Towards Measuring Resilience in Anonymous Communication Networks

Fatemeh Shirazi  
KU Leuven, ESAT/COSIC and iMinds, Leuven, Belgium  
fsrirazi@esat.kuleuven.be

Claudia Diaz  
KU Leuven ESAT/COSIC, iMinds, Leuven, Belgium  
claudia.diaz@esat.kuleuven.be

Joss Wright  
Oxford Internet Institute, University of Oxford  
joss.wright@oii.ox.ac.uk

ABSTRACT

Prior research on anonymous communication networks has focused, to a large extent, on achieving, measuring, and evaluating anonymity properties. In this work we address another important security property that has so far received much less attention, namely resilience against denial-of-service attacks that degrade the performance of the network. We formally define resilience and propose a metric for quantifying the resilience of anonymous communication networks (ACNs) against active adversaries. Our metric expresses the degradation in quality of service of an ACN as the decrease in performance resulting from the adversarial removal or disabling of network nodes. We illustrate the practicality of the metric by applying it to a simulated version of the Tor network, and providing an evaluation of the resilience of Tor towards various adversarial strategies.

Categories and Subject Descriptors
C.2.0 [Computer-Communication Networks]: Security and Protection

Keywords
Resilience; anonymous communication networks; Tor; performance

1. INTRODUCTION

Anonymous communication networks (ACNs) enable users to communicate and access information anonymously over the Internet. Low-latency ACNs, such as Tor [6], establish bi-directional channels that can be used by interactive applications with low-latency constraints, such as web browsing. Using an ACN, users can conceal the destination of their communications from local adversaries such as their ISP, as well as protect their identity from the destination itself, e.g., a website. Modern ACNs are typically overlay networks composed of a set of routers, also called relays, or nodes, over which communications are relayed to achieve anonymity. In addition to providing robust anonymity properties, acceptable performance levels and strong resilience against both intentional and unintentional adverse network conditions are essential for an ACN that aims to sustain a broad user base. Evaluating these characteristics in ACNs requires formal definitions for the desired properties and a means to measure the extent to which the properties are satisfied by a given system design.

To the best of our knowledge, resilience, in terms of maintaining performance against active attacks has so far not been studied in the context of low-latency ACNs. In this work, we consider an active adversary whose aim, rather than de-anonymizing users, is to degrade an ACN service with the ultimate intention of rendering it practically unusable, and thus dissuading users from taking advantage of its protections. We note that this is a realistic threat: recently leaked NSA documents explicitly mention degrading the quality of service of Tor to discourage its use as a possible attack strategy [2]. Furthermore, such denial-of-service attacks can be deployed by adversaries with relatively modest resources. It may even be feasible for adversaries to deploy such attacks on short notice as a response to key events, such as moments of social unrest. An ACN failure at critical moments could seriously endanger users with a pressing need to communicate, who must then do so over unprotected channels. More generally, a sustained degradation of communication quality discourages users leading to a smaller user base, and thus to weaker anonymity [5].

The resilience of a communication network can be thought of as “the ability to provide and maintain an acceptable level of service” [19]. This paper introduces a metric for measuring the resilience of ACNs that takes into account the different features and constraints of ACN routing policies. We aim at measuring resilience against active adversaries that want to degrade as much as possible the expected quality of service. We illustrate our metric with examples of various attack scenarios against a simulated Tor network, evaluating Tor’s resilience against these attacks.

1.1 Related Work

Resilience is often measured in terms of network connectivity and failure rates [1, 18]. Doyle et al. [7] define the resilience of a network as the relation between the number of connected node pairs and the total number of nodes. Existing resilience metrics are, however, often inappropriate for ACNs as they differ from standard communication networks in several important ways. Firstly, the underlying topology of ACNs is often a near-complete graph, and as such connectivity is largely guaranteed. Secondly, in ACNs we are
chiefly concerned with active adversaries who strategically attack nodes to degrade the ACN service, and thus the modeling of random failures is of less direct relevance. Finally, whereas routing policies in standard networks are usually aimed at minimizing cost factors and finding the shortest available routes, ACN routing policies are randomized to provide stronger anonymity properties.

Borisov et al. [4] consider an active attack in which the adversary controls a subset of nodes, and tries to deanonymize communications by deploying selective DoS attacks against honest nodes. In the attack, the adversary performs a DoS attack only when he cannot compromise a communication. The key idea is to force the user to re-establish the connection and thereby improve the chances for deanonymization, which is achieved when the first and last node of the connection are adversarial. Their proposed metric focuses on evaluating the impact of these attacks on anonymity, rather than on the network performance. Techniques that have been proposed for disabling Tor nodes are for example [8] [11] [3]. The strategies we consider in our evaluation aim instead at optimally picking nodes to attack to maximize the impact of the attack.

2. RESILIENCE IN ANONYMOUS COMMUNICATION NETWORKS

Here we present our metric for measuring the resilience of ACNs. The metric does not consider connectivity, but rather measures the expected loss in the quality of communications (modeled as a random variable) when the adversary takes out a number of nodes.

2.1 Adversary Model

We consider an active adversary whose aim is to decrease the quality of service offered by some ACN. The adversary is not directly interested in compromising anonymity, but instead aims to discourage or prevent use of the network. We allow the adversary to select strategically the nodes to be attacked by a denial of service. The adversary might be restricted in different ways, such as only being able to remove nodes in a specific subnet, or country; being bound by the number of nodes he can disable; or being bound by the total amount of bandwidth that he can remove from the ACN. In our experiments we consider the latter case, in which the adversary can only remove a set of nodes whose combined bandwidth does not surpass the adversary’s budget.

2.2 Resilience Model

Let \( G = (V, E) \) be a graph. \( V \) and \( E \) represent, respectively, the set of ACN routers and the connections between them. Each vertex is labeled with a non-negative real number corresponding to the router bandwidth capacity. We define a route in \( G \) to be a finite sequence of distinct vertices and view a communication circuit from the sender to the receiver as a route through \( G \).

**Definition 1.** Let \( G \) be a graph. A routing policy on \( G \) is a set of rules specifying all admissible routes \( R \) through \( G \), together with a probability distribution on \( R \).

The routing policy thus determines which routes, selected from all possible ones, can be used for relaying communications, as well as the probability that a specific route is chosen.

**Definition 2.** An anonymous communication network \( A \) consists of a graph \( G \) together with a routing policy on \( G \) and a set of users \( U \).

For the purposes of measuring resilience we do not concern ourselves with details of how anonymity is provided, considering that this an implicit part of the routing policy.

**Definition 3.** Let \( A \) be an anonymous communication network with graph \( G \) and let \( X \) be a discrete non-negative random variable measuring the quality of communication in \( A \), where the quality metric may be bandwidth, latency, throughput, or some other observable variable of interest. We define the quality \( Q_X(A) \) of \( A \) with respect to \( X \) as the expected value of \( X \):

\[
Q_X(A) := \sum_x x \cdot Pr[X = x].
\]

In our attack model we consider an adversary who wishes to significantly lower \( Q_X(A) \), which is achieved by removing a subset of nodes \( S \) from the network, which have a total bandwidth of \( c \). We denote such an adversary as \( Ad_{X,S} \).

**Definition 4.** Let \( A \) be an ACN on \( G \) and let \( A_S \) be the the ACN obtained after removing a set \( S \) of nodes from \( G \). The routing policy of \( A_S \) is induced from \( A \). We define the resilience of \( A \) against \( Ad_{X,S} \) to be

\[
R_{Ad_{X,S}}(A) := \begin{cases} Q_X(A_S) & \text{if } Q_X(A_S) < Q_X(A) \\ 1 & \text{otherwise} \end{cases}
\]

According to this definition, the resilience has an upper bound of 1. Considering an adversary who can remove set of nodes \( S \) having a total bandwidth of \( c \), we define the resilience of \( A \) with respect to \( X \) to be

\[
R_{X,c}(A) := \min_S \{ Q_X(A_S) \mid \sum_{s \in S} bw(s) = c \} \cdot \frac{Q_X(A)}{Q_X(A_S)}.
\]

Thus, while \( R_{Ad_{X,S}} \) measures resilience with respect to the removal of an arbitrary set \( S \) of nodes, \( R_{X,c} \) does so with respect to an optimal attack. Comparing \( R_{Ad_{X,S}} \) and \( R_{X,c} \) gives us a measure of the effectiveness of an adversary’s strategy. Our metric can be extended to measure the minimum number of nodes that, if removed, degrade communication quality by a factor \( \delta \). This factor is of interest both to ACN designers, to understand the resilience of their network against targeted attacks, and to adversaries seeking to attack the ACN most effectively.

**Definition 5.** The \( \delta \)-threshold of \( A \) with respect to \( Ad_{X,S} \) is given by

\[
T_{\delta,Ad_{X,S}}(A) := \min(c > 0 : Q_X(A_S) = \delta \cdot Q_X(A)).
\]

3. MEASURING RESILIENCE OF TOR

3.1 Background on Tor

In the next section we apply our resilience metric to evaluate the Tor network via simulation. We give here a brief background on Tor [6]; more information can be found at www.torproject.org.

To communicate anonymously through Tor, users first download the consensus from a directory server. Consensuses are generated hourly, and comprise a list of available
Tor nodes, also known as relays or onion routers (OR), together with their contact details, such as IP address, public keys, and additional information like node bandwidth and various node flags. The client software then selects nodes from the most recent consensus (according to Tor’s routing policy) to construct anonymous circuits. The first node is called the entry or guard node, the second node is the middle node, and the last node is the exit node. While all routers can potentially act as middle nodes, only some nodes are flagged as guard or exit nodes. Each node’s operator decides whether that node should advertise as an exit node; guard nodes, on the other hand, are flagged as guards based on superior bandwidth and mean uptime. Nodes that are both guard and exit nodes are called guard-exit nodes. Finally, due to performance considerations, Tor’s routing policy does not select each possible route with the same probability; instead, preference is given to high-bandwidth nodes.

4. TOR’S RESILIENCE IN TERMS OF LATENCY

In this section we examine Tor’s resilience in terms of latency against DoS attacks. Latency is one of the main challenges of ACNs. User studies have reported reported user dissatisfaction with Tor due to its comparatively higher latency than normal browsing: [13] [15]. Given the latency increase inherent in Tor’s use of three intermediary relays for each connection, even a minor increase in overall latency is likely to have significant negative effects on user uptake and retention of Tor as a whole.

The traffic flowing through Tor is highly variable and depends on the number of users and applications they use, which include mainly web browsing and file transfer. As the resources in the network are limited, heavier traffic can lead to congestion. In turn, congestion at the node level increases the latency for communications relayed through these nodes [17].

To model Tor’s routing, we define $R$ to be our set of admissible routes in $G$, and $r$ to be the probability distribution on $R$ that is given by the routing policy. We define the latency experienced by a user $u$ choosing a route $r$ in an ACN as follows.

**Definition 6.** Let $A$ be an ACN on graph $G$ and let $r$ be a route in $G$. We define a function $L_u(r)$ that returns the latency of route $r$ when chosen by user $u$. The nature of the function $L$ is complex. Latency in Tor is affected by various factors including node congestion, propagation delay, and internet traffic-based delays, among others. In this work we make use of empirical measurements to obtain values for our function $L$, and consider the time until the first byte of a communication arrives, and the time it takes for the last byte to arrive. We denote these two metrics as ‘ttfb’ (‘time to first byte’) and ‘ttlb’ (‘time to last byte’), respectively, and consider both separately. We note that there are other latency measures that can be taken into account as measures of interest, such as the round trip time.

Let $R$ be the set of all admissible routes in $G$, and let $L$ be the set of latencies of all admissible routes. We compute the probability of a communication having a latency $l$ as:

$$Pr[L_u = l] = \sum_{r \in R: L_u(r) = l} Pr[R = r].$$

Higher latency decreases the quality of service, and as such we define the quality of communication as:

$$Q_{L_u}(A) = 1/\text{Exp}(L_u) = 1/\sum_{l} l \cdot Pr[L_u = l].$$

Exp can be replaced by other appropriate statistical measures such as first quartile, median, and third quartile. Note that we are considering $Q_{L_u}(A)$ to represent the quality for a single communication.

4.1 Resilience against latency reduction

In this section we demonstrate the practical use of our metric by evaluating attacks that aim at increasing Tor’s latency. For this we perform experiments using the Tor simulator Shadow. We assume that the attacker is capable of taking out 10% of Tor’s bandwidth. Hence, for each consensus the attacker takes out nodes, following her attack strategy, until 10% of the bandwidth has been removed. We consider two attacks and investigate their impact on Tor’s latency. In the first attack the adversary chooses which nodes to remove according to their bandwidth. In the second the nodes are chosen taking into account their geographical location.

**Attacking nodes according to their bandwidth.**

In this set of experiments we want to investigate the effect of removing nodes of a certain bandwidth on Tor’s resilience. We investigate the question: “given an adversary that is bounded in bandwidth, what has more impact on Tor’s expected latency, taking out a few big nodes or a larger number of smaller nodes?” For this we compare two node removal strategies. In the first, naive attack, we remove the nodes with the highest bandwidth until our adversary limit is reached. In the second, which we call distributed attack, we remove nodes with lower bandwidth until the limit is reached. Specifically, we do not attack nodes that are in the top 7% of largest bandwidth nodes, and attack instead nodes of high bandwidth outside that subset. Since the ttfb of the web user is equal to the ttfb of the bulk user, it is not included in the graphs.

**Attacking nodes in different geographic regions.**

In this experiment we want to investigate the effect of removing nodes in different geographical locations. We have three strategies for this experiment. We remove nodes from North America, Europe, or Asia and compare the effect of performing DoS attacks on nodes located in these regions. Within each region we chose the biggest nodes possible until the adversary’s capacity was reached.

4.2 Experimental Setup

For our experiments we used the Tor simulator Shadow [10], which simulates a downscaled version of Tor. We used two different user models in Shadow, a web user model, where the Tor user sends only small web requests, and a bulk user, which uses Tor for large file transfers. We used the 1.9.2 Shadow version. The Shadow simulator has the advantage that it simulates the network environment including the Tor users. However Shadow also has limitations, it needs substantial computational power, and running a Shadow simulation with the same size as the Tor network is infeasible in a standard computer. Another limitation of Shadow is that its routing policy corresponds to an older version than the one currently implemented in Tor, meaning
that there are additional weights that are not incorporated in the current version of Shadow [14]. For each execution of the simulation we chose a set of guard lists. This led to differences of up to 2 seconds in the measurements. The choice of guards has a very large influence on propagation delay. The geographic distribution of the relays also has a strong influence on the propagation delay. Note that in Shadow, the downscaled network is not representative of the geographic distribution of Tor, but rather the general Internet [9]. The geographic distribution of the users modeled in Shadow is based on [12]. Note that data from different consensuses are not independent, as the same routers are likely to appear in many consensuses. As such, we picked consensuses that were spaced some time apart, selecting one consensus at random from the first week of each month in the first half of 2013. This approach also aims to reduce daily, weekly, or other seasonal effects on the consensus. Shadow uses one script as input for the simulation, the so-called host file, which contains a list of nodes and users that are used for the simulation. We chose the number of servers, users and relays to closely match the Tor topology. For user activity we follow the model presented in [9], where web clients repeatedly download a 320 KiB file while pausing between 1 to 60 seconds after every download. Bulk clients continuously download 5 MiB files without pausing. The perf clients download a file every 60 seconds, with 75 clients downloading a 50 KiB file, 75 clients downloading a 1 MiB file, and 75 clients downloading a 5 MiB file. The network size for our Shadow experiments was 394 relays, from which 201 were middle relays, 73 exit relays, 91 guard relays, and 29 exit-guard relays. The user set consisted of 1080 web clients, 120 bulk clients, 225 perf clients and 350 file servers. Attacks are simulated by removing the targeted relays (based on each of the attack strategies) from the host file. In Table 1 we list the number of nodes removed for each attack.

### 4.3 Latency Measurements and Resilience Results

In this section we review the results of our latency experiments. Since the propagation delay causes a significant variance in our measurements we chose to take time to last byte (ttlb) for bulk users as our measure of interest. Compared to other measures (such as time to first byte for either interactive or bulk users) the ttlb is rather influenced by the delay caused by congestion. In order to decide on the effectiveness of the attacks, we chose as a measure of comparison the expected latency of each attack. In Table 2 we show the mean latency measurements for 6 consensuses picked from the first half of 2013 under the naïve and distributed attack strategies.

<table>
<thead>
<tr>
<th>Consensus</th>
<th>Before naïve</th>
<th>tttl</th>
<th>NA</th>
<th>EU</th>
<th>Asia</th>
</tr>
</thead>
<tbody>
<tr>
<td>06.01.13 at 17h</td>
<td>54.723</td>
<td>66.189</td>
<td>65.034</td>
<td>65.691</td>
<td>62.677</td>
</tr>
<tr>
<td>07.02.13 at 09h</td>
<td>53.321</td>
<td>51.855</td>
<td>56.964</td>
<td>55.041</td>
<td>51.855</td>
</tr>
<tr>
<td>04.03.13 at 22h</td>
<td>42.063</td>
<td>40.595</td>
<td>43.434</td>
<td>43.294</td>
<td>43.865</td>
</tr>
<tr>
<td>02.04.13 at 23h</td>
<td>40.709</td>
<td>40.067</td>
<td>43.392</td>
<td>43.176</td>
<td>38.754</td>
</tr>
<tr>
<td>07.05.13 at 04h</td>
<td>39.757</td>
<td>43.466</td>
<td>46.113</td>
<td>44.021</td>
<td>41.468</td>
</tr>
<tr>
<td>04.06.13 at 20h</td>
<td>39.988</td>
<td>39.356</td>
<td>40.224</td>
<td>39.612</td>
<td>38.651</td>
</tr>
</tbody>
</table>

Table 1: Number of nodes attacked in the naïve and distributed attack strategies and regional specific attack strategies.

<table>
<thead>
<tr>
<th>Consensus</th>
<th>naïve distr. NA EU Asia</th>
</tr>
</thead>
<tbody>
<tr>
<td>06.01.13 at 17h</td>
<td>4 9 6 4 40</td>
</tr>
<tr>
<td>07.02.13 at 09h</td>
<td>3 7 4 3 46</td>
</tr>
<tr>
<td>04.03.13 at 22h</td>
<td>4 10 4 4 31</td>
</tr>
<tr>
<td>02.04.13 at 23h</td>
<td>3 9 5 4 24</td>
</tr>
<tr>
<td>07.05.13 at 04h</td>
<td>4 6 7 4 31</td>
</tr>
<tr>
<td>04.06.13 at 20h</td>
<td>3 5 3 5 26</td>
</tr>
</tbody>
</table>

In Tor, the available capacity is often not fully used because of the /16 IP subnet conflict rule (meaning that circuits must not contain two nodes within the same /16 subnet). This is in particular true for big nodes, which are often clustered in a small number of /16 IP subnets. When attacking medium sized nodes, the adversary decreases bandwidth in more of these /16 IP subnets than when taking out big nodes. Our measurements also show a trend when attacking nodes from US. This can be due to two reasons. First, the nodes located in in our sample data were smaller in bandwidth than in Europe and hence the number of nodes removed was higher. The second reason might be related to the Alexa websites used for the downloads which are mostly located in North America (270 out of 350) which increases the propagation delay from the exit node to the destination website when the exit is outside North America. Moreover, we see that with our DoS attacks, which disable 10 % of the networks bandwidth, the latency has only a modest increase.
Table 3: Resilience in terms of ttlb (in seconds) for bulk users for 6 consensuses picked from the first half of 2013 after applying the distributed attack strategy when quality of communication is defined as different types of statistical measures.

<table>
<thead>
<tr>
<th>Consensus from</th>
<th>$Q_{L_{\text{bulk}}}$:</th>
<th>$Exp(L)$</th>
<th>$Q_1$</th>
<th>$Q_2$</th>
<th>$Q_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>06.01.13 at 17h</td>
<td>Resilience: 0.841</td>
<td>0.939</td>
<td>0.857</td>
<td>0.798</td>
<td></td>
</tr>
<tr>
<td>07.02.13 at 09h</td>
<td>Resilience: 0.936</td>
<td>0.943</td>
<td>0.914</td>
<td>0.891</td>
<td></td>
</tr>
<tr>
<td>04.03.13 at 22h</td>
<td>Resilience: 0.968</td>
<td>0.965</td>
<td>0.969</td>
<td>0.958</td>
<td></td>
</tr>
<tr>
<td>02.04.13 at 23h</td>
<td>Resilience: 0.938</td>
<td>0.984</td>
<td>0.970</td>
<td>0.936</td>
<td></td>
</tr>
<tr>
<td>07.05.13 at 04h</td>
<td>Resilience: 0.862</td>
<td>0.940</td>
<td>0.922</td>
<td>0.848</td>
<td></td>
</tr>
<tr>
<td>04.06.13 at 20h</td>
<td>Resilience: 0.994</td>
<td>0.971</td>
<td>0.968</td>
<td>0.981</td>
<td></td>
</tr>
</tbody>
</table>

for most measurements (with a few exceptions) indicating that the network is relatively resilient against our attacks. We note as limitation that since Shadow experiments are resource intensive, we used a significantly smaller network size compared to the whole Tor network which is currently approximately 7000 nodes [16].

5. CONCLUSION

We have proposed a metric for measuring the resilience of anonymous communication networks and demonstrated its practicality by applying it to a simulation of Tor. Our main observation was that attacking a larger number of medium-size nodes is more effective than attacking a smaller number of big-size nodes, for the same total bandwidth. We note however that our experiments suffer from a number of limitations, notably the downscaled version of Tor, and that larger-scale experiments are needed to confirm that these results hold in larger networks. Some open questions following this work include measuring the adversarial resources needed to degrade the performance of Tor to unusable levels, and finding optimal adversarial strategies for node removal.

Acknowledgment

The authors would like to thank Ciaran Mullan and Johannes Buchmann for the helpful discussions, and Bernd Conrad for his help with the experiments. This work was supported in part by the Research Council KU Leuven: GOA TENSE (GOA/11/007) and the Flemish Government FWO G.0360.11N Location Privacy, FWO G.068611N Data mining and by the European Commission through H2020-DS-2014-653497 PANORAMIX and H2020-ICT-2014-644371 WITDOM. This work was supported by Microsoft Research through its PhD Scholarship Programme.

6. REFERENCES


