Minimising Anonymity Loss in Anonymity Networks under DoS Attacks

Mu Yang and Vladimiro Sassone

ECS, University of Southampton

Abstract. Anonymity is a security property of paramount importance as it helps to protect users' privacy by ensuring that their identity remains unknown. Anonymity protocols generally suffer from denial of service (DoS) attack, as repeated message retransmission affords more opportunities for attackers to analyse traffic and lower the protocols' privacy. In this paper, we analyse how users can minimise their anonymity loss under DoS attacks by choosing to remove or keep 'failed' nodes from router lists. We also investigate the strategy effectiveness in those cases where users cannot decide whether the 'failed' node are the targets of DoS attacks.

1 Introduction

Protecting online privacy is an essential part of today's society and its importance is increasingly recognised as crucial in many fields of computer-aided human activity, such as eVoting, eAuctions, bill payments, online betting and electronic communication. One of the most common mechanisms for privacy is *anonymity*, which generally refers to the condition of being unidentifiable within a given set of subjects, known as the *anonymity set*.

Several mechanisms have been proposed to enforce privacy through the use of anonymity networks (e.g. [5,15,19,11,18]). Yet, the open nature of such networks and the unaccountability which results from the very idea of anonymity, make the existing systems prone to various attacks (e.g. [14,16,17,9]). The evaluation of such attacks on anonymity systems has largely focused exclusively on security – that is, how likely anonymity is to be compromised – with all other possible metrics considered tangential. Recent work, however, has started to address parameters such as performance and reliability [8,13,20], which are factors are self-evident importance: an unreliable system, or anyway a scarcely unusable system, will cause users to take their communications to other channels, possibly non-anonymous, and defeat the system's purpose altogether. An adversary may selectively affect the reliability of the system in those states which are hardest to compromise, in their attempt to make the system prefer less secure states. Faced with poor reliability, many users (and a lot of software too) will naturally attempt to repeat communication (resend messages), offering more opportunities for traffic analysis and attacks. Consequently, a considerable amount of research has recently been focussing on DoS attacks to anonymity networks [27,2,3,21].

In a DoS attack, the attacker is able to deanonymise users by affecting system reliability. More specifically, the attacker will choose users who are hardest to compromise,

S. Qing et al. (Eds.): ICICS 2011, LNCS 7043, pp. 414–429, 2011.

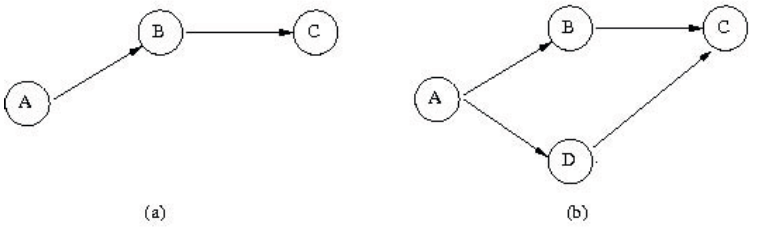


Fig. 1. Partial anonymity networks

and will attempt to make them appear as 'failed' to other users. That is, the attacker will actively make the targeted nodes very busy, so as to make them become unresponsive to their peers who seek cooperation to forward messages according to the selected anonymity protocol. Effectively, the target nodes remain cut off very soon indeed (cf. e.g. [24]). Throughout the paper, we refer to such slow/unresponsive/compromised nodes simply as *failed* (or apparently failed) nodes. While nodes under attack become unresponsive, malicious users on the network make sure to keep relative idle, thereby inducing honest users (and software alike) to attempt to communicate (viz., reroute their communication path) through them. In this way, malicious users obtain a double advantage: they decrease the ratio of cooperating users in the system, and make themselves look as very efficient communication peers, to be preferred over the others. Consider for instance the partial anonymity networks of Figure 1. In (a) user A sends a message to C via B. Assume now that B is targeted by a DoS attacker, then B appears failed to A. So A has to rebuild a path for forwarding messages to C. User A now sends the message to C via D in Figure 1(b). Therefore, if D is a malicious user, then the probability of identifying A increases because of D's own guess or both B and D's collaboration or other eavesdroppers.

Papers [2,3,21] showed that the DoS attacks in anonymous systems reduce anonymity considerably, so that in recent years, a considerable amount of research has been focusing on refining such systems. In particular, trust-and-reputation-based metrics have become quite popular in this domain [1,6,7,10,25,26]. Yet, introducing trust into the picture opens the flank to new security attacks, exploiting trust mechanisms explicitly, as proved in [24]. In such work, the authors evaluated the anonymity loss in trust-based Crowds protocol when DoS attacks take place. At the moment, there are no perfect solutions to address the DoS attacks. So, it becomes of great practical relevance to study what a user can do to minimise her anonymity loss when exposed to such attack scenarios.

In this paper, we investigate the two strategies a user can adopt when confronted with a failed node, viz., whether to remove or keep the 'failed' node from their router list, and try to determine which one is the best approach under different scenarios.

The Crowds protocol allows Internet users to perform anonymous web transactions by sending their messages through a random chain of users participating in the protocol. Each user in the '*crowd*' must establish a path between her and a set of servers by selecting randomly some users to act as routers (or forwarders). Such routing paths are formed so as to guarantee that users do not know whether their predecessors are

message originators or just forwarders. Each user only has access to messages routed through her. It is well known that Crowds cannot ensure strong anonymity in presence of corrupt participants [23,4], yet when the number of corrupt users is sufficiently small, it provides a weaker notion of anonymity known as *probable innocence*: informally, a sender is probably innocent if to an attacker she is no more likely to be the message originator than not to be.

In this paper we use the metric of probable innocence to measure anonymity loss. In other words, we consider DoS attacks as well as the classical insider attack to anonymity networks, which is when the malicious users in the systems collaborate to report (to some unspecified authority) the most likely initiator of each message they intercept. The list of participants (or users, or members) in Crowds is kept by a server called *blender*, which provides it to all registered users. When new node starts running the system to join the crowd, and is willing to act as a message-forwarder, the blender adds it to its list of routers. Each user maintains their own local list of routers, which is updated when the user receives notices of new or deleted members from the blender. Yet, the user can also drop routers from her list by her own decision, if she detects failed nodes.

Structure of the paper. The paper is organised as follows: we recall the fundamental ideas behind the Crowds protocol and its properties, including the most popular anonymity metrics, in §2 below. Then in §3 we present our first contribution: a model of the interactions between users and DoS attackers and, by using the game theoretic notion of *Nash Equilibrium*, an analysis of the strategies chosen by the users. Then, §4 repeats the analysis for a refined model in which we take into account the users' uncertainty about the nature the 'failed' node, viz., whether it is malicious or a honest user under a DoS attack. In such analysis, we introduce and discuss a key parameter, the probability that the 'failed' node is malicious, and presents some preliminary results of its analysis.

2 Related Work

Using conditional probabilities to measure anonymity level was proposed by Reiter and Rubin [23]. They proposed a hierarchy of anonymity notions in the context of Crowds. These range from '*absolute privacy*,' where the attacker cannot perceive the presence of an actual communication, to '*provably exposed*,' where the attacker can prove a sender-and-receiver relationship. The most important level is '*probable innocence*' which was originally defined as "A sender is probably innocent if, from the attacker's point of view, she appears no more likely to be the originator than to not be the originator."

Let n be the number of users participating in the protocol and let c and $n - c$ be the number of the corrupt and honest members, respectively. Since anonymity makes only sense for honest users, we define the set of anonymous events as $\mathcal{A} = \{a_1, a_2, \ldots, a_{n-c}\}$, where a_i indicates that user i is the initiator of the message.

As it is usually the case in the analysis of Crowds, we assume that attackers will always deliver a request to forward immediately to the end server, since forwarding it any further cannot help them learn anything more about the identity of the originator. Thus in any given path, there is at most one detected user: the first honest member to

forward the message to a corrupt member. Therefore we define the set of observable events as $O = \{o_1, o_2, \ldots, o_{n-c}\}$, where o_j indicates that user j forwarded a message to a corrupted user. In this case, we also say that user j *is detected* by the attacker. The corresponding notion of probable innocence is formalised by Reiter and Rubin [23] via the conditional probability that user i is detected given that she is the initiator, in symbols $P(o_i \mid a_i)$. Probable innocence holds if

$$\forall i.\ P(o_i \mid a_i) \leq \frac{1}{2} \tag{1}$$

In [23] it is also proved that the following holds in Crowds:

$$P(o_j \mid a_i) \;=\; \begin{cases} 1 - \dfrac{n-c-1}{n} p_f & i = j \\ \dfrac{1}{n} p_f & i \neq j \end{cases} \tag{2}$$

Therefore, probable innocence (1) holds if and only if

$$n \geq \frac{p_f}{p_f - 1/2}\,(c+1) \quad \textit{and} \quad p_f \geq \frac{1}{2}\,.$$

The formulae above show that the number n of participating users influences substantially the anonymity level that the system can provide to its users. If honest users are lost to the network, either because compromised or voluntarily withdrawn or removed from a router list following a DoS attack, then the remaining honest users will indirecIty suffer a loss of anonymity. This happens of course as a side-effect of being part of a network which can only provide a lower anonymity guarantee, due to a less favourable ratio of honest and malicious users.

Numerous denial of service (DoS) attacks have been reported in the literature. In particular, the *'packet spinning'* attack of [21] tries to lure users into selecting malicious relays by targeting honest users by DoS attacks. The attacker creates long circular paths involving honest users and sends large amount of data through the paths, forcing the users to employ all their bandwidth and then timing out. These attacks motivate the demand for mechanisms to enhance the reliability of anonymity networks. In particular, paper [3] investigates the effects of DoS attacks on some anonymity systems, such as Tor, Hydra-Onion, Cashmere, and Salsa, and shows greater opportunities to compromise anonymity under DoS attack, and the systems cannot tolerate a majority of nodes being malicious.

To address the DoS attack, several mechanisms were proposed to enhance anonymity against DoS attacks. Trust-and-reputation-based metrics are quite popular in this domain [1,6,7,10,25,26]. Enhancing the reliability by trust and reputation, not only does improve the system's usability, but may also increase its anonymity guarantee. Indeed, a trust-based selection of relays improves both the reliability and the anonymity of the network, by delivering messages through 'trusted' routers. Moreover, the more reliable the system, the more it may attract users and hence improve the anonymity guarantee by growing the anonymity set. Introducing trust in anonymity networks does however open the flank to novel security attacks, as proved in [24].

Importantly, a users' own response to a DoS attack may effectively decrease their anonymity loss. In particular, in the design of Crowds, users can choose removing or keeping routers if they detect those routers (or nodes) failed, and each response will lead to different anonymity payoffs. In this paper, we investigate this and find out which response yields the best strategy for users under different scenarios.

3 Minimizing Users' Anonymity Loss

3.1 Preliminaries and Notations

As discussed in the previous section, the Crowds protocol and the related metrics to measure its anonymity have been enhanced against DoS attacks. Among these, the notion of trust has been considered. However, introducing trust in anonymity networks paves the way novel attack opportunities. In this section, we focus on how to minimize the anonymity loss without introducing new mechanisms. We first describe our approach based on *game theory*, then model users and DoS attackers as players in our game, and devise formulae for these players' payoffs. Finally, we investigate the strategy users should choose and its effect compared with other response actions under DoS attack.

We adopt two working assumptions: users want to forward messages to others, and they value their anonymity. Thus in our analysis of the paper, we take anonymity as part of evaluation of users' payoffs. The anonymity systems generally consist of two groups of components under the DoS attacks: users compete against DoS attackers for good anonymity services (e.g., good privacy level guaranteed by the systems) and, at the same time, the DoS attackers try their best to de-anonymize the systems. Here, users' strategic objective, their 'utility', is to minimize their anonymity loss, while the attackers endeavour to gain maximum benefits from the attacks.

As discussed in §2, the DoS attack is deployed to make the target (honest) user, say k unavailable or unresponsive. This causes the message forwarder, say i, to reroute her (anonymity) path to her messages' destinations, say D, and thus suffer anonymity loss implied by the rerouting activity and, at the same time, run an increased risk to pick the malicious users on her new paths which illustrated in Figure 2.

As such, the interaction between user i and the attackers A in the anonymity systems is best modeled as a *non-cooperative* game among the rational and strategic players. Players are rational because they wish to maximize their own gain, and they are

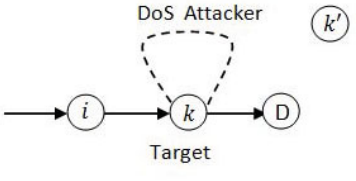


Fig. 2. The attacker targets node k making it unavailable to user i. User i chooses her strategy –whether removing k from her router list– by predicting the attacker's strategy.

strategic because they can choose their actions (e.g., remove failed node from router list) that influence both their payoffs. Another force tending to affect users' anonymity payoffs is the so-called 'churn problem' [22], which is caused by frequent 'arrivals' and 'departures' of users. In order to coordinate such membership events, the standard implementation of Crowds makes use of a dedicated process, the so-called 'blender.' Among its functions, the blender provides updates to the users' lists of crowd members at regular intervals, In this paper we use T to to represent such interval, and model a user's behaviour during the time interval between two list updating events.

With respect to the target node k, the attackers have the following two strategies:

- *Strategy* S'_1: keep targeting k;
- *Strategy* S'_2: target a user other than k, say k'.

The user i also has two strategies:

- *Strategy* S_1: remove k from router list;
- *Strategy* S_2: keep k on the router list.

With these strategies above, the payoffs of these two players are expressed in the form of 2×2 matrices, $U_i(\_)$ and $U_A(\_)$, which denote user i's and the attackers' payoff, respectively. A game is called a *zero-sum* game, if the sum of the utilities is constant in every outcome, so that whatever is gained by one player must be lost by the other players, that is, in our case, if the sum of $U_i(\_)$ and $U_A(\_)$ is constant, then our game is a zero-sum game.

Interesting economic behaviour occurs when the utility of a player depends not only on her own strategy, but on the other's strategy as well. The most popular way of characterising this dynamics is in terms of *Nash equilibrium*. Since a player's utility depends on her strategy, she might unilaterally switch her strategy to improve her utility. This switch in strategy will affect the other player, so that she might decide to switch her strategy as well. One game reaches a Nash equilibrium if no player can improve her utility by unilaterally switching strategy. In Section 3.2, we will model the scenario illustrated in Figure 2 and study the equilibrium of our game. We present here for the reader's convenience a table summarizing those variables that will appear in the modeling.

3.2 Users' Protection Model

We first discuss the notations which are used in the evaluation of i's utility function:

- $U_i(t)$: it reflects i's anonymity level guaranteed by the system. At time $t \in T$, user i suffers the DoS attack and then she chooses her strategy, thus her payoff at time $t+1 \in T$ is defined as $U_i(t+1)$. For time t, the payoff $U_i(t)$ can be evaluated by the anonymity measure metrics as

$$U_i(t) = 1 - P(o_j \mid a_i)_t$$

  where $P(o_j \mid a_i)_t$ represents the anonymity level measuring of user i at time t. Thus, the value of $U_i(t)$ is not greater than one. The smaller the value of $P(o_j \mid a_i)_t$, the higher anonymity level of user i guaranteed, that is, the greater i's payoff $U_i(t)$;

– L_i: when user i mistakenly removes an normal honest user from her router list, she will suffer some anonymity loss which is measured by L_i. The loss here can be evaluated by the probable innocence anonymity metrics, which we described in §2. In particular, in systems like Crowds protocol, the loss will be

$$\begin{aligned}&\left[1-\left(1-\frac{n-c-1}{n}p_f\right)\right]-\left[1-\left(1-\frac{n-1-c-1}{n-1}p_f\right)\right]\\ =&\frac{n-c-1}{n}p_f-\frac{n-1-c-1}{n-1}p_f\\ =&\frac{c+1}{n(n-1)}p_f.\end{aligned}$$

– L_{ik}: if the attacker successfully perpetrates a DoS attack by targeting k, then i will suffer anonymity loss related to re-routing. Here L_{ik} measures two kinds of anonymity loss: one is the information leak because of this another forwarding path, the other one is that due to the failed target node cannot response to i, the probability of choosing malicious users as forwarders on the path increases.

We define i's payoff matrix as follows:

$$U_i(t+1)=[u_{qp}]_{2\times 2}=\begin{bmatrix}U_i(t) & U_i(t)-L_{ik}\\ U_i(t)-L_{ik'}-L_i & U_i(t)-L_{ik'}\end{bmatrix}$$

where $[u_{pq}]_{2\times 2}$ denotes the player's payoff when she chooses strategy p and the other player chooses strategy q. Here u_{11} represents the payoff if the players follow the strategy pair (S_1, S'_1), which is when the attacker chooses to target k and the user chooses to remove the same user k. Thus, for i, the utility is her original utility value of $U_i(t)$. The term u_{21} represents the payoff corresponding to the strategy pair (S_1, S'_2), which is when the attacker targets a new node k', but i still removes k. In this case we subtract from the original utility the loss due to the attack to k', as well as the loss of removing the honest node k. The term u_{12} represents the payoff of strategy pair (S_2, S'_1), that is the attacker still targets k and i chooses to keep k, respectively. The term u_{22} represents the payoff of strategy tuple (S_2, S'_2), which is when the attacker targets k' other than k

Table 1. Variables used in the modeling

$U_i(t)$:	user i's payoff at time t
$U_A(\_)$:	the DoS attacker's payoff at time t
L_i:	i's anonymity loss following the removal of an honest user from router list
L_{ik}:	i's anonymity loss because of DoS target k
n:	the number of users in the system
c:	the number of malicious users in the system
θ:	the percentage of target nodes among honest users
p_f:	the forwarding policy of the system
C_k:	the cost of targeting node k for the DoS attackers
B_k:	the benefit of de-anonymization for the DoS attackers because of successfully targeting k
B':	the benefit of de-anonymizing for the DoS attackers when one honest user is removed from router list by i
$P(o_j \mid a_i)_t$:	the anonymity level measuring of user i at time t

and i keeps k. In this case we subtract from the original utility the loss of the attack for successfully targeting k'.

We define the attacker's payoff matrix as follows:

$$U_A(\_) = \begin{bmatrix} -C_k & B_k - C_k \\ B_{k'} - C_{k'} + B' & B_{k'} - C_{k'} \end{bmatrix}$$

Here for B_k, since user i's anonymity loss measured by L_{ik} and $L_{ik'}$ are actually the attacker's aim, the benefit B_k, $B_{k'}$ are equal to L_{ik} and $L_{ik'}$, respectively. Similarly to B_k above, the benefit B' is equal to L_i. Thus, we have,

$$U_A(\_) = \begin{bmatrix} -C_k & L_{ik} - C_k \\ L_{ik'} + L_i - C_{k'} & L_{ik'} - C_{k'} \end{bmatrix}.$$

Note that when the cost C_k of targeting node k is the same as $C_{k'}$, the game is a *zero-sum* game, that is, the total payoff of these two players' at each strategy pair is always

$$U_i(t+1) + U_A(\_) = U_i(t) - C_k.$$

User i's loss is exactly balanced by the gains of the DoS attacker.

In order to find an equilibrium of our model, we turn our attention to Nash's theorem [12], which proved that such games always have at least one equilibrium in mixed strategies, we have:

Proposition 1.

$$X = \left(\frac{(L_{ik} - L_{ik'}) - (C_k - C_{k'})}{L_i + L_{ik}}, \frac{(L_i + L_{ik'}) + (C_k - C_{k'})}{L_i + L_{ik}} \right), \qquad Y = \begin{pmatrix} \frac{L_i}{L_i + L_{ik}} \\ \frac{L_{ik}}{L_i + L_{ik}} \end{pmatrix}.$$

Proof. Typically, in a given game represented by a payoff matrix $A_{p\times q}$, vectors x_j and y_j below form a pair of mixed strategies if $\{x_j \in \mathbb{X}^p, x_j \geq 0, \sum_{j=1}^{p} x_j = 1\}$, and $\{y_j \in \mathbb{Y}^q, y_j \geq 0, \sum_{j=1}^{q} y_j = 1\}$ hold. For our game, let us suppose that user i can play:

$$X = \begin{bmatrix} x_1 & x_2 \end{bmatrix}$$

where $x_1 + x_2 = 1$. An attacker can also play:

$$Y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$

where $y_1 + y_2 = 1$. Then, according to [12], the best-response correspondence function is given by: $P(x, y) = \sum_{i=1}^{p} \sum_{j=1}^{q} A_{ij} x_i y_j$. Thus, replacing A with respectively $U_i(t+1)$ and $U_A(\_)$, we obtain the correspondence functions for user i and the attacker, respectively.

$$P_i(x, y) = U_i(t)x_1y_1 + (U_i(t) - L_{ik})x_2y_1 + (U_i(t) - L_{ik'} - L_i)x_1y_2 + (U_i(t) - L_{ik'})x_2y_2$$

$$P_A(x, y) = -C_k x_1y_1 + (L_{ik} - C_k)x_2y_1 + (L_{ik'} + L_i - C_{k'})x_1y_2 + (L_{ik'} - C_{k'})x_2y_2.$$

By solving the payoffs matrix above, we get the Nash Equilibrium of the game in mixed strategies. □

The intuition behind the above equilibrium is that X is of the form $[\gamma, 1-\gamma]$, that is in order to gain maximum payoff, user i should remove k with probability γ (or at γ times). If the anonymity system is symmetric (i.e., the users have the same equilibrium point) or each user updates the router list from the blender of the system, instead of managing it herself, then the two players of our game become: the DoS attackers and the group of all the users. In this case if β nodes are reported failed to the blender, then the optimal strategy for the blender in our equilibrium analysis is to remove $\beta \times \gamma$ nodes from router list.

The game can reach a pure Nash equilibrium when some conditions are satisfied.

Corollary 1. *Strategy 'Keeping k' is the best strategy, that is* $X = [0, 1]$ *if the following holds.*
1. $L_{ik} = L_{ik'}$*;*
2. $C_k = C_{k'}$.

Observe that in Crowds protocol, users are typically indistinguishable from each other from the attacker's point of view. The formulae of Corollary 1 are therefore often satisfied in Crowds-based systems. It follows from the above proposition that in those systems the best response strategy for user i is to keep k.

3.3 Evaluating the Anonymity Loss of the Strategies

We now focus on the impact of the choice of a strategy on the anonymity of i. Let i's utilities at $t+1$ when i chooses strategy S be noted as $U_i(t+1)_S$, and the mixed strategy respectively at $U_i(t+1)_{Mixed}$.

If user i always chooses to strategy S_1(, that is removing node k), the attacker will always answer by choosing the strategy to target a different node k', and the utility at time $t+1$ is evaluated as

$$U_i(t+1)_{S_1} = U_i(t) - L_{ik'} - L_i\,.$$

Similarly, the utilities of user i if she keeps k(, that is strategy S_2), or selects the mixed Nash equilibrium strategy can be computed respectively as follows.

$$U_i(t+1)_{S_2} = \begin{cases} U_i(t) - L_{ik'} & L_{ik'} \geq L_{ik} \\ U_i(t) - L_{ik} & L_{ik'} \leq L_{ik} \end{cases}, \qquad U_i(t+1)_{Mixed} = U_i(t) - \frac{L_{ik'} + L_i}{L_{ik} + L_i} L_{ik}$$

All these three utilities decrease as time increases. We are of course interested in minimizing the anonymity loss.

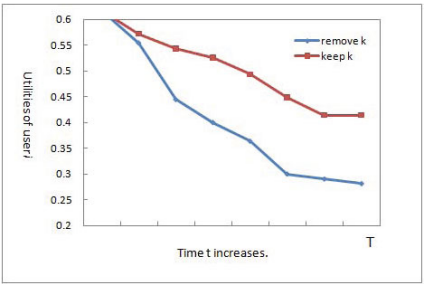
Proposition 2.

$$U_i(t+1)_{S_1} < U_i(t+1)_{Mixed}\,, \quad U_i(t+1)_{S_2} \leq U_i(t+1)_{Mixed}\,.$$

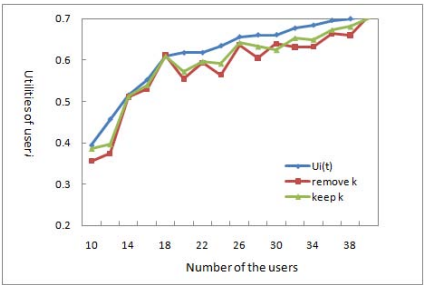
Proof. Because $U_i(t+1)_{S_1} - U_i(t+1)_{Mixed} = -\frac{L_{ik'}+L_i^2}{L_{ik}+L_i} < 0$ holds, thus we obtain $U_i(t+1)_{S_1} < U_i(t+1)_{Mixed}$. If $L_{ik} \geq L_{ik'}$, then $U_i(t+1)_{S_2} - U_i(t+1)_{Mixed} = L_{ik}\frac{L_{ik'}-L_{ik}}{L_i+L_{ik}} \leq 0$; If $L_{ik} \geq L_{ik'}$, then $U_i(t+1)_{S_2} - U_i(t+1)_{Mixed} = L_{ik}\frac{L_{ik}-L_{ik'}}{L_i+L_{ik}} \leq 0$. Thus, $U_i(t+1)_{S_2} \leq U_i(t+1)_{Mixed}$. □

The proposition shows that by selecting the equilibrium strategy, user i will gain the maximum utility $U_i(t+1)_{Mixed}$. And as time increases, the distance between the mixed strategy and other two will increase. Our simulations were written in Java, executed on Sun's JVM and based on Crowds protocol with indistinguishable users. The parameters are: $c = 4$, $p_f = 0.8$; the DoS attackers always target two honest users at one time; we vary the number n of users and increase time t until T. Because the users are indistinguishable, the mixed strategy is actually the pure strategy –keeping node k– according to Corollary 1.

Figure 3(a) shows that by adopting the mixed strategy (keeping k), user i will gain better utility as time increases. Figure 3(b) depicts the utilities of user i at time $t+1$ when adopting the mixed strategy versus removing node k, compared with $U_i(t)$. The diagram shows that although these two utilities are smaller than $U_i(t)$, selecting the 'keep k' strategy will always minimise i's anonymity loss. As the number of users increases (that is, the ratio of malicious users among the users decreases), the utilities increase and the mixed strategy has more strength on decreasing the anonymity loss.



(a) i's utilities decrease as time t increases. $n = 20, c = 4, p_f = 0.8$.



(b) i's utilities at time $(t+1)$ increase when the number n of users increases, compared with $U_i(t)$. $c = 4$, $p_f = 0.8$.

Fig. 3. i's utilities

4 Refined Protection Model

We have worked so far under the assumption that users know that the failed node k is the DoS target. Now we proceed to relax such an assumption and generalise our results, by taking the view that users cannot be sure about the type of k. Arguably, this is a rather realistic hypothesis in open and dynamic systems, where honest users can also be unavailable just because they suffer power cuts or network downtime or congestion and overloading. Another possibility is that the failed nodes are malicious users which are

carrying out attacks, such as reporting their predecessors as the most likely initiators, which will slow them down. Because of that, their predecessors may also classify them as unavailable/unresponsive nodes. In this section we therefore assume that k may be a normal honest user or a malicious user, and repeat our analysis of DoS attack protection model under such an assumption.

This new scenario differs from those we considered so far in the paper in that when a node k is detected as a failed node, rather than just considering it as a DoS target, user i has to decide whether it is a malicious user or simply a normal honest user who is just temporarily slowed down by e.g. network congestion. We define the uncertainty about the type of a failed node as $P_{ik}(\alpha \mid F)$, where α is the type to which the failed node belongs. More specifically, the term $P_{ik}(t \mid F)$ represents the probability that k encountered by user i is one DoS target and $P_{ik}(m \mid F)$, $P_{ik}(h \mid F)$ are the probabilities of k being respectively a malicious user and normal user type. For these three probabilities, we obviously have $P_{ik}(t \mid F) + P_{ik}(h \mid F) + P_{ik}(m \mid F) = 1$.

In the rest of this section we work out again the best response strategy for user i and analyze the impact of the different strategies on i's anonymity under this refined scenario.

4.1 Re-modeling and the New Equilibrium

Our technical development proceeds *mutatis mutandis* as in the previous section. In particular, as before we first model the interactions between users and attackers building on our previous model, then we study the equilibrium from which find the best response for the users, and finally, we analyze the results.

Now, let L_i' denote the anonymity loss which user i suffers if she is attacked by a malicious user. When i encounters a failed node, she may think it as a malicious node (user) and then remove it. Thus she suffers the anonymity loss if the node is actually honest user. We use L_i (which is described in our model design section) to denote this anonymity loss incurred by i when she removes a normal honest user from router list. Then the utility function of user i becomes as follows.

Proposition 3. *The utilities of user i under different strategy pairs are evaluated as follows.*

$$U_i(t+1) \;=\; U_i(t) - \begin{cases} P_{ik}(h \mid F)L_i & (S_1, S_1') \\ \Big(P_{ik}(h \mid F)L_i + P_{ik}(t \mid F)(L_{ik'} + L_i)\Big) & (S_1, S_2') \\ \Big(P_{ik}(m \mid F)L_i' + P_{ik}(t \mid F)L_{ik}\Big) & (S_2, S_1') \\ \Big(P_{ik}(m \mid F)L_i' + P_{ik}(t \mid F)L_{ik'}\Big) & (S_2, S_2') \end{cases}$$

Proof. When strategy pair (S_1, S_1') is adopted, that is user i chooses to remove k from router list and the attacker still targets k, we subtract the anonymity loss of removing an honest user with probability $P_{ik}(h \mid F)$. When i still chooses to remove k but the attacker targets a different node k', then we first subtract the loss L_i that k is a normal honest

user with probability $P_{ik}(h \mid F)$ from the original utility $U_i(t)$. Then, with probability $P_{ik}(t \mid F)$ we subtract the loss exerted by a successfully DoS attack ($L_{ik'}$) and that of removing k (L_i). Here we omit the proof of the utilities under the last two strategy pairs because they are similar to the first two. □

We then start studying the equilibrium of the refined game model.

Proposition 4. *For the DoS attacker, if all the targets look alike, that is $L_{ik} = L_{ik'}$ and $C_k = C_{k'}$, then the following two Pure Nash equilibriums hold.*

$$(S_1, S_2'), \quad \text{if } P_{ik}(m \mid F) > \frac{L_i}{L_i + L_i'};$$

$$(S_2, S_2'), \quad \text{if } P_{ik}(m \mid F) < \frac{L_i}{L_i + L_i'}.$$

Proof. Since $L_{ik} - C_k = L_{ik'} - C_{k'}$ and $L_{ik'} + L_i - C_{k'} > C_k$, the attacker will always choose strategy S_2'. Under this situation, user i will compare the two utilities of her two strategies S_1, S_2, and choose the one which brings her greater payoff. Thus we have that if

$$P_{ik}(m \mid F)L_i' + P_{ik}(t \mid F)L_{ik'} > P_{ik}(h \mid F)L_i + P_{ik}(t \mid F)(L_{ik'} + L_i),$$

then i should choose S_1, which consist of removing k from the router list. Otherwise, strategy S_2 should be chosen. From the formulae above and because $P_{ik}(t \mid F) + P_{ik}(m \mid F) + P_{ik}(h \mid F) = 1$, we obtain

$$P_{ik}(m \mid F) > \frac{L_i}{L_i + L_i'}.$$

□

The value $\frac{L_i}{L_i + L_i'}$ depends on the certain system. The greater the value of $\frac{L_i}{L_i + L_i'}$, the better the strategy S_2. Figure 4 shows that user i's best response strategy is influenced by the probability $P_{ik}(m \mid F)$ that k is malicious. Note that the value of $P_{ik}(m \mid F)$ is in the range

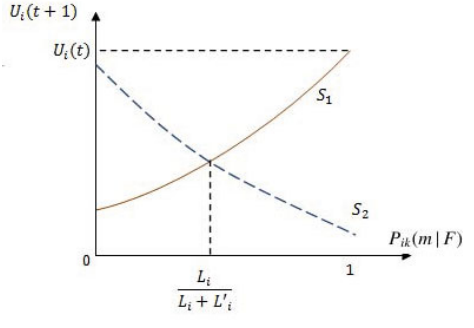


Fig. 4. Utilities $U_i(t+1)$ of user i choosing strategy S_1 and S_2 are influenced by the probability $P_{ik}(m \mid F)$ that k is a malicious user

$[0, 1]$. The two utilities intersect at the point $P_{ik}(m \mid F) = \frac{L_i}{L_i + L'_i}$, which brings i the same utility whatever strategy she chooses. When $P_{ik}(m \mid F) = 0$, i.e., failed node k cannot be malicious but either is the DoS target or a normal honest user, i should keep k in that $U_i(t+1)_{S_2}$ is greater than $U_i(t+1)_{S_1}$. However, when $P_{ik}(m \mid F)$ increases to 1, the utility of S_1 remains at $U_i(t)$ as before, while the utility of S_1 becomes $U_i(t) - L'_i$. Therefore, user i should remove k when k is more likely to be a malicious user, more precisely in the range $[\frac{L_i}{L_i + L'_i}, 1]$.

4.2 Predictions of $P_{ik}(m \mid F)$

Our model shows that $P_{ik}(m \mid F)$ is an important parameter which determines what strategy user i should choose. However, the exact value of $P_{ik}(m \mid F)$ is difficult to determine for i. User i cannot tell what type the failed node she encountered belongs to. In this section we therefore focus on $P_{ik}(m \mid F)$ so as to give i a way to approximate and predict its value.

The probability $P_{ik}(m \mid F)$ is a conditional probability representing that given i encounters a failed node, say k, then k is malicious. By Bayes Theorem, it can be computed as follows

$$P_{ik}(m \mid F) = \frac{\Pr[F \mid k = m]\Pr[k = m]}{\Pr[F]} \tag{3}$$

where $\Pr[F \mid k = m]$ is the conditional probability of encountering a failed node given the node is malicious, $\Pr[k = m]$ is the probability that node k is a malicious user, and $\Pr[F]$ is the probability of encountering a failed node.

We first study the probability $\Pr[F]$ in Eq. 3 which can be evaluated by applying the *total probability theorem*. There are three possibilities: the node is failed because (1) he is a normal honest user but suffering accidents; (2) he is an attacker; and (3) he is the DoS target. We indicate them by respectively h, m and t these possibilities. Thus we have:

$$\Pr[F] = \sum_{\alpha = h,m,t} \Pr[F \mid k = \alpha]\Pr[k = \alpha]\,. \tag{4}$$

For evaluation, $\Pr[k = \alpha]$ is determined by the composition of different types of users in the system. For instance, the probability that one node is malicious $\Pr[k = m]$ is evaluated by the ratio of malicious users c among all the users n:

$$\Pr[k = m] \;=\; \frac{c}{n}\,. \tag{5}$$

For other two types, by introducing θ–the percentage of target nodes among honest users–defined in Table 1, we have

$$\begin{aligned} \Pr[k = t] \;&=\; \frac{n - c}{n} \cdot \theta\,, \\ \Pr[k = h] \;&=\; \frac{n - c}{n} \;\cdot\; (1 \;-\; \theta)\,. \end{aligned} \tag{6}$$

As for the probabilities $\Pr[F \mid k = \alpha]$ where $\alpha = t, h, m$, since the target nodes are always failed to other users, the equation $\Pr[F \mid k = t] = 1$ always holds. Now from Eq. 3–6, we obtain

Proposition 5.

$$P_{ik}(m \mid F) = \frac{\Pr[F \mid k = m] \cdot \frac{c}{n}}{\Pr[F \mid k = m] \cdot \frac{c}{n} \; + \; 1 \cdot \frac{n-c}{n} \cdot \theta \; + \; \Pr[F \mid k = h] \cdot \frac{n-c}{n} \cdot (1-\theta)}$$

We then study the partial derivatives of Proposition 5 to get the relationships among $P_{ik}(m \mid F)$ and its parameters.

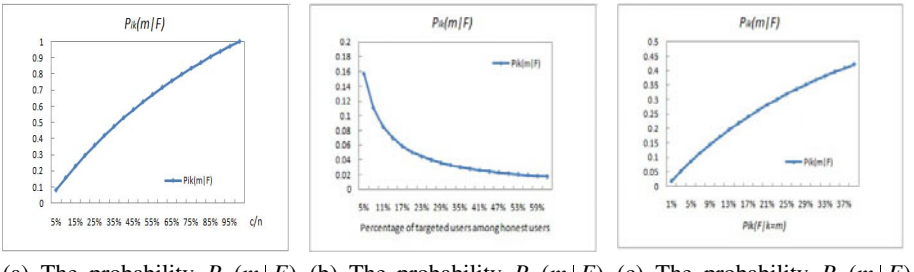
Corollary 2. *For user i, the following hold.*

$$\frac{\partial P_{ik}(m \mid F)}{\partial \Pr[k = m]} \geq 0\,, \quad \frac{\partial P_{ik}(m \mid F)}{\partial \Pr[k = h]} \leq 0\,, \quad \frac{\partial P_{ik}(m \mid F)}{\partial \Pr[F \mid k = m]} \geq 0\,, \quad \textit{and} \quad \frac{\partial P_{ik}(m \mid F)}{\partial \Pr[k = t]} \leq 0\,.$$

From the corollary above, one sees that if user i finds the system to consists of relatively more malicious users, then $P_{ik}(m \mid F)$ is more likely in the range of $[\frac{L_i}{L_i + L_i'}, 1]$, thus it is better for her to remove failed nodes. When the probability $\Pr[k = m]$ increases to one, the probability $P_{ik}(m \mid F)$ increases to one as well.

Such a prediction of $P_{ik}(m \mid F)$ can be observed by evaluating the anonymity level guaranteed by the system, in that the anonymity level $P(o_j \mid a_i)$ reflects the portion (via $\Pr[k = m]$) of malicious users among all nodes. Consider an example where 10% of malicious users are found unavailable due to being busy at deanonymizing the systems, and 1% of normal honest users are observed as failed nodes because of overloading or other accidental reasons, then we have $\Pr[F \mid k = m] = 0.1$ and $\Pr[F \mid k = h] = 0.01$. We get Figure 5(a), 5(b) depicting the changes of $P_{ik}(m \mid F)$ by varying $Pr[k = m]$ from 5% to 100%, increasing the percentage θ of targeted users among honest users respectively. These results are proved in Corollary 2 as well.

The probabilities $\Pr[F \mid k = \alpha]$ where $\alpha = h, m, t$ depend on the attacks and the observations of the system. Since the nodes targeted by DoS attackers always appear as failed nodes, the equation $\Pr[F \mid k = t] = 1$ always holds. The value of $\Pr[F \mid k = h]$ can be learned by observing how often normal honest users appear to have 'failed'; this should normally be relative small because not many normal users will suffer overloading, or other network accidents. As for $\Pr[F \mid k = m]$, it is usually small due to that the



(a) The probability $P_{ik}(m \mid F)$ increases when the percentage of malicious users increases.

(b) The probability $P_{ik}(m \mid F)$ decreases when θ increases.

(c) The probability $P_{ik}(m \mid F)$ decreases when $\Pr[F|k = m]$ increases.

Fig. 5. The value of $P_{ik}(m \mid F)$

malicious users do not want to be noticed or detected of doing de-anonymizing things. In Figure 5(c), we have $\Pr[F \mid k = m]$ vary from 1% to 37%, $c/n = 10\%$ and $\theta = 5\%$. When $\Pr[F \mid k = m]$ is quite small, $P_{ik}(m \mid F)$ is very small and thus under this case, $P_{ik}(m \mid F)$ is more likely in the range of $[0, \frac{L_i}{L_i+L'_i}]$. Therefore, keeping k in the router list is the best strategy for i.

5 Conclusion

In this paper we have investigated the best response for users to minimise their anonymity loss when they come across 'failed' nodes under DoS attacks. We used a game-theoretic approach for our analysis.

We modelled the problem and formalised the payoffs of users and attackers according to the strategies they choose. By Nash Equilibria, we showed that in a symmetric protocol like Crowds, keeping failed node is the strategy users should choose. We then re-modelled the problem by taking into account that the user's uncertainty about the typology of failed nodes they encounter. Our results showed that when the important parameter $P_{ik}(m \mid F)$ is in the range of $[\frac{L_i}{L_i+L'_i}, 1]$, users should remove the failed nodes and when it is smaller than $\frac{L_i}{L_i+L'_i}$, the best strategy is instead to retain them. We proposed a way to predict the value of $P_{ik}(m \mid F)$ and showed its changes when the parameters vary.

References

1. Backes, M., Lorenz, S., Maffei, M., Pecina, K.: Anonymous Webs of Trust. In: Atallah, M.J., Hopper, N.J. (eds.) PETS 2010. LNCS, vol. 6205, pp. 130–148. Springer, Heidelberg (2010)
2. Bauer, K., McCoy, D., Grunwald, D., Kohno, T., Sicker, D.: Low-resource routing attacks against tor. In: Proceedings of the 2007 ACM Workshop on Privacy in Electronic Society, WPES 2007, pp. 11–20. ACM, New York (2007)
3. Borisov, N., Danezis, G., Mittal, P., Tabriz, P.: Denial of service or denial of security? In: Proceedings of the 14th ACM Conference on Computer and Communications Security, CCS 2007, pp. 92–102. ACM, New York (2007)
4. Chatzikokolakis, K., Palamidessi, C.: Probable innocence revisited. Theor. Comput. Sci. 367(1-2), 123–138 (2006)
5. Chaum, D.: Untraceable electronic mail, return addresses, and digital pseudonyms. Commun. ACM 24(2), 84–88 (1981)
6. Damiani, E., di Vimercati, S.D.C., Paraboschi, S., Pesenti, M., Samarati, P., Zara, S.: Fuzzy logic techniques for reputation management in anonymous peer-to-peer systems. In: Wagenknecht, M., Hampel, R. (eds.) Proceedings of the 3rd Conference of the European Society for Fuzzy Logic and Technology, pp. 43–48 (2003)
7. Dingledine, R., Freedman, M.J., Hopwood, D., Molnar, D.: A Reputation System to Increase Mix-net Reliability. In: Moskowitz, I.S. (ed.) IH 2001. LNCS, vol. 2137, pp. 126–141. Springer, Heidelberg (2001)
8. Dingledine, R., Mathewson, N.: Anonymity loves company: Usability and the network effect. In: Proceedings of the Fifth Workshop on the Economics of Information Security, WEIS 2006 (2006)

9. Dingledine, R., Mathewson, N., Syverson, P.F.: Tor: The second-generation onion router. In: USENIX Security Symposium, pp. 303–320. USENIX (2004)
10. Dingledine, R., Syverson, P.F.: Reliable MIX Cascade Networks through Reputation. In: Blaze, M. (ed.) FC 2002. LNCS, vol. 2357, pp. 253–268. Springer, Heidelberg (2003)
11. Freedman, M.J., Morris, R.: Tarzan: a peer-to-peer anonymizing network layer. In: Atluri, V. (ed.) ACM Conference on Computer and Communications Security, pp. 193–206. ACM (2002)
12. Fudenberg, D., Tirole, J.: Game Theory. MIT Press (1991)
13. Golle, P., Juels, A.: Parallel mixing. In: Atluri, V., Pfitzmann, B., McDaniel, P.D. (eds.) ACM Conference on Computer and Communications Security, pp. 220–226. ACM (2004)
14. Hopper, N., Vasserman, E.Y., Chan-Tin, E.: How much anonymity does network latency leak? ACM Trans. Inf. Syst. Secur. 13(2) (2010)
15. Jakobsson, M.: Flash mixing. In: Annual ACM Symposium on Principles of Distributed Computing, PODC 1999, pp. 83–89 (1999)
16. McLachlan, J., Tran, A., Hopper, N., Kim, Y.: Scalable onion routing with Torsk. In: Al-Shaer, E., Jha, S., Keromytis, A.D. (eds.) ACM Conference on Computer and Communications Security, pp. 590–599. ACM (2009)
17. Murdoch, S.J., Danezis, G.: Low-cost traffic analysis of tor. In: IEEE Symposium on Security and Privacy, pp. 183–195. IEEE Computer Society (2005)
18. Nambiar, A., Wright, M.: Salsa: a structured approach to large-scale anonymity. In: Juels, A., Wright, R.N., di Vimercati, S.D.C. (eds.) ACM Conference on Computer and Communications Security, pp. 17–26. ACM (2006)
19. Neff, C.A.: A verifiable secret shuffle and its application to e-voting. In: ACM Conference on Computer and Communications Security, pp. 116–125 (2001)
20. Øverlier, L., Syverson, P.F.: Improving Efficiency and Simplicity of Tor Circuit Establishment and Hidden Services. In: Borisov, N., Golle, P. (eds.) PET 2007. LNCS, vol. 4776, pp. 134–152. Springer, Heidelberg (2007)
21. Pappas, V., Athanasopoulos, E., Ioannidis, S., Markatos, E.P.: Compromising Anonymity Using Packet Spinning. In: Wu, T.-C., Lei, C.-L., Rijmen, V., Lee, D.-T. (eds.) ISC 2008. LNCS, vol. 5222, pp. 161–174. Springer, Heidelberg (2008)
22. Ray, S., Slutzki, G., Zhang, Z.: Incentive-driven P2P anonymity system: A game-theoretic approach. In: ICPP, p. 63. IEEE Computer Society (2007)
23. Reiter, M.K., Rubin, A.D.: Crowds: Anonymity for web transactions. ACM Trans. Inf. Syst. Secur. 1(1), 66–92 (1998)
24. Sassone, V., Hamadou, S., Yang, M.: Trust in Anonymity Networks. In: Gastin, P., Laroussinie, F. (eds.) CONCUR 2010. LNCS, vol. 6269, pp. 48–70. Springer, Heidelberg (2010)
25. Singh, A., Liu, L.: Trustme: Anonymous management of trust relationships in decentralized P2P systems. In: Shahmehri, N., Graham, R.L., Caronni, G. (eds.) Peer-to-Peer Computing, pp. 142–149. IEEE Computer Society (2003)
26. Wang, Y., Vassileva, J.: Trust and reputation model in peer-to-peer networks. In: Shahmehri, N., Graham, R.L., Caronni, G. (eds.) Peer-to-Peer Computing. IEEE Computer Society (2003)
27. Zhuang, L., Zhou, F., Zhao, B.Y., Rowstron, A.I.T.: Cashmere: Resilient anonymous routing. In: NSDI. USENIX (2005)