Anti-RAPTOR: Anti routing attack on privacy for a securer and scalable Tor

Conference Paper · July 2015

DOI: 10.1109/ICACT.2015.7224775

3 authors:

Nguyen Phong Hoang
University of Chicago
30 PUBLICATIONS **303** CITATIONS
SEE PROFILE

Yasuhito Asano
Kyoto University
75 PUBLICATIONS **409** CITATIONS
SEE PROFILE

Masatoshi Yoshikawa
Kyoto University
384 PUBLICATIONS **4,797** CITATIONS
SEE PROFILE

Anti-RAPTOR: Anti Routing Attack on Privacy for a Securer and Scalable Tor

Nguyen Phong HOANG, Yasuhito ASANO, Masatoshi YOSHIKAWA

Department of Social Informatics, Graduate School of Informatics, Kyoto University, Japan

hoang.nguyenphong.jp@ieee.org, asano@i.kyoto-u.ac.jp, yoshikawa@i.kyoto-u.ac.jp

Abstract—**Regardless of Tor's robustness against individual attackers thanks to its distributed characteristics, the network is still highly vulnerable to those very powerful adversaries, such as oppressive regimes which have control over a large proportion of the Internet. As recently confirmed by Edward Snowden, Autonomous-System level adversary is no longer theoretical, but poses a real danger to the Tor network. Therefore, through this research, we strive to propose an improved design in Tor to against the most contemporary de-anonymizing attack techniques, especially RAPTOR: Routing Attacks on Privacy in Tor. Different from most previous works, the scalability aspect of the overall Tor network is also taken into consideration in this study since the number of both end users and voluntary relays is foreseen to keep increasing in the next coming years. To against RAPTOR, we suggest that an Internet AS-level topology file should be periodically maintained and distributed by Directory Authorities. The file is only fetched by the guard and exit relays in addition to the conventional consensus network status document to preserve the scalability of the network. The user then decides to initiate her anonymous circuit based on the result of the intersection between two sets of ASes: the set of ASes *between the user and the guard relay*, and the set of ASes *between the exit relay and the final destination*. The paper concludes by summarizing pros and cons of the proposed design from various points of view including the Directory Authorities, the voluntary relays and the end users; and suggesting future works that are necessary for a state-of-the-art anonymity technique.**

Keywords— **Privacy, Anonymous Communication, Tor, Autonomous System**

I. Introduction

In recent years, the problems of censorship and surveillance in cyberspace have become more and more serious. Even in areas with a long history of freedom of speech like America and Western Countries, the government surveillance activities are happening day to day [33], [36]. According to the Guardian, the NSA has a program called Marina which can track and store the online metadata of millions of internet users for up to a year [29], while the Britain's GCHQ has secretly gained access to the networks of cables which transmit the world's phone calls and internet traffic [30]. Moreover, as recently confirmed by Edward Snowden, Autonomous System level adversary is no longer theoretical, but poses a real threat to internet users' privacy [32]. For that reason, anonymous communication has drawn remarkable attention from both researchers and ordinary internet users as they grow to support more and more users with various applications to protect their privacy [3], [34]. Despite the existence of many pro-privacy and anti-censorship tools that are freely available on the Internet such as proxy servers, Virtual Private Network (VPN) software and so on, The Onion Router (Tor) [16] has been proved to be the most robust anonymous communication tool [13], [33].

In this paper, we start by providing background on Tor's recent design and path selection algorithm in section II. Next, the problems of scalability and Autonomous-System (hereafter: AS) level adversary in Tor are comprehensively analyzed with real-world data in section III. We then demonstrate a new design in Tor network to help Tor user against routing attack on privacy, and implement it in a manner that preserves the scalability for the overall Tor Network in section IV. Finally, conclusion about the proposed design from various points of view and necessary future works are declared in section V and section VI.

II. Background of The Onion Router

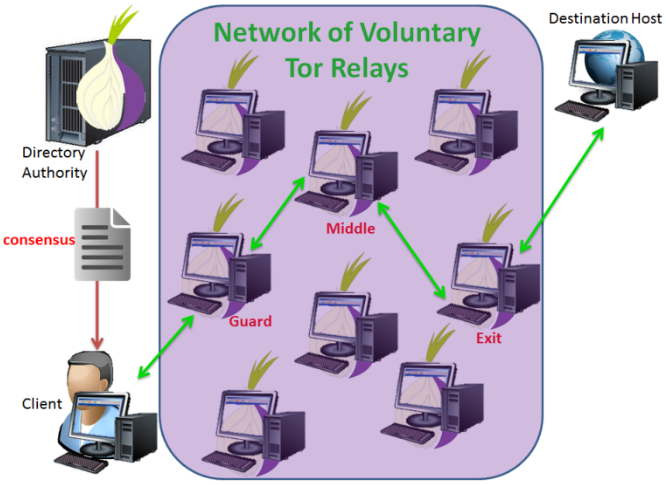

Figure 1. System Architecture of the Tor network

Tor is originated from onion-routing [5], which was primarily developed by U.S. Naval Research Laboratory to use within the American Military. In general, Tor network has two main elements: **Directory Authorities**, which are centralized trusted servers keeping track of the whole Tor network, **and relays**, *aka routers or nodes*, which are voluntary computers distributed around the world, and can be categorized into bridge, guard, middle and exit relays based on their functionality and position in the virtual circuit.

As shown in Figure 1, Tor user's software (*aka client* which can be a Tor Browser Bundle [18], Tails [19], or any interface that talks to the Tor network) initially fetches a consensus "network status" document containing all Tor relays' information from hard-coded Directory Authorities. Before being sent to destination host, data packet is multiply encrypted and forwarded through a virtual circuit made up of several relays using Onion Routing [2], [4]. (Ideally three relays for direct users and more in case of bridge users [20]). At each relay, the packet is decrypted one layer to reveal the next relay address until the packet reaches the final destination. The whole mechanism is similar to the onion-peeling-off process so that each relay only knows the previous and the next relays. By this way, without traffic analysis, none of the relays in the virtual circuit can correlate the original sender and the destination.

In order to improve both performance and overall traffic balance in Tor network, each client selects relays in proportion to their bandwidth to form the virtual circuit in recent version of Tor. For instance, given total number ***N*** of routers in the network and the bandwidth $\boldsymbol{b_i}$ provided by router ***b***, the probability in which the router ***b*** is chosen is calculated as:

$$\frac{b_i}{\sum_{k=1}^{N} b_k}$$

III. Problem Analysis In Tor Design With Real Data

In this section, we analyze recent problems in detail, and introduce important related works. Unlike previous studies that addressed the problems in theoretical ways, we point out the problems based on real data so that the reader can acquire the most complete and comprehensive understanding about the remaining problems in recent Tor design.

A. Scalability

In order for Tor to function properly, all Tor clients try to continually maintain a live consensus network status document containing information about all relays that is necessary for the client to build the anonymous circuit. As defined in Tor Directory Protocol, a network status document is "live" if the time in its valid-until field has not passed [27]. At the time of writing, Tor Directory Authorities publish a new consensus network status document every one hour, and that consensus is valid for three hours, which means all Tor clients try to update a new consensus network status document every few hours (less than 3 hours) to keep their information about all available relays fresh.

Nevertheless, McLachlan *et al.* had foreseen that Tor network would be spending more bandwidth on serving the consensus network status document because of the dramatic increase in the number of both relays and clients [10]. In more details, given n clients and r relays in the Tor network, then the bandwidth cost for distributing the consensus network status document to all clients in recent Tor design is estimated as $O(nr)$, while the bandwidth available for the whole network only increases as $O(r)$.

For a more comprehensive view of how the number of clients and relays impact on the bandwidth cost for serving the consensus network status document, past data of direct users and relays were retrieved from Tor Metrics [17] and plotted in the Figure 2 and Figure 3.

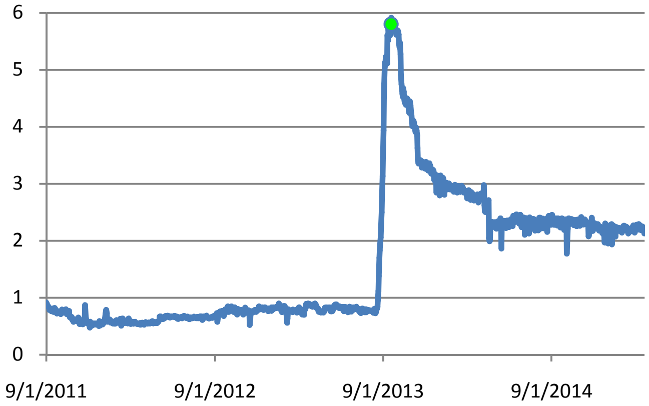

Figure 2. Daily direct users (in million)

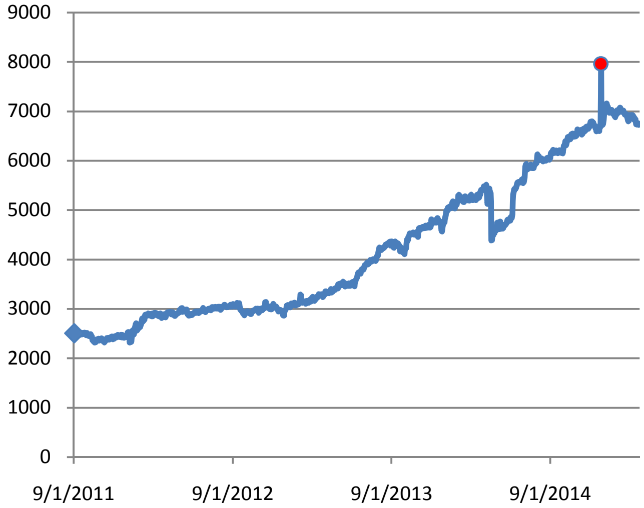

Figure 3. Relays in Tor network over time

As shown in Figures 2 and 3, the number of users was doubled, while the number of relays has increased roughly three times in the last 5 years. As highlighted by the green node in Figure 2, around the end of August 2013, Tor was under a coordinate attack form a bot twice the size of the regular Tor network. That is the reason why there were about 6 million daily users at that moment [31]. However, Tor was still usable at that time, but could not avoid wasting a huge amount of bandwidth (approximately 18 TB was the peak of the attack as also highlighted by the green node in Figure 4) on answering the directory requests (*aka request to download the consensus network status document*), and some relays were saturated by the amount of connections and circuits that they had to handle. In addition, during the period of December

2014, there was a Sybil attack [7] to Tor, in which a large number of small exit relays were created as highlighted by the red node. It was detected immediately [35], but still caused increasing in the size of the consensus network status document as highlighted by the red nodes in Figures 3, 4 and 5.

Figures 4 and 5 show how sensitive the cost of bandwidth for serving the consensus network status is, particularly to the change in the number of clients and relays in Tor network.

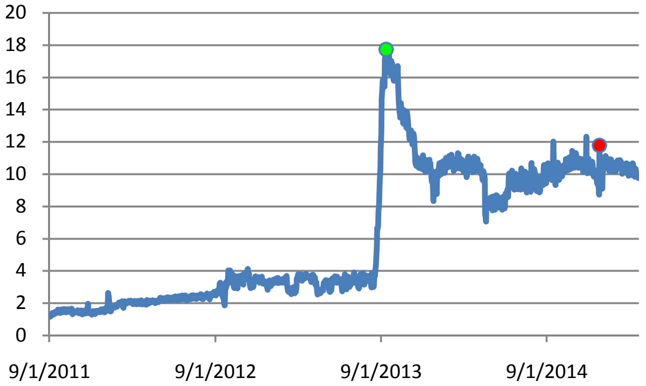

Figure 4. Bandwidth spent on serving directory request (in TB/day)

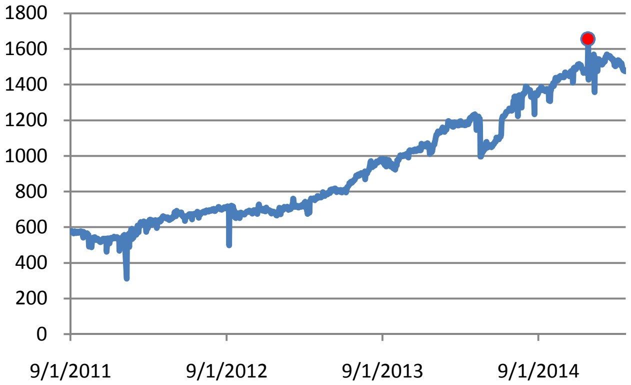

Figure 5. Size of the consensus network status document (in KB)

It is the fact that data for Figure 4 and Figure 5 are not readily available. Therefore, we first fetch raw data of *number of bytes spent on answering directory requests* from Tor Metrics [17] and *past data of relay descriptors* from Tor Collector [21]. Then, some calculations and parsing procedures were conducted to retrieve necessary data for plotting the graphs. By observing Figures 2 through 5, there are obvious correlations in the increasing pattern between Figures 2 and 4, and between Figures 3 and 5, while Figures 3 and 4 have similar pattern during the period from March 2014 until March 2015. As a result, we can intuitively conclude that both number of clients and relays have effect on bandwidth cost for serving the consensus network status document. However, number of clients has more obvious effect on the bandwidth cost. In other words, number of clients is the most sensitive and intimate factor with bandwidth cost compared to other factors. To that end, there are four patterns which can probably happen in the future as shown in Table 1.

TABLE 1. Possible changing patterns of bandwidth cost for serving the consensus network-document in Tor network

	Clients	Relays	Bandwidth cost for serving the consensus network-document
1	decrease	decrease	decrease
2	decrease	increase	decrease more likely happens than increase
3	increase	decrease	increase more likely happens than decrease
4	increase	increase	dramatically increase

As it is foreseen that the number of Tor client will keep increasing in the next coming years due to deep concerns about censorship and surveillance activities in cyberspace [34], so the last two highlighted patterns will more likely to happen. Hence, future works need to take the scalability into careful consideration and restrain from unnecessarily increasing the size of the consensus network status document or requiring clients to download more consensus documents, because most of the Tor's relays are voluntarily operated [22], or run by limited donation [23], thus necessitating saving even a small amount of bandwidth. We will discuss further about our proposal in terms of scalability aspect in Section IV.

B. RAPTOR: A Real Threat from AS Level Adversary

In this part, real-time data is analyzed to pinpoint which part of Tor is the most vulnerable to AS-level adversary.

It is worth noting that different from other latency-tolerance anonymity techniques such as Mixminion [1], which introduces delays into packet to against a very powerful adversary, Tor strives to provide low-latency anonymity especially for interactive application, such as web surfing. As an inevitable aftermath, an adversary with ability to monitor traffic at two ends of the Tor communication (i.e., the channel between the client and the guard relay, and the channel between exit relay and final destination) can deploy traffic analysis technique to correlate the size and time of transmitted data packets to deanonymize Tor clients [6], [9].

Edman and Syverson had found that a single AS was capable to monitor 39.4% of the randomly generated Tor circuits [11], even greater than the former result of Feamster and Dingledine in [8]. As a result, the authors in [11] have shown that the dramatic growth in number of both voluntary relays and clients does not effectively alleviate attacks from adversarial ASes.

One of the primary reasons is because Tor relays are volunteer-based and not evenly distributed among ASes. At the time of this study, we again analyzed the AS-diversity in Tor. Examining the consensus network status document retrieved at 1200 UCT on March 27th 2015 by using Stem Python library [24], we found that although there are 6697 relays in the network, there are only 1220 unique ASes [1]. Moreover, there are few ASes that dominantly occupy a remarkable number of relays in Tor network as shown in the Table 2.

[1] We map IP to AS by http://ipinfo.io/ which uses MaxMind's datasets [37] to provide IP look up service.

TABLE 2. Top 10 ASes which contain most of the relays

ASN	Accumulative BW	% BW	total unique relays	% relays
16276	3592823	11.65	401	5.99
24940	2507641	8.13	302	4.51
3320	74010	0.24	194	2.90
7922	44446	0.14	178	2.66
12876	3125003	10.13	146	2.18
6830	59479	0.19	124	1.85
20473	121342	0.39	80	1.19
701	105370	0.34	78	1.16
36351	61310	0.20	77	1.15
12322	66531	0.22	68	1.02
Total	9757955	31.64	1648	24.61

Summing up the result provides us an insight that more than 30% of bandwidth (KB/s) and nearly 25% of relays are allocated within the top 10 ASes, thus potentially raising the vulnerability and attracting adversary to exploit these ASes.

Actually, threat from AS-level adversary has been theoretically discussed and predicted for a long time ago [8], [11], [12]. However, Sun *et al.* has recently introduced RAPTOR [15], which is a suite of attack techniques, can be launched by an AS-level adversary to compromise user's anonymity. In more details, RAPTOR encompasses three small attack fashions:

1. *Exploiting the asymmetric nature of the Internet to conduct traffic analysis and correlation attack at both ends of the communication*
2. *Exploiting the dynamics in the Internet Topology to increase the probability that a particular adversarial AS can appear simultaneously on both ends of the communication*
3. *Manipulating Internet Routing Policy through BGP hijack (to locate guard relays used by the client) and interception (to perform traffic analysis)*

The authors in [15] have tested RAPTOR in real Tor network, and succeeded in compromising user's anonymity (note also that: with ethical considerations, their experiments were tested within self-created Tor relays, and no users' privacy was harmed).

In spite of conducting successful real attacks on self-created relays in the research, Sun *et al.* only use some samples and self-created AS topology graphs, which does not reflect reality of the Tor network, to demonstrate their attacks. Therefore, to help the reader of this paper gain a comprehensive insight into the reality of the circumstance, we use real data from recent Tor to point out at which AS that threat like RAPTOR could potentially happen.

Since previous works often focus on those ASes that occupy a dominant amount of relays, a similar version of analysis like Table 2, in which ASes are sorted in descending order of number of relays allocated within those ASes, could probably has been seen before. However, let us argue that it is unavoidable and natural for such ASes to see a relatively large view of the overall Tor network since the original Internet is unevenly distributed. Unlike other works, we additionally analyzed the real-time data from Tor's consensus and sorted the Table 3 in descending order of accumulative bandwidth of all relays in each AS. We then found an anomaly that there are 7 ASes that did not appear in Table 2, but appeared in Table 3 with comparatively high ranks. In other words, there is a paradox that some ASes contain very few relays, while covering a striking amount of bandwidth of the overall network. With the top remaining 3 ASes, these 10 ASes cover roughly 50% of total Tor network's bandwidth.

TABLE 3. Top 10 ASes which cover a large proportion of bandwidth

ASN	Accumulative BW	% BW	total unique relays	% relays
16276	3592823	11.65	401	5.99
12876	3125003	10.13	146	2.18
24940	2507641	8.13	302	4.51
8972	1568756	5.09	60	0.90
24961	1364022	4.42	54	0.81
60781	805707	2.61	57	0.85
37560	599144	1.94	3	0.04
43350	527741	1.71	10	0.15
51395	520938	1.69	22	0.33
13030	506128	1.64	19	0.28
Total	**15117903**	**49.02**	**1074**	**16.04**

Owing to recent Tor's bandwidth-based path selection algorithm, this group of ASes is believed to be more attractive to AS-level adversary. Especially, those ASes do not occupy a large number of relays, but cover a significant amount of bandwidth. For example, AS37560 contains only 3 relays (46.246.46.27, 46.246.32.223, 197.231.221.211), but occupy nearly 2% of the whole network's bandwidth. The three relays' information is summarized in Table 4:

TABLE 4. Relays in AS37560 (F: Fast, G: Guard, E: Exit, C: Country)

IP	BW	F	G	E	C	multi origin ASN
46.246.46.27	26000	●	●		SE	42708
46.246.32.223	144	●		●	SE	42708
197.231.221.211	573000	●	●	●	LR	

Although AS37560 was not listed in Table 2 and only at rank #7 in Table 3, but with just 3 fast guard/exit relays, its probability to be selected as guard and exit is even definitely higher than AS3320 (rank #3) and AS7922 (rank #4) in Table 2. We use Compass [25] to validate this conclusion. At the time of writing this paper, the total probabilities of being chosen as guard/exit are 0.14%|2.63%, 0.02%|0.02% and 0.00%|0.06% for AS37560, AS3320 and AS7922 respectively.

Since without any apparent evidence, we do not come to an intuitive conclusion that any of those relays or AS37560 is an adversary in this case. However, due to recent Tor's design (i.e., the bandwidth-based relay selection algorithm and choosing relays which do not belong to same /16 subnet), the one that have control over some particular ASes, such as ISP or government agency, can have no difficulty to figure out an attack fashion by following these steps:

1. Locating ASes which have distinctly fast guards/exits, but do not occupy many relays (e.g., AS37560, AS43350)

2. Injecting another fast exit/guard, which does not belong to the same /16 subnet with the guard/exit detected in step 1, into the same AS

3. Snooping at AS-level to conduct traffic analysis with low-cost since there are few relays in the anonymity set located in that particular AS

By this way, the adversary even does not need to conduct BGP hijacking or intercepting attacks to monitor the traffic as mentioned in RAPTOR.

Despite numerous proposals such as configuring Tor client to geographically pick up relays located in different countries [11], our real-data analysis shows that the three relays are located in different countries regardless of being operated under one AS. Moreover, Sweden (SE) and Liberia (LR) are even two far away countries. Therefore, such proposals are not effective and can still be disabled by AS-level adversary if the client uses one of the followings sample paths:

- (46.246.46.27, any middle relay, 197.231.221.211)
- (197.231.221.211, any middle relay, 46.246.32.223)

Next, for the purpose of completeness, we keep using the case of AS37560 as real data to demonstrate how the interdependent and dynamic Internet topology (#2 RAPTOR attack fashion) and BGP intercepting (#3 RAPTOR attack fashion) could increase the ability of an AS to monitor more Tor's traffic.

To visualize AS37560 in Internet Topology, we investigated the latest dataset of AS relationship made available by CAIDA [38], and obtained the graph in Figure 6.

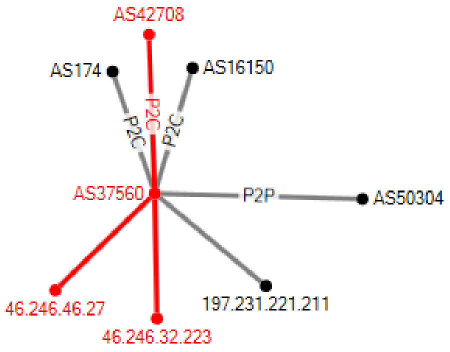

Figure 6. AS37560's Internet topology Visualization

We found that AS37560 is a customer AS of other 3 provider ASes (AS174, AS42708 and AS16150) and has AS50304 as a peer AS. That means the three provider ASes are the main gate for traffic to traverse from/to AS37560 to/from the entire Internet. For that reason, threat (*if any*) of correlation attack posed by AS37560 is also propagated to these 3 provider ASes, thus the total probabilities of being chosen as guard/exit ASes should be propagated as well since they also monitor traffic of AS37560. We notice that recent design of Compass [25] does not take this circumstance into account. Therefore, let us approximately re-compute the total probabilities of being chosen as guard AS and exit AS for these 3 ASes in term of threat-propagated-awareness as shown in Table 5, in which probabilities of being chosen as guard AS and exit AS for each provider AS should be increased by ***x*** smaller than guard/exit probability of customer AS37560.

TABLE 5. Re-computing probability of being-chosen-as-guard/exit at AS-level due to interdependent nature of the Internet Topology

AS	No. Relays	Guard Probability	Exit Probability
37560	3	0.14	2.63
174	9	0.73 + (x < 0.14)	0 + (x < 2.63)
42708	15	0 + (x < 0.14)	0.02 + (x < 2.63)
16150	0	0 + (x < 0.14)	0 + (x < 2.63)

Because of the complexity in routing policy of the Internet, and traffic allocation between ASes is confidential information in contract between ISPs, we cannot really know how much traffic is allocated among these 3 provider ASes. What we can be sure is that if one of these 3 ASes went down due to infrastructure maintenance, corruption or being under DDOS attack, the other remaining two ASes can see more traffic of AS37560. And if two of them are down, the left one could completely monitor traffic of AS37560. That is how the interdependent and dynamic Internet topology could increase the ability of an AS to monitor more Tor's traffic.

Next, by mapping IPs to ASes in Table 4 with Hurricane Electric's BGP toolkit [26], we found that the first two IPs are announced by two ASes: AS37560 and AS42708 in the last column. However, only AS37560 with more specific advertised prefix 46.246.32.0/**19** is retrieved by https://ipinfo.io/ in Tables 2 and 3 instead of AS42708 with less specific advertised prefix 46.246.0.0/**17**. And, we then found that AS42708 is provider AS of AS37560 in Figure 6.

At the time of retrieving data from Compass, there was no Tor relay operated under AS16150. However, as discussed in RAPTOR's attack fashion #3, AS16150 can totally monitor by maliciously advertising a more specific prefix than AS42708, such as 46.246.0.0/**18**. In this case, the traffic is still redirected to its correct destination, so AS16150 would successfully snoop on Tor's traffic from the entire Internet to AS37560. BGP hijacking can also be conducted in the similar manner with BGP intercepting, but will not last for a long time since hijacking BGP is just a black hole which does not send response packet to the original sender, thus communication will be eventually dropped. That is a scenario in which how an AS-level adversary can abuse BGP intercepting attack to increase its ability to sniff Tor's traffic.

IV. Anti-RAPTOR

Taking into account of the scalability and AS-level adversary problems discussed with real Tor data above, we propose anti-RAPTOR as a countermeasure to the AS-level attack. Since we have shown in section III.A that the increase in the number of clients directly and intimately affects the increase in the bandwidth cost for serving the consensus document, and because the recent scalability problem has not been efficiently handled, our proposal is designed in a manner that we try not to introduce more unnecessary load to the network, especially load relating to the number of clients, which probably will cause additional scalability problem.

Although RAPTOR is referred in this study as the newest attack technique on privacy in Tor which was published on March 13th 2015, our proposal is not just an instantaneous countermeasure to this attack. It has been studied and planned as a long term solution after Edward Snowden confirms the NSA has an autonomous system called MonsterMind [32], which is allegedly said to be able to intercept network traffic flows. Therefore, our proposal is not only a particular reaction to RAPTOR, but rather a long-term planned countermeasure to against AS-level adversary in general.

A. Design

First of all, it is important to note that Tor is operated by volunteer-based relays, and because of the nature of the Internet, plus different policy between ISPs and governments, we cannot command where volunteers should run their relays. Therefore, preventing the AS-level vulnerability like the particular case in real-world discussed in section III.B is not really a trivial task. In order to neutralize such threat from AS-level adversary, we try to improve the procedure in which the client should initiate virtual circuit in an AS-awareness manner, and suggest some necessary changes in the Tor network as illustrated in Figure 7.

AS_FCG: *ASes set in forward link between client and guard*
AS_RCG: *ASes set in reverse link between client and guard*
AS_FED: *ASes set in forward link between exit and destination*
AS_RED: *ASes set in reverse link between exit and destination*

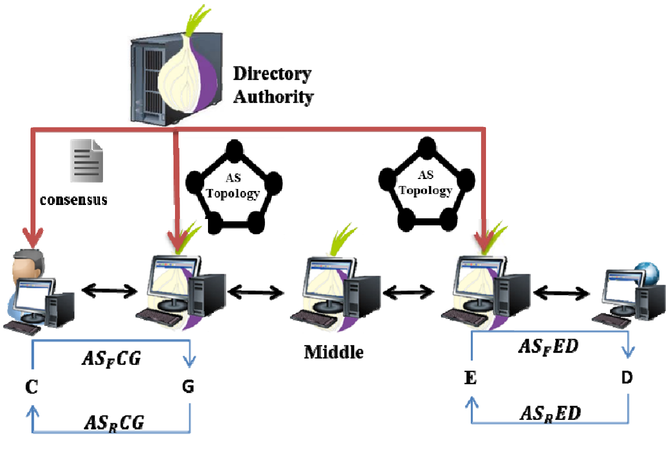

Figure 7. System Design of Anti-RAPTOR

1) AS-Awareness

With Anti-RAPTOR, we propose that an AS Topology is maintained and distributed to guard and exit relays by trusted central Directory Authorities. It is a directed graph of the Internet AS-level topology which will be used by the guard and exit relays to look up AS_FCG, AS_RCG, AS_FED, and AS_RED. In the current Tor's design, circuits are built based on the bandwidth-preferred algorithm and choosing relays which do not belong to the same /16 subnet. However, as shown in section III.B, these constraints are not really efficient to against AS-level adversary. Therefore, client has to pick up circuit in an AS-awareness manner to avoid using circuit in which a same AS appears at both ends of the communication. Hence, before really initiating a particular circuit, client sends two requests to check AS paths between herself and the guard, and between the exit and the destination at the same time when she sends request to open the virtual circuit. After receiving requests, the guard and exit look up on the AS Topology file to parse AS_FCG, AS_RCG, AS_FED, and AS_RED and send back to the client. Basing on retrieved ASes sets from the guard and the exit, the client implements this algorithm:

```
AS-awareness algorithm
if:
    $(AS_FCG \cup AS_RCG) \cap (AS_FED \cup AS_RED) = \emptyset$
then initiate circuit
else drop
go to next circuit in the list of pre-emptively built circuits
```

2) Scalability

Our approach is quite similar to LASTor [12] in term of AS checking procedure. However, LASTor requires client to initially download 13 MB of AS-related data including inter-AS link, AS three-tuples, and AS path lengths to against AS-level adversary. Nevertheless, AS topology is very dynamic, thus maintaining such files from client side in order for client to properly infer a near-to-real-time AS path is not an easy work, and may cause more loading overhead in a network which has a huge number of clients like Tor. On the contrary, our design only requires guard and exit relays to download the AS topology and update it when Directory Authorities publish a new one, instead of all clients. One may question that the size of response packet from guard and exit relays to the client would add more load to the Tor network. However, we have analyzed the AS Topology from CAIDA [38], and found that the longest path within the Internet is 11-AS long, and the average path is 4-AS long. It means the largest response is the packet that contains 11 ASes, and in average the response packet contains only 4 ASes. With that relatively few information, we believe that the response packet containing ASes numbers will not cause serious loading overhead to the Tor network. Additionally, to preserve the recent Tor's approach of improving performance by using bandwidth-based algorithm, we apply directly our algorithm on the top of the list of pre-emptively built circuits. From top to bottom, all circuits are checked before being initiated.

B. Discussion

In a compatible manner with proposal of Dingledine *et al.* [14], if the guard is stable and used for a long period of time in the future Tor design, then cost for computing $(AS_FCG \cup AS_RCG)$ can be reduced, since the client just need to request the guard to report $(AS_FCG \cup AS_RCG)$ for the first time, and reuse the result until a configurable time has passed.

In [15], the authors suggest that client should chose a closer guard relay to lower the possibility that an adversarial AS will appear in the channel between herself and the guard. However, such approach could potentially make the client distinguishable, thus a "shorter" AS path still need to be carefully researched. As we just mentioned that average distance between two arbitrary ASes is 4, future work may take this number to estimate how many ASes in a path would be the most efficient, and after how long the $(AS_FCG \cup AS_RCG)$ should be updated depended on the length of AS path between the client and the guard relay.

Next, the authors in [15] also recommend that Tor should advocate its voluntary relay operators to run Tor relays with a prefix longer than /24 to against BGP hijacking and intercepting. However, as mentioned above, due to different policy between ISPs and governments, we cannot command where volunteers should run their relays. Furthermore, it is widely known in the networking community that BGP hijacking and BGP interception happen when there is an AS try to advertise a more specific prefix for various ill intensions. In other words, when such kind of attack occurs, the entire Internet would probably see two different ASes announce a common range of prefixes. However, this judgment is not always correct, because while multi-origin routes could be an indication of BGP hijacking, it can also due to an organization not having their own AS, so they have many ISPs announce their IP space for them, or possibly a case of anycast usage but from various ASes. By observing BGP toolkit provided by Hurricane Electric Internet Service [26], we found that the daily-updated multi-origin routes include 6600 unique prefixes which are advertised by 14000 different ASes, half of the prefixes are /24 subnets. Therefore, if we just drop any circuit based only on the AS testing result above without carefully taking this circumstance into account, we would waste Tor's limited resource, since the phenomenon of multi-origin route is happening every day. To that end, we would suggest that central trusted Directory Authorities also observe the multi-origin routes and blacklist those ASes which advertise a same prefix for the first time, and un-blacklist them after a "trustable" period of time has passed. That set of blacklisted ASes should be distributed together with consensus (but should be not activated by default and only for those clients who opt for AS-aware path selection algorithm). Additionally, we can also take completely advantage of Bandwidth Scanner Authorities to detect abnormal change in Internet AS-level topology while sending packet to guard and exit relays to test their bandwidth.

It is vital to mention that threat to our design occurs only when both of guard and exit relays try to strategically, systematically and simultaneously lie about the AS paths of AS_FCG, AS_RCG, AS_FED, and AS_RED, so that the client cannot recognize that she is going to use a circuit in which a same adversarial AS may appear at the two ends. Nevertheless, thanks to the conventional wisdom of the three relays in Tor circuit, the guard relay cannot know which exit relay it is contacting to, and vice versa. As a result, the scenario that both guard and exit relays know that they are handling a same circuit and provide false response about AS paths is difficult to happen.

V. Conclusion

In this paper, we have comprehensively analyzed and discussed in detail about recent scalability and AS-level adversary problems with Tor's real-world data. Through the statistical Tor data, we have shown that the number of clients is the most associative and intimate factor with bandwidth cost among many other factors. In addition, based on previous theoretical discussion about AS-level adversary, we used real-time data to demonstrate a particular case in which the AS-level adversary can really post real threat to the users' privacy.

Recently, RAPTOR is introduced and classified into three small attacks by exploiting many aspects of the nature of the Internet and recent Tor's design, but they are all basically depending on the vulnerability from AS-level adversary. Therefore, it can be generally reviewed as an attack in which adversarial AS tries to exist at two ends of the Tor's anonymous communication. Based on that property, the paper then proposed anti-RAPTOR as an efficient and effective way to help Tor clients protect their privacy, while not introducing more scalability problem to the Tor network.

Our proposal is expected to help Directory Authorities reduce cost for distributing the AS topology to a huge number of clients. Instead, the file is distributed only to guard and exit relays, and those relays also do not need to spend a large amount of bandwidth to serve the whole file to the clients, since they only response by sending a relatively small packet containing ASes information upon request from the clients. Anonymity is prioritized in Tor, but not all Tor clients really need to be completely anonymous. Some of the clients just need to overcome the censorship of local authority to reach the external Internet, and would prefer performance but not anonymity. Therefore, anti-RAPTOR should only be enabled and used by those who really care about their privacy while being online, instead of being applied widely by default.

VI. Future work

Although dataset from CAIDA is used in this paper, there are still many projects that provide accurate AS topology datasets such as AS-topology powered by Cyclops [39], iPlane [40], and so on. Therefore, at the time of writing this paper, we have started working on analyzing those Internet AS-level Topology datasets to find out the most suitable one to adopt in Tor so that a near-to-real-time AS path is guaranteed. Additionally, the procedure of modifying Tor's source code to simulate our proposal in Shadow [28] has been taken place.

Finally, through this paper, we want to address that threat from AS-level adversary is really urgent, and need more attention from Tor researchers and developers. We notice that Tor project has some sub-projects to analyze ASes such as Compass, but still at statistic level. Therefore, we suggest that future implementation should take both scalability and AS-level adversary into careful consideration.

ACKNOWLEDGMENT

This work was supported by JSPS KAKENHI Grant Number 15K00423 and the Kayamori Foundation of Informational Science Advancement.

REFERENCES

[1] David L. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms" *Communications of the ACM,* Vol. 24, Issue 2, pp. 84-90, 1981.

[2] Paul Syverson, David Goldschlag, and Michael Reed, "Anonymous connections and onion routing," *IEEE Journal on Selected Areas in Communications,* Vol.16, issue 4, pp. 482-494, May 1998.

[3] Nguyen Phong Hoang and Davar Pishva, “A Tor-Based Anonymous Communication Approach to Secure Smart Home Appliances,” *ICACT Transaction on Advanced Communications Technology (TACT)* Vol.3, issue 5, pp. 517-525, Sept. 2014.

[4] David Goldschlag, Michael Reed, and Paul Syverson, "Hiding routing information," *Information Hiding*, Springer Berlin Heidelberg, 1996.

[5] David Goldschlag, Michael Reed, and Paul Syverson, "Onion routing," *Communications of the ACM* Vol. 42, issue 2, pp. 39-41, Feb. 1999.

[6] Paul Syverson, Gene Tsudik, Michael Reed, and Carl Landwehr, "Towards an analysis of onion routing security," *In Designing Privacy Enhancing Technologies*, pp. 96-114, Springer Berlin Heidelberg, 2001.

[7] John R. Douceur, "The sybil attack," *In Peer-to-peer Systems*, pp. 251-260, Springer Berlin Heidelberg, 2002.

[8] Nick Feamster and Roger Dingledine, "Location diversity in anonymity networks," *In Proceedings of the 2004 ACM workshop on Privacy in the electronic society*, pp. 66-76, ACM, 2004.

[9] Vitaly Shmatikov and Ming-Hsiu Wang, "Timing analysis in low-latency mix networks: Attacks and defenses," *In Computer Security–ESORICS 2006*, pp. 18-33, Springer Berlin Heidelberg, 2006.

[10] Jon McLachlan, Andrew Tran, Nicholas Hopper, and Yongdae Kim, "Scalable onion routing with torsk," *In Proceedings of the 16th ACM conference on Computer and communications security*, pp. 590-599, ACM, 2009.

[11] Matthew Edman, and Paul Syverson, "AS-awareness in Tor path selection," *In Proceedings of the 16th ACM conference on Computer and communications security*, pp. 380-389, ACM, 2009.

[12] Masoud Akhoondi, Curtis Yu, and Harsha V. Madhyastha, "LASTor: A low-latency AS-aware Tor client," *In 2012 IEEE Symposium on Security and Privacy (SP)*, pp. 476-490, IEEE, 2012.

[13] Nguyen Phong Hoang and Davar Pishva, “Anonymous communication and its importance in social networking,” *the 16th International Conference on Advanced Communication Technology,* pp. 34-39, IEEE, Feb. 2014.

[14] Roger Dingledine, Nicholas Hopper, George Kadianakis, and Nick Mathewson, "One fast guard for life (or 9 months)," *In 7th Workshop on Hot Topics in Privacy Enhancing Technologies (HotPETs 2014)*. 2014

[15] Yixin Sun, Anne Edmundson, Laurent Vanbever, Oscar Li, Jennifer Rexford, Mung Chiang, and Prateek Mittal, "RAPTOR: Routing Attacks on Privacy in Tor," arXiv preprint arXiv:1503.03940, 2015.

[16] Tor Project. [Online]. Available: https://www.torproject.org

[17] Tor Metrics. [Online]. Available: https://www.metrics.torproject.org

[18] Tor Browser. [Online]. Available: https://www.torproject.org/projects/torbrowser.html

[19] The Amnesic Incognito Live System. [Online]. Available: https://tails.boum.org

[20] Tor: Bridges. [Online]. Available: https://www.torproject.org/docs/bridges

[21] CollecTor. [Online]. Available: https://collector.torproject.org

[22] The Tor Challenge. [Online]. Available: https://www.eff.org/torchallenge

[23] Noisebridge Tor. [Online]. Available: http://tor.noisebridge.net

[24] Stem Python Library. [Online]. Available: https://stem.torproject.org

[25] Tor Compass. [Online]. Available: https://compass.torproject.org

[26] Hurricane Electric Internet Services BGP Toolkit. [Online]. Available: http://bgp.he.net

[27] Tor directory protocol, version 3. [Online]. Available: https://gitweb.torproject.org/torspec.git/tree/dir-spec.txt

[28] The Shadow Simulator. [Online]. Available: https://shadow.github.io/

[29] James Ball, *NSA stores metadata of millions of web users for up to a year, secret files show,* the Guardian, Sept. 2013.

[30] Ewen MacAskill, Julian Borger, Nick Hopkins, Nick Davies and James Ball, *GCHQ taps fibre-optic cables for secret access to world's communications,* the Guardian, June 2013.

[31] Lunar, *Tor Weekly News — September 4th, 2013.*

[32] Dave Neal, *Edward Snowden says the NSA has an autonomous Monstermind,* the Inquirer, Aug. 2013.

[33] Shane Harris and John Hudson, *Not Even the NSA Can Crack the State Dept.'s Favorite Anonymous Network,* Foreign Policy, Oct. 2014.

[34] Mary Madden, *Public Perceptions of Privacy and Security in the Post-Snowden Era,* Pew Research Center, Nov. 2014.

[35] Harmony, *Tor Weekly News — December 31st, 2014.*

[36] Jeremy Scahill and John Begley, *the Great SIM Heist*, the Intercept, Feb. 2015.

[37] MaxMind GeoLite2 Databases. [Online]. Available: http://dev.maxmind.com/geoip/geoip2/geolite2

[38] The CAIDA UCSD [as-relationships] - [retrieved on March 27th 2015]. Available: http://www.caida.org/data/as-relationships

[39] Internet AS-level Topology Archive. [Online]. Available: http://irl.cs.ucla.edu/topology/

[40] iPlane: An Information Plane for Distributed Services. [Online]. Available: http://iplane.cs.washington.edu

Nguyen Phong HOANG was born in Tien Giang Province, Vietnam in 1992. He received his undergraduate degree in Business Administration majoring in Information & Communications technology (ICT) from Ritsumeikan Asia Pacific University, Japan. He is presently pursuing his graduate studies at the Graduate School of Informatics at Kyoto University in Japan. His research interests include information security, privacy and anonymous communication. He hopes to advance his research on TOR (The Onion Router), one of the most robust anonymous tools, during his graduate studies. He participated in the 16th International Conference on Advanced Communication Technology and received Outstanding Paper Award from the Conference. He has been an IEEE member since 2013.

Yasuhito ASANO received the BS, MS, and DS degrees in information science, the University of Tokyo in 1998, 2000, and 2003, respectively. In 2003-2005, he was a research associate in the Graduate School of Information Sciences, Tohoku University. In 2006-2007, he was an assistant professor in the Department of Information Sciences, Tokyo Denki University. He joined Kyoto University in 2008, and he is currently an associate professor in the Graduate School of Informatics. His research interests include web mining, network algorithms. He is a member of the IEICE, IPSJ, DBSJ, and OR Soc. Japan.

Prof. Masatoshi YOSHIKAWA received the BE, ME, and PhD degrees from the Department of Information Science, Kyoto University, in 1980, 1982, and 1985, respectively. From 2002 to 2006, he served as a professor at Nagoya University. He has been a professor at Kyoto University since April 2006. His current research interests include database technologies and their application to medical and healthcare domains. He is a member of the ACM and IPSJ.