self-collected data and the Tor official data. We have also identified that the long-lived super nodes usually provide higher bandwidth. We consider these results significant in affecting the anonymity in Tor, because (i) Identifying super nodes from common relay nodes affects the anonymity metrics. (ii) Attackers can exploit the difference to improve existing attacks and introduce new attacks exploiting super nodes and their characteristics. As a result, the defense mechanism of Tor may have to be improved correspondingly. In the next section, we will examine how anonymity is affected due to the existence of super nodes in Tor.

4. THEORETICAL ANALYSIS

4.1 Simple Anonymity Formulation

When using information theory to define anonymity[7, 8], an anonymity set is denoted as$\mathrm{A} = \{\mathrm{a}_1, \mathrm{a}_2, \dots, \mathrm{a}_n\}$, and $\mathrm{n} = |\mathrm{A}|$.$X = \{\mathrm{x}_1, \mathrm{x}_2, \dots, \mathrm{x}_n\}$isa discrete random variable and its probability density is denoted as $p_i = \mathrm{P}(\mathrm{x}_i)$, and $\sum_{i=1}^{n} p_i = 1$. Thus the entropy of X is defined as $\mathrm{H}(X) = -\sum_{i=1}^{n} p_i \log_2(p_i)$. Assume $\mathrm{H_M}$is the maximum entropy, and $\mathrm{H_M} = \log_2(\mathrm{n})$. The information that an attacker can acquire is represented as $\mathrm{H_M} - \mathrm{H(X)}$. So the anonymity of a communication system is defined as

$$\mathrm{d} = 1 - \frac{\mathrm{H_M} - \mathrm{H(X)}}{\mathrm{H_M}} = \frac{\mathrm{H(X)}}{\mathrm{H_M}} \qquad (1)$$

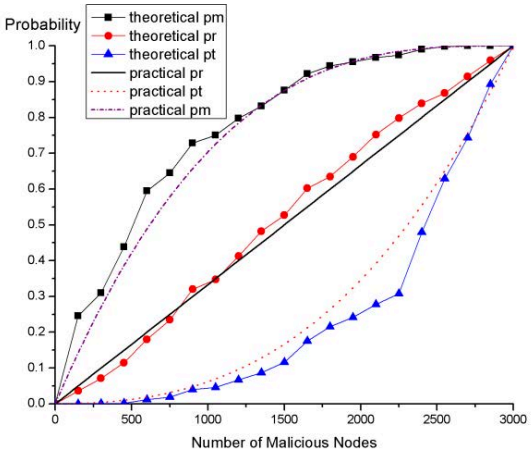


Figure 6.Evaluation of $\mathrm{p_m}$, $\mathrm{p_r}$ and $\mathrm{p_t}$.

Clearly, $0 \le d \le 1$, and *d=0* when there is only one element in the anonymity set. (Or, when certain p_i=1, the anonymity of the system is minimal, *d*=0.) While p_i=1/*N*, the system anonymity reach its maximum, *d*=1.

In an ideal Tor, every relay is treated as the same in a Mix-net system. Super nodes and normal nodes act in the same fashion for forwarding packets. However, as we demonstrated in the previous section, because super nodes can be identified by correlation, we have to consider them in a different category when evaluating the system anonymity, i.e., the existence of super nodes affect the anonymity, stability, and usability of Tor.

In the following, we reexamine system anonymity by considering the existence of super nodes. We divide the same anonymity set Aintoa super-node subset $A_u = \{\mathrm{a}_1, \mathrm{a}_2, \dots, \mathrm{a}_t\}$ and a normal-relay subset $A_o = \{\mathrm{a}_{t+1}, \dots, \mathrm{a}_n\}$. We have $\mathrm{n} = |\mathrm{A}|$, $n_u = |A_u|$, $n_o = |A_o|$, and $n = n_u + n_o$. X is a discrete random variable and its probability density is denoted as $p_i = \mathrm{P}(\mathrm{x}_i)$, and $\sum_{i=1}^{t} p_i + \sum_{i=t+1}^{n} p_i = \sum_{i=1}^{n} p_i = 1$.Thus the entropy of X is denoted as $H(X) = -\sum_{i=1}^{n} p_i \log_2(p_i) = -(\sum_{i=1}^{t} p_i \log_2(p_i) + \sum_{i=t+1}^{n} p_i \log_2(p_i)) = H_u(X) + H_o(X)$. Based on formula (1), the anonymity of the system with super nodes is denoted as

$$d' = \frac{H(X)}{H_M} = \frac{H_u(X) + H_o(X)}{H_M} \qquad (2)$$

Based on this new definition, without the knowledge of super nodes, the anonymity of Tor network is the same as before. However, when distinguishing super nodes from normal nodes, the system anonymity is reduced.

Assume we have a total of 3000 nodes in the network, similar to the number of daily nodes in the current Tor. Assume under the ideal conditions the same type of nodes has the same probability density, we have d=1. Now, let us consider the probability density of a super node is **R** times higher than that of a normal node where **R** is defined in Sec.3.3. We choose **R** = 2, 3, 4, 8, 16, 64, 256to compare d and d', as shown in Fig.5. Clearly, when nodes are divided into two types, the anonymity of the system is significantly reduced. (i) When the number of super nodes reaches a certain percentage in the system, the anonymity reaches the minimal value. For example, when **R**=4.0 and the percentage of super nodes is 30%, the attacker's chance increase from the ideal 0%to 3%. This means that the attacker has almost no chances before and now it has a good chance if it repeats the attack enough times. (ii) The ratio **R** heavily affects the anonymity. When **R**

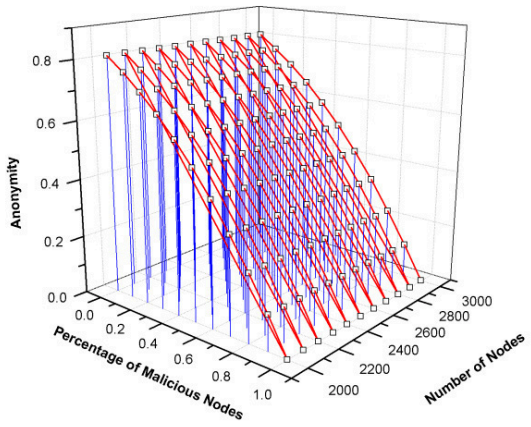


Figure 7.Anonymity under Malicious-Node Attack.

increases (the differences between a super node and a normal node become larger), the attacker will have much higher chances, e.g., from 0% to 10% when **R**=16.

We also consider the worst case in which attackers focus on super nodes only. In this case, the probability density of a normal relay is $p_i = 0$, where node i $\in A_o$, thus $H_o(X) = 0$. As the bottom curve shown in Fig.5, the lowest anonymity entropy reaches 70% (a 30% drop), a dramatic decline of system anonymity. Clearly, the existence of super nodes significantly damages anonymity. In the following, we will further examine anonymity under concrete attack models.

4.2 Different Attack Mode

When considering attacks exploiting the knowledge of super nodes, we examine several different situations as follows.

4.2.1 Brute-Force Attack on Service Availability

A brute-force attack may block (or deny the service of)a set of nodes in the system to make Tor unavailable, maybe initiated by a powerful attacker such as an ISP (or a government).Even with anti-censorship technology in which a user can connect to Tor by (semi-hidden) bridges, blocking (or DoS) super nodes is still a great challenge to the usability and stability of Tor. When focusing on super nodes instead of just any relays, these attacks can be more effective.

Let us first consider system performance. Assume we have ***s*** super nodes in a system of ***n*** nodes. Assume the bandwidth contribution of the whole system is ***B***. Then the overall contribution in a time interval ***T*** is ***C=T*B***. Let us denote the average bandwidth of super nodes as b_1, and the average bandwidth of normal nodes as b_2. So the contribution of super nodes is $C_1=Tb_1$, and the contribution of normal nodes is $C_2=Tb_2$, and $C=C_1+C_2$. Based on the analysis of super nodes in the Tor official data, the contribution made by super nodes is about 66% of the contribution of the whole system ($C_1/C = 66\%$), with only 21% of nodes are super nodes ($s/n = 21\%$).

Let us now consider the system anonymity and security. When the system suffers a brute-force attack without the knowledge of super nodes, the anonymity measure will decrease linearly with the number of nodes being attacked, based on formula (1). However, when attackers recognize super nodes, they aim at super nodes to achieve more damages, because Tor uses a bandwidth-weighted path selection mechanism. In general, the probability of node ***i*** being chosen for building a Tor path is $b_i/\sum_{k=1}^{n} b_k$, where b_i is the bandwidth contributed of node ***i***. So the influence of super nodes on anonymity is b_1/b_2 times than that of normal nodes. The maximum value of this ratio is 4 in the dataset we collected, which means the impact of attacking one super node equals to that of attacking four normal nodes. To maximize attack impact, attackers will aim at super nodes. In this case, $p_i = 0$, when i is a normal node; when i is a normal node, $p_i = \frac{1}{n_u}$, the maximum anonymity measure is obtained as $d_{max} = \frac{H_u(X)+H_o(X)}{H_M} = \frac{H_u(X)}{H_M} = \frac{\log_2 n_u}{\log_2 n}$. Thus, when super nodes are recognized, the system anonymity is reduced.

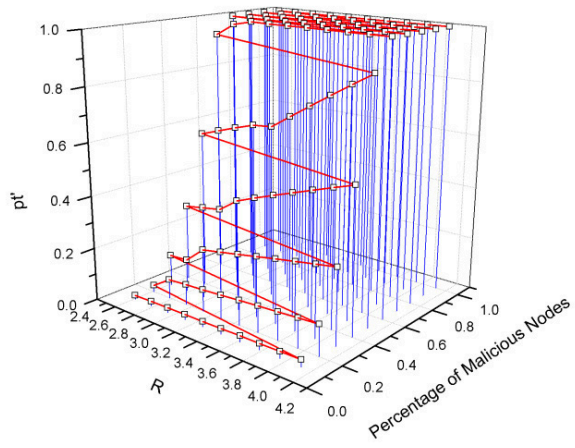
4.2.2 Malicious-Node Attack

If an attacker controls some nodes in the system, it can launch different kinds of complex attacks, including timing correlation attacks, disclosure attacks, predecessor attacks, collusion attacks, or Sybil attacks. We now focus on a simple malicious-node attack, in which an attacker compromises the first or the last relay node or both ends of a three-hop onion routing path.

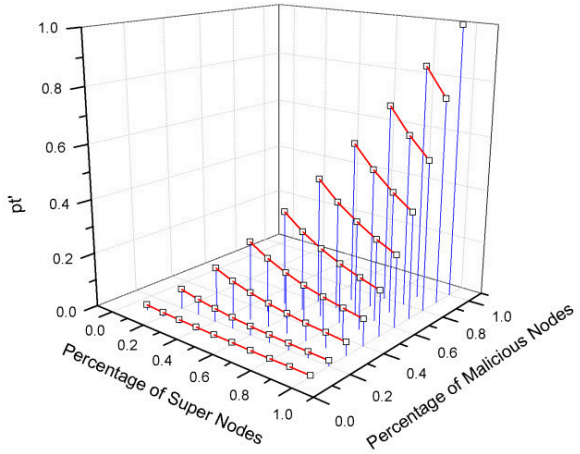
The position of a malicious node on a path is critical in this attack, because a node only knows its predecessor and successor on the path in Tor. When a malicious node is at the first hop of a path, the sender identity, its traffic patterns (e.g., packet size or timing), and other sender information is revealed. When a malicious node is at the last hop (acting as an exit node), the receiver and the plain-text traffic is exposed. When a malicious node is at the middle hop (the default path length is 3), only other relay nodes are revealed, and it will not affect the system security unless combined with other attacks. When an attacker controls both ends of a path, the sender, the receiver, and their relationship, are all exposed. Therefore, different positions on a path should be treated differently when calculating the anonymity measure under this malicious-node attack model.

Assume the system has ***n*** nodes, and ***c*** nodes out of ***n*** nodes are malicious nodes. The default path length ***L*** is three in Tor. The probability that attacker knows the sender is $p_r = c/n$. The probability that attacker only know the receiver is the same as p_r. As the probability that all three hops are all non-malicious nodes is $\frac{n-c}{n} * \frac{n-c-1}{n-1} * \frac{n-c-2}{n-2}$, we know the probability that a path includes at least a malicious node as $p_m = 1 - \frac{n-c}{n} * \frac{n-c-1}{n-1} * \frac{n-c-2}{n-2}$. We define the joint probability that the first hop and the last hop of a path are both malicious nodes as p_t. To find p_t, we need consider two cases: only the first and the last hop are malicious nodes and all three hops are malicious nodes. We have $p_t = \frac{c}{n} * \frac{c-1}{n-1} * \frac{n-c}{n-2} * \frac{P_2^2}{P_3^3} + \frac{c}{n} * \frac{c-1}{n-1} * \frac{c-2}{n-2}$, when malicious nodes are chosen randomly and regardless of the distribution of nodes. We present the theoretical analysis to p_m, p_r and p_t in Fig.6. We further generalize the anonymity measure formation for any path length **L** based on information entropy theory as

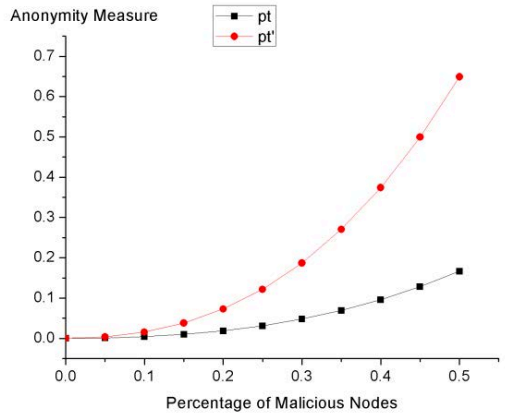
$$d'' = \frac{H(X)}{H_M} = \frac{n*\sum_{k=2}^{L} \frac{p_{n-c}^{k-1}}{p_n^k}}{\left(1+n*\sum_{k=2}^{L} \frac{p_{n-c}^{k-1}}{p_n^k}\right)*\log_2(n-c)} * \log_2 \frac{(n-c)*\left(1+n*\sum_{k=2}^{L} \frac{p_{n-c}^{k-1}}{p_n^k}\right)}{n*\sum_{k=2}^{L} \frac{p_{n-c}^{k-1}}{p_n^k}} + \frac{1}{\left(1+n*\sum_{k=2}^{L} \frac{p_{n-c}^{k-1}}{p_n^k}\right)*\log_2(n-c)} * \log_2(1+n*\sum_{k=2}^{L} \frac{p_{n-c}^{k-1}}{p_n^k}) \quad (3)$$



a.p'_t with fixed $\frac{s}{n} = 0.2$



b.p'_t with fixed $R = 2.61$



c.p'_tvsp_t with fixed R and$\frac{s}{n}$

Figure 8. The analysis to new anonymity measure p'_t.

In formula (3), H_M is the maximum entropy of the system, H(X) is the entropy of stochastic variable X, and k is the hop position where a malicious node appears on the path.

Using formula (3), the anonymity measure is shown in Fig.7. When there is no malicious node, the anonymity is about 0.8. The reason that the anonymity cannot reach 1 is because the default Tor path length is three. When L becomes larger, the anonymity will converge to 1 without malicious nodes. The clear drops in the figure show that the percentage of malicious nodes in the system heavily affects the anonymity measure. The total number of nodes in the system does not affect the anonymity measure very much when the system reaches a certain size. For example, the anonymity measure varies slightly as we increase the total number of nodes from 2000 to 3000 nodes in the tests. In the following, we will further examine the system anonymity under malicious nodes attacks which exploit the existence of super nodes.

5. SECURITY CONCERNS DUE TO THE EXISTENCE OF SUPER NODES

Attackers can discover super nodes as we did, and use the information to compromise the system anonymity by improving known attacks or developing new attacks.

5.1 Improving Known Attacks

With the knowledge of super nodes, the effectiveness of known attacks can be improved. To evaluate such impact, we first analyze its damages in this section and then evaluate it on a simulation platform in the next section.

5.1.1 Brute-Force Attack

When under a brute-force attack, attacking all nodes without differences is less effective because of the large number of relay nodes. However, attackers can focus their attacks on super nodes to damage the stability and usability of Tor, or filter all traffic passing through super nodes to reduce the availability of Tor. As mentioned in the above, the average bandwidth contribution of super nodes is about 66% of the whole system with only 21% of nodes. Therefore, attacker can effectively cripple 66% of network traffic by aiming at only 21% nodes. The attack is about three times more effective.

5.1.2 Malicious-Node Attack

Under malicious-node attacks, we use p_m, p_r and p_t to characterize the catalytic role of super nodes. Again, we have total n nodes in the system with s super nodes and c malicious nodes. Assume b_k is the bandwidth of node k, and b_1 represents the bandwidth of a super node and b_2 represents the bandwidth of a normal node. We define the selection factor of super nodes as $\frac{b_1}{\sum_{k=1}^{n} b_k}$, and the selection factor of normal nodes as $\frac{b_2}{\sum_{k=1}^{n} b_k}$. So a super node is $R = \frac{b_1}{b_2}$ times likely to be chosen in a path, as the *selection ratio of super nodes over normal nodes* discussed in Sec.3.3. Assuming the original bandwidth of each node is the same equal to 1, when the total bandwidth of the system is B=n, we have $b_1 = \frac{R*n}{(R-1)*s+n}$ and $b_2 = \frac{n}{(R-1)*s+n}$. Recall that $p_m = 1 - \frac{n-c}{n} * \frac{n-c-1}{n-1} * \frac{n-c-2}{n-2}$, $p_r = \frac{c}{n}$ and $p_t = \frac{c}{n} * \frac{c-1}{n-1} * \frac{n-c}{n-2} * \frac{P_2^2}{P_3^3} + \frac{c}{n} * \frac{c-1}{n-1} * \frac{c-2}{n-2}$ in the basic model of malicious node attack. Now, when attacks aim at super nodes, we focus on the impact on p_t. Because compromising the both ends of a path damages anonymity the most, we have

$$p'_t = {b_1}^2 * p_t = \left(\frac{R*n}{(R-1)*s+n}\right)^2 * \left(\frac{c}{n} * \frac{c-1}{n-1} * \frac{n-c}{n-2} * \frac{P_2^2}{P_3^3} + \frac{c}{n} * \frac{c-1}{n-1} * \frac{c-2}{n-2}\right).$$

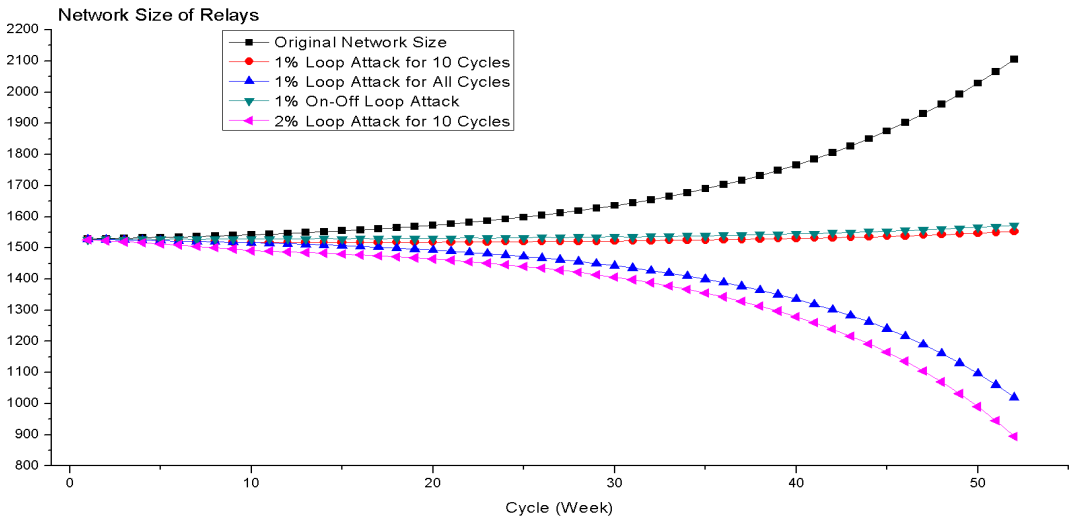


Figure 9.Analysis to Loop Attack.

When we fix the number of super node **s** as 600 in the total of **n**=3000 nodes, the result is shown in Fig.8(a). An attacker will prefer to compromise super nodes than normal nodes. When 900 nodes (30% of the total nodes) are *malicious nodes* (with 600 super nodes among them), the anonymity measurep'_t is about 0.3.

Meanwhile,p'_t becomes higher with the increase of **R**, while the original p_t is only related with the total number of malicious nodes. The range of **R** is between 2.6 to 4 from the real data presented in Sec.3.2. (However, **R** may be larger in many cases.) Here we set **R** to its minimum value of 2.6, and the results is shown in Fig.8(b). There are fewer points shown in the figure because we set the number of malicious nodes less than the number of super nodes to show the effect when *the attack focuses on super nodes only*. Both the percentage of super nodes and the percentage of malicious nodes have significant influences on p'_t. p'_t increases exponentially with the increase of the percentage of malicious nodes.

We further compare p_t and $p'_t$when $R = 2.6$and $s/n = 0.5$, as shown in Fig.8(c). When all super nodes are malicious, p'_t increases more rapidly than p_t, and reaches about 0.6, which is over three times more than p_t (at the same percentage of malicious nodes). In summary, the above simple analysis shows that, aiming at super nodes, malicious-node attacks can be improved significantly.

5.1.3 Other Attacks

Besides brute-force attacks and malicious-node attacks, attackers can control some super nodes and configure them as exit nodes. By capturing traffic of these exit super nodes, attackers intercept plain-text traffic, just like what Wikileaks did but more effective.

5.2 Potential New Attack

Here we introduce a new attack method, called *loop attack*, which is designed for attackers managing a large section of a network. This attack can amplify the knowledge of super nodes repeatedly to lower the perceptibility of users who are located in the network managed by the attacker. First, the adversary starts normal attacks to super nodes, e.g., just blocking them. The users may still use the Tor service with poor availability and performance. If the number of affected users is sufficient large, assuming the attacker is a large ISP or a government, the total number of Tor users in the network will decrease. Such a decrease leads to even worse performance, anonymity and availability of Tor, and then further lower the user confidence in the system and thus the number of users. With such a vicious cycle, the total number of users will reduce gradually. The attacker can also launch other attacks at the mean time to enhance the effect of this attack. Normal users are not aware of the attack; Tor operators also have a hard time to detect the attack when the statistics are changed gradually. So this attack is invisible at the beginning until its accumulative damage to some extent.

We model the effect of this attack as follows. The notation is summarized in Table 1. Assume the number of nodes following the simple iteration: $n_{t+1} = n_t + \alpha_t$, $\alpha_t = w_d * (d_t - d_0) + w_c * (c_t - c_0) + \beta$, where α_t can be either positive or negative, representing the increase or decrease of the number of nodes. It is affected by the user experience of Tor in both security and performance. Let $d_t$denotes the anonymity entropy of the system in cycle t, and $c_t$denotes the contribution of the system in cycle t. Let $d_0$denotes the threshold of anonymity of the system, and c_0 represents the threshold of contribution of the system, which are both calculated based on the initial data of the system. We use w_d and w_c asan anonymity weight and a contribution weight, which are used to balance their influences. The total number of nodes in cycle t is denoted as n_t, in which s_t are super nodes. The average bandwidth of super nodes is $b_1 = R * n_t/((R-1) * s_t + n_t)$, and the average bandwidth of normal nodes is$b_2 = n_t/((R-1) * s_t + n_t)$. Again, attackers prefer super nodes over normal nodes with a ratio $\boldsymbol{R=b_1/b_2}$. Thus the overall contribution of the system in cycle t is $c_t = b_1 * s_t + b_2 * (n_t - s_t)$. The estimated contribution determines system performance and user experience.

To evaluate the anonymity of the system in cycle t, we approximate the entropy as $d_t = \log_2 n_t$ in cycle t. We set the parameters based on the Tor official dataset of Year 2010.We obtain the values which fit the data well. We choose one loop iteration as one cycle, and consider one cycle as a week. There are 52 cycles in the data.

The effect of the new attack is shown in Fig. 9. We consider several different stealth attack strategies. (1) *1% loop attack* means that 1% of all nodes and 5% of super nodes are blocked by the attacker. By blocking the same number of nodes over only 10

Table 1.Parameters of Loop Attack

Parameter	Implication	Value
n_t	Network size in cycle t	-
α_t	Change of network size in cycle t	-
w_d	Weight of anonymity	40
d_t	Entropy in cycle t	-
d_0	Entropy threshold	10.5774
w_c	Weight of Contribution	0.04
c_t	Contribution in cycle t	-
c_0	Contribution threshold	1528
β	Const	1
R	Selection Ratio	2.6
s_t	Number of super nodes in cycle t	n_t*20%

cycles, the network size increases very little in one year (shown as the third curve from the top), compared to the original increase of several hundred nodes (the top curve). (2) *1% on-off attack* means that the attacker blocks 1% of all nodes and 5% super nodes in one cycle, and unblocks them in the next cycle, and so on. Although the results of 1% on-off attack (the second curve from the top) are similar to that of 1% loop attack over 10 cycles, the on-off alternative nature makes it hard to detect. (3) *1% attack over all cycles* means that the attacker blocks 1% nodes and 5% of super nodes over all the cycles. (4) *2% attack over 10 cycles* means that the attacker blocks 2% of nodes and 10% of super nodes. Clearly, case 3 and case 4 (the bottom two curves) cause the number of users drop significantly in 52 cycles. As we see, the loop attack on super nodes generates a vicious cycle in the system.

The main feature of this new attack is to utilize the knowledge of super nodes. The advantage of this attack is its concealment. Attackers only need to attack 1% or 2% of nodes in the system based on the knowledge of super nodes. Consequently the network size is reduced significantly, which serious damages the anonymity measure and performance of the system. Note that we use the minimum value of R=2.6 from our data collection. R could be larger in other cases, where the effect of this attack becomes more significant.

6. EXPERIMENT EVALUATION

6.1 Simulation Platform

We have evaluated our analysis with experiments set up in our lab. Because it is impossible to test our methods on the real Tor, we have built a simulation platform to simulate the entire Tor system, in order to evaluate the practical implication of super nodes on performance and anonymity. We simulate Tor algorithms with 3000 nodes. A node in the simulation platform behaves as the exactly same as a node in the real Tor. The only difference is that we have the complete control of each node to set up attacks and collect data. Besides relay nodes, directory servers are also simulated. The simulation platform can perform node joins, path selection, various attacks, etc. To the best of our knowledge, this is the first large-scale simulation platform of Tor, which has the same size as the current Tor. It is able to perform practical route operations as Tor, and initialize and observe attacks. This

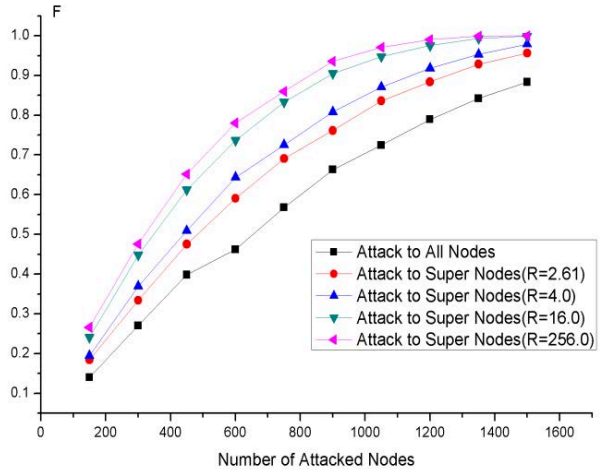


Figure 10.Results of Brute Force Attack.

simulation platform gives us more confidence in investigating Tor related issues.

There are a few limitations for our simulation platform. First, we do not perform the actually data encryption/decryption and transmissions over the network. The nodes only perform key features such as join, leave, path selection, etc. This simplifies the computation load, while not reflecting the real network complexity. For our investigation, network delivery does not affect our anonymity study. Second, the simulation system is driven by events such as the initialization/termination of connections. The important point is that these limitations do affect the investigation of super nodes in this paper. We can use the current simulation platform to examine brute-force attacks and malicious-node attacks. The simulated results reflect the same trend to the real Tor algorithms and setting. We are improving the simulation platform to support more features as the real Tor system for our future investigation.

6.2 Simulated Attacks

6.2.1 Evaluating Brute-Force Attacks

We have evaluated the effect of super nodes under brute-force attacks with 3000 relay nodes. In the first test, a brute-force attack aims at all nodes with the same probability, just as a normal brute-force attack without the knowledge of super nodes. In the second test, attackers aim at super nodes. The number of super nodes is set to 1500, half of the network size.

We define a *communication failure rate F* to evaluate the effect of brute-force attacks. When a node on path is attacked, we consider the path is disabled. We define *F* as the number of failed paths over the number of all paths. The test results are shown in Fig.10. The selection ratio of super nodes over normal nodes ***R*** is chosen to be 2.6, 4, 16, and 256 to show their effects on the failure rate. When the selection ratio ***R*** is 2.6 (the minimum seen in our dataset), the failure rate of attacking super nodes (the second curve from the bottom) is already much higher than that of attacking random nodes. In one extreme case, blocking all super nodes (half nodes of the system) makes the failure rate close to 100%. In another case, when attacking 150 super nodes (only 5% of all nodes in the system), the failure rate is already near 20%. Furthermore, when ***R*** becomes larger, the failure rate F rises rapidly (top three curves). For example, when attacking 750 super

nodes and R=16, the failure rate is higher than80%. This will clearly damage the system performance and the user experience.

6.2.2 Malicious-Node Attack

We have presented our analysi sis of p_m, p_r and p_t in Fig.6. We also use the simulation platform to verify the theoretical analysis. The results are also shown in Fig.6. From the comparison, the simulated p_m, p_r and p_t curves fit the theoretical analysis very well, and confirm that the theoretical analysis to p_m, p_r and p_t is accurate.

7. CONCLUSIONS AND FUTURE WORK

In this paper, we have confirmed the existence of super nodes in Tor through experiments and analysis. We have then revisited the basic anonymity measure, analyzed several attacks exploiting super nodes, and developed a new attack.

For our follow-up work, we will further investigate the practical and theoretical impacts of these attacks exploiting the existence of super nodes. We will also investigate mitigation solutions to address attacks exploiting super nodes, e.g., developing distributed lightweight trust and supervising mechanisms. Although the super nodes discussed here are Tor relays, such issues are also applicable to many other p2panonymous communication systems with similar properties. We will look into more general cases and conduct more theoretical analysis.

8. ACKNOWLEDGMENTS

Prof. Yingfei Dong's current research is supported in part by US NSF Grants CNS-1041739, CNS-1120902,CNS-1018971, and CNS-1127875. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of National Science Foundation.

This research is supported by the 973 National Basic Research Program of China (2007CB311102).

Funded by Tsinghua National Laboratory for Information Science and Technology (TNList) Cross-discipline Foundation.

9. REFERENCES

[1] Vasserman, E., Jansen, R., Tyra, J., Hopper, N., and Kim, Y. 2009. Membership-concealing overlay networks. In Proc. of the 16th ACM conference on Computer and Communications Security, New York, NY, USA.

[2] Nambiarand, A., and Wright, M. 2009. Salsa: A Structured Approach to Large-Scale Anonymity. In Proc. of the 13th ACM conference on Computer and Communications Security, New York, NY, USA.

[3] Dingledine, R., Mathewson, N., and Syverson, P. 2004.Tor: the second-generation onion router. In Proc. of the 13th conference on USENIX Security Symposium, Vol. 13, 2004.

[4] Tor Metrics, http://metrics.torproject.org/, 2011.

[5] BGP Table, http://bgp.potaroo.net/as2.0/bgptable.txt, Jan. 2011

[6] Tor directory protocol, version 3,http://tor.eff.org/svn/trunk/doc/spec/dir-spec.txt, 2010.

[7] Serjantov, A., and Danezis, G. 2002. Towards an information theoretic metric for anonymity. In Proc. of Privacy Enhancing Technologies workshop (PET'02), LNCS 2482, San Francisco, CA, USA, 2002:41-53.

[8] Díaz, C., Seys, S.,Claessens, J., and Preneel, B. 2002. Towards measuring anonymity. In Proc. of Privacy Enhancing Technologies workshop (PET'02), LNCS 2482, San Francisco, CA, USA, 2002:54-68.

[9] Reiter, M. K., and Rubin, A. D.1998. Crowds: Anonymity for web transactions, ACM Transactions on Information and System Security (TISSEC) 1(1), 66-92.

[10] Berthold, O., Federrath, H., and Köpsell, S.2001. Web MIXes: A system for anonymous and unobservable Internet access. Designing Privacy Enhancing Technologies: Design Issues in Anonymity and Observability. Springer-Verlag, LNCS2009, pp. 115-129.

[11] WikiLeaks and Julian Paul Assange. 2010. The New Yorker. http://www.newyorker.com/reporting/2010/06/07/100607fa_fact_khatchadourian?currentPage=all#ixzz0pWdlAepe.

[12] Hughes, D., and Shmatikov, V. 2004.Information hiding, Anonymity and Privacy: A modular approach. Journal of Computer security, 2004, 12(l):3-36.

[13] Shields, C., and Levine, B. N. 2000.A protocol for anonymous communication over the Internet. In Proc. of the 7th ACM conference on Computer and Communications Security, Athens, Greece, 2000:33-42.

[14] Chaum, D. 1981. Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms. Communications of the ACM, 1981, 4(2): 84-88.

[15] Wikileaks, http://en.wikipedia.org/wiki/WikiLeaks.

[16] Linode, http://www.linode.com.

[17] Li, F., Luo, B., Liu, P., and Chu, C. H. 2010. A Node-failure-resilient Anonymous Communication Protocol through Commutative Path Hopping. In Proc. of the IEEE INFOCOM, 2010.

[18] Fu, X. W., Zhu Y., Graham, B., Bettati, R., and Zhao W. 2005. On flow marking attacks in wireless anonymous communication networks. In Proc. of IEEE Inter. Conf. on Distributed Computing Systems (ICDCS), April 2005.

[19] Murdoch, S. J., and Danezis, G. 2006. Low-cost traffic analysis of tor. In Proc.of the IEEE Security and Privacy Symposium (S&P), May 2006.

[20] Ling, Z., Luo, J. Z., Yu, W., Fu, X. W., Xuan, D., and Jia, W. J. 2009. A New Cell Counter Based Attack Against Tor. In Proc. the 16th ACM conference on Computer and Communications Security, New York, NY, USA.

[21] Evans, N. S., Dingledine, R., and Grothoff, C. 2009. A practical congestion attack on Tor using long paths. In Proc. of USENIX Security Symposium, USENIX, Aug. 2009.

[22] D. M. Goldschlag, M. G. Reed, and P. F. Syverson. 1999. Onion routing. *Communications of the ACM*, 42(2):39–41, 1999.