PRICE: PRivacy preserving Incentives for Cooperation Enforcement

Leucio Antonio Cutillo Refik Molva Melek Önen
EURECOM
Sophia Antipolis
France
Email: {cutillo, molva, onen}@eurecom.fr

***Abstract*—Many incentive mechanisms have been proposed to foster cooperation among nodes in Peer-to-Peer (P2P) networks. Unfortunately, most of existing solutions rely on the existence of an online centralized authority that is in charge of a fair distribution and transaction of credits (incentives) between peers. Such centralized mechanisms mainly suffer from privacy leakage and single point of failure problems. To cope with these problems, we propose to take advantage of the distributed nature of P2P networks in order for the peers to take care of credit-based operations. Cheating and other DoS attacks are prevented thanks to a threshold security mechanism where the operation should be approved by a predefined certain number of peers. The main novelty of the proposed mechanism is the fact that a "credit" is assigned to some peers using distributed hash tables, hence, peers can follow and control the history of operations with respect to this credit, only. Thanks to this new approach, a malicious node cannot easily keep track of all operations originating from a single node and the impact of cheating or similar attacks would be strongly reduced.**

***Keywords*-Peer-to-peer; incentive mechanisms;**

I. Introduction

Peer-to-Peer systems are nowadays broadly adopted to provide several services such as file sharing [1], [2], data storage [3], [4], secure communication [5], [6], social networking [7], [8], [9]. The correct execution and the availability of such services strongly depend on the collaboration among peers. Several studies [10], [11], [12] pointed out that, unfortunately, peers often engage in ***free riding***, i.e. they try to consume as more resources as possible and on the contrary, contribute with as few resources as possible. This kind of selfish behavior has a strongly negative impact on the overall performance of the system and may even lead to its failure.

Several cooperation enforcement solutions [13], [14], [15], [16], [17], [18], [19] have been proposed to foster cooperation among peers in P2P or Mobile Ad-hoc Networks (MANETS). Most of them rely on credit-based mechanisms whereby nodes receive a reward whenever they cooperate for the execution of the requested action. These credit-based incentive mechanisms often rely on the existence of a centralized authority that ensures a fair distribution of the credits and also acts as a mediator in case of litigation during transactions. The adoption of such a centralized authority raises serious security and privacy concerns: Indeed, this online trusted authority has a direct access on the history of actions of any peer since it is in charge of distributing rewards corresponding to each granted service. Therefore, as in all existing centralized services, current credit-based incentive mechanisms suffer from privacy problems such as traceability or monitoring [18], [19].

In this paper, we propose PRICE, an incentive mechanism that can be adopted as a built-in service for any DHT-based P2P system. PRICE takes advantage of the distributed nature of the P2P network itself to manage credits and ensures the correctness and the security of transactions. The management of credits, defined as "coins" in PRICE, is distributed among peers in the network based on the use of the inherent P2P functionality, that is, a distributed hash table (DHT). A coin transaction only succeeds if a quorum among a pre-defined number of peers agrees on it. Although the task of credit management is distributed among several peers and therefore it can decrease the privacy of the system, PRICE offers an original approach by assigning the management of each single coin to a different set of peers instead of the account of a given peer. Therefore, on the one hand, no entity in the system is able to discover the total amount of credits a user holds; hence, as opposed to centralized solutions, a user's history of actions cannot be traced by any node; on the other hand, even if there is a privacy leakage with respect to a single transaction, this will not a have an impact on the privacy of the user's overall actions. The association between the credit involved in the transaction and the peers that are responsible for the transaction itself is based on the pseudorandomness of the security functions used for the generation of the coins.

The rest of this paper is organized as follows: section II introduces the main security and privacy challenges of an incentive mechanism. Section III proposes an overview of PRICE which is, then, formally described in section IV. The evaluation of PRICE is discussed in section V.

II. Problem Statement

A. Cooperation enforcement in P2P networks

The correct execution of many P2P services relies on the collaboration of nodes involved in the network. Cooperation enforcement mechanisms would encourage nodes to perform a fair share of basic operations. Inducing cooperation

between nodes can be based either on some reputation or rewarding mechanisms. Reputation mechanisms [13], [14], [20], [21] ensure that each node accepts to cooperate with its neighbors based on the past behavior of the latters. On the other hand, credit based schemes [22], [16], [18], [19] provide node collaboration by rewarding cooperating nodes with a certain amount of credits that they further can use for their own benefit. Credits can be in the form of E-cash [23], [24] or a tradable good/service such as future cell phone call time.

B. Credit-based incentive mechanisms

Existing rewarding mechanisms encourage nodes to cooperate in performing the required operations (forwarding, data storage, etc.). These solutions consist of virtual currencies that nodes receive whenever they cooperate. Unfortunately, because such solutions suffer from lack of fairness, they require the existence of a centralized online trusted third party mainly for credit management. Indeed, for example, in [16], the rewarding mechanism named as Sprite requires an immediate reachability of the TTP defined as Credit Clearance Service (CCS). Such mechanisms also suffer from the single point of failure problem as nodes must contact the CCS whenever they forward the message in order to receive their rewards. Furthermore, this centralized entity has full control over these rewards and keeps track of any node's actions.
Distributed credit-based incentive mechanisms such as Karma [18] solve the single point of failure problem, since a set of peers in the DHT, namely the bank, stores a user's account. Still, this set of nodes can trace the user's actions.

C. Security and Privacy Challenges

As for any credit-based mechanism, a credit based incentive mechanism should prevent nodes from cheating. Therefore the proposed mechanism should exhibit the following security properties:

- ***unforgeability***: a valid credit cannot be forged by any user;
- ***no double spending***: credits resulting from duplication or copying of valid credits should be prevented or immediately detected;
- ***communication confidentiality***: any action taken under the incentive mechanism should not leak information regarding the underlying service application;
- ***transaction untraceability***: a selfish or malicious user should not be able to monitor any legitimate user's account.

Existing cooperation enforcement schemes either rely on tamperproof hardware or on the existence of an online trusted third party that is responsible for the case of possible litigation. Unfortunately, such solutions are either costly or suffer from the problem of both privacy leakage and single point of failure. Therefore, the new incentive mechanism should provide the previously listed guarantees without the help of a centralized authority.

III. Solution Overview

In order to cope with the previously described security and privacy challenges we propose PRICE, a credit-based incentive mechanism whereby, as opposed to existing centralized solutions, the management of the credits, defined as ***coins***, is distributed among several peers in the network. While this distributed mechanism allows a better robustness of the system and prevents the problem of single point of failure, the privacy challenge becomes even more important since many peers can be aware of others' activities. The proposed management of coins hence prevents such a possible leakage by assigning different sets of peers for each coin rather than defining one responsible per node's account (activities). The peer assignment follows the inherent nature of P2P by taking advantage of distributed hash tables (DHT). In the following sections, the proposed mechanism is summarized and illustrated with a scenario.

A. Environment

As previously mentioned, the correct execution of PRICE relies on the use of a distributed hash table (DHT) based P2P network where every peer node is also considered as the application user. A peer is assigned to a unique identity, the ***Peer Identifier***, and the assignment of coins to peers is managed by the DHT: in addition to P2P services such as data storage or data retrieval, peers also participate on the management of coins. To prevent DoS attacks including Sybils [25] or eclipse [26] the mechanism defines an off-line ***Trusted Identification System*** (TIS) which mainly computes the Peer Identifier and ensures that this value is unique and is assigned to its corresponding peer by generating a cryptographic certificate over the identifier. Any attack due to the multiple identities creation or identity manipulation is thus unfeasible in PRICE. In order to ensure the security of the rewarding mechanism, coins are generated and signed by a trusted entity named as ***Coin Generator*** (CGEN) whose unique role is to ensure the correctness and validity of the coin.

B. Scenario

In the following, we present a scenario in which two users Alice and Bob take part in a P2P network offering data storage services and use PRICE to manage their transactions. In the P2P network, let Alice be a user interested in storing her file using Bob's resources. Whenever Alice sends her request to Bob, she grants him with a ***coin*** for this additional service. This transfer should of course be considered as valid and Bob should be able to verify that he is the new owner of the coin. Therefore there is a strong need for defining a third entity or a witness to validate such a transfer. In the proposed mechanism, a set of peers is assigned for this role

and they are defined as ***notaries***. The track of each coin is kept by a different set of notaries. Therefore, whenever Alice would like to grant a coin to Bob, she contacts one of the notaries corresponding to this specific coin, namely the ***caretaker notary***, and informs it about the new ownership of the coin. With the agreement of the other notaries, the caretaker then sends a proof of this transfer to Bob. Even if a malicious node succeeds in discovering current transfer of this coin, it will not be able to trace all actions taken by Alice or Bob since the management of each coin is assigned to different notaries.

Based on this scenario which is illustrated in figure 1, we identify three main steps of the proposed incentive mechanism:

- ***account creation***, whereby a newcomer receives his peer identifier (PI) from the TIS and an initial number of coins from the CGEN;
- ***payment order***, whereby the newcomer requests to grant a coin to a beneficiary by sending a PAY message to the caretaker notary;
- ***payment notification***, whereby the caretaker notary collects the agreement of a sufficient number of notaries, and informs the payer and the beneficiary about the success of the transaction.

IV. Description

In this section, we first define the security properties of a coin and introduce the main components of PRICE which are the Coin Generator, the DHT based P2P substrate, and the Trusted Identification Service. We then formally describe the three steps of the proposed incentive mechanism, namely, **the account creation**, **the payment order** and the **payment notification**.

A. Preliminaries

1) The P2P substrate and the Trusted Identification Service: In the DHT, every user is associated to a peer node by a unique Peer Identifier PI which is computed by the TIS. By granting a certificate together with every identifier, the TIS protects the PRICE mechanism from different DoS attacks such as Sybil [25], impersonation, or eclipse[26]. Following the very definition of a DHT, a PI is defined as a number over a "key space" in order to facilitate the functions of data lookup.

2) The rewarding mechanism and the Coin Generator: PRICE relies on a specific implementation of rewards which are named as ***coins***, generated by a trusted entity, the ***Coin Generator***, and are defined by a tuple with the following parameters:

- a ***Coin Identifier*** CI, which is a pseudo-random number generated by the Coin Generator over the keyspace K and will be used as the input of a **coin lookup** operation in the P2P network;
- the signature of CI computed by the Coin Generator with its secret key as a proof of the validity of the coin.

Thanks to the security of the pseudo-random generator used by the CGEN, a coin c is unique. The signature of the CGEN provides the protection against forging attacks.

Table I summarizes the notation used for the description of PRICE.

Table I
Notation

$\mathcal{A}$	node $\mathcal{A}$
$PI_{\mathcal{A}}$	peer identifier of $\mathcal{A}$
$K^-_{\mathcal{A}}, K^+_{\mathcal{A}}$	private and public keys of $\mathcal{A}$
$\{\cdot\}S_{\mathcal{A}}$	signature generated with the private key of $\mathcal{A}$
$Cert(I, K^+)$	certificate associating an identifier I to a public key K^+
MK	master key
$h_{MK}(\cdot)$	keyed hash function with master secret MK
$E_{\mathcal{B}}\{M\}_{S_{\mathcal{A}}}$	message M signed by $\mathcal{A}$ and encrypted for $\mathcal{B}$
c	coin c
CI_c	coin identifier of c
$\mathcal{CR}(CI_c)$	coin registry of c
$\mathcal{NS}(CI_c)$	notary set of c
K	DHT keyspace
N	set of all the peer nodes in the DHT
R	set of all the resources stored in the DHT
C	set of all the coins in the DHT
$id_x(x)$	map of x to an identifier in K
$\rho(x)$	responsibility function mapping x to a set $\{PI\}$
q	number of coins every notary is responsible for
w	number of welcome coins granted to a newcomer

B. Account creation

Whenever a new user enters the system, it first needs to receive its peer identifier and its set of initial coins. Therefore, the account for a new user $\mathcal{A}$ is created in three separate steps: 1) identity creation and authentication, where $\mathcal{A}$ obtains its identifier, 2) P2P substrate join, where $\mathcal{A}$ takes its place in the DHT, and 3) welcome coin attribution, where $\mathcal{A}$ is granted with a predefined number of coins by the CGEN.

1) Identity creation: In order to get its peer identifier, $\mathcal{A}$ generates an asymmetric keypair $\mathcal{K}_{\mathcal{A}} = \{K^-_{\mathcal{A}}, K^+_{\mathcal{A}}\}$ and sends an out of band request to the TIS. This request contains $\mathcal{A}$'s public key $K^+_{\mathcal{A}}$, together with his claimed identity $ID_{\mathcal{A}}$. Once this request is received, the TIS generates $\mathcal{A}$'s peer identifier as $PI_{\mathcal{A}} = h_{MK}(ID_{\mathcal{A}})$, where $h_{MK}(\cdot)$ is a keyed hash function whose master secret MK is known only by the TIS and nobody else. The TIS sends back to $\mathcal{A}$ the certificate $Cert(PI_{\mathcal{A}}, K^+_{\mathcal{A}})_{S_{TIS}}$ and informs the coin generator a new user has joined the system.

2) Welcome coins attribution: As the CGEN receives a message from the TIS stating a new user $\mathcal{A}$ has arrived, it generates a new set of coins $\{c_i\}$, and provides $\mathcal{A}$ with this set by sending him a PAY message for every coin signed by the CGEN itself. $\mathcal{A}$ can collect CGEN's coins by sending these messages to the DHT. The signature of

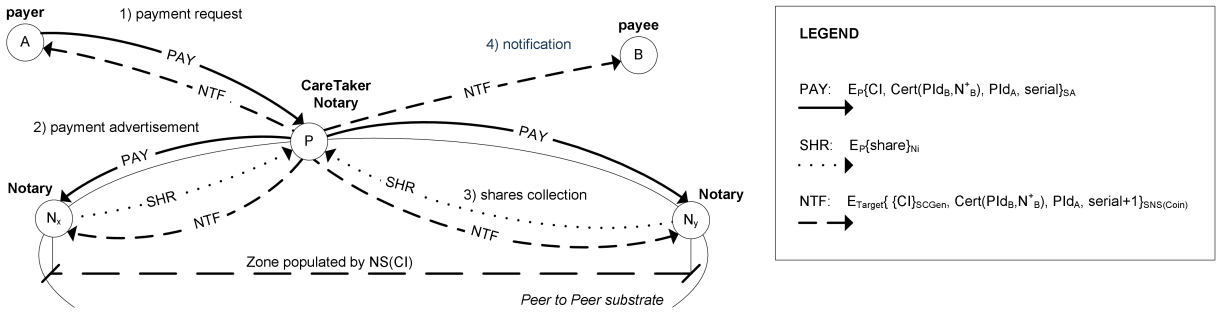

Figure 1. Payment scheme.

the CGEN prevents a malicious user from modifying the PAY messages and steal the welcome coins by changing the beneficiary.

Once $\mathcal{A}$ has successfully received its coins, it can join the P2P system and actively participate to any application or service offered by the P2P network and use PRICE for transactions accordingly.

3) P2P substrate join: On reception of the certificate, $\mathcal{A}$ joins the P2P substrate, and contacts other peers to advertise its presence and populate its routing table following usual P2P protocols. It also finds out the identities of the notaries corresponding to each of its coins using a map function ρ which, as an input of the identity of the coin c, outputs the set of peer identities responsible for its management, that are, the notaries. Upon reception of the initial coin, a caretaker notary, adds in its current coin registry the following information:

- the coin identifier,
- the signature of the CGEN,
- $\mathcal{A}$'s certificate,
- $\mathcal{A}$'s peer identifier which further will be replaced by the previous owner of the coin at each transaction of this coin,
- a serial number which is used to synchronize notaries and prevent replay attacks,
- a group signature generated by a subset of dedicated notaries.

C. Payment order

In order for $\mathcal{A}$ to transfer a coin c to $\mathcal{B}$, $\mathcal{A}$ has to indicate to the P2P system the new owner of the actual coin. This action takes place in two steps: 1) notary lookup, 2) payment request.

1) Notary lookup: In this step, $\mathcal{A}$ performs a lookup in the DHT using the coin identifier CI_c as a lookup key. In O(log (n)) steps, $\mathcal{A}$ reaches a node $\mathcal{P}$ in the notary set of the coin. $\mathcal{P}$ will act as the caretaker of the transaction $\mathcal{A}$ is going to make.

2) Payment request: In order for $\mathcal{A}$ to grant a coin c to $\mathcal{B}$, $\mathcal{A}$ sends a signed payment message PAY to $\mathcal{P}$ containing the signed coin identifier CI_c proving c is a valid coin, the certificate of the new owner $\mathcal{B}$ proving $\mathcal{B}$ is a valid node in the system, $\mathcal{B}$'s IP address, and a serial number SN used to avoid replay attacks. This message is encrypted with $\mathcal{P}$'s public key to prevent eavesdropper from tracing the transaction. In case $\mathcal{A}$ is not the current owner of the coin or there is a mismatch between the serial number in the PAY message and that one in the coin registry, $\mathcal{P}$ simply discards the message, otherwise $\mathcal{P}$ forwards it to its neighborhood in the notary set $\mathcal{NS}(CI_c)$.

Please note that $\mathcal{P}$ is responsible for more than a single coin in the system and can receive several PAY messages for several coins from different users at the same time.

D. Payment notification

In order for the payment to succeed, a predefined quorum among the notaries of c has to agree on the update (or creation) of the entry associated to c in the coin registry performed by the caretaker $\mathcal{P}$. Once this agreement is met, the caretaker can notify the payer, the beneficiary and the notaries about the success of the transaction. These actions take place in two steps: 1) coin registry update, 2) payment confirmation.

1) Coin registry update: Every node in the DHT stores a coin registry $\mathcal{CR}$ keeping the association between every coin identifier it is responsible for and the peer identifier of the current owner of that coin. An entry in the coin registry has the form:

$$\mathcal{CR}(CI_c) = \left\{CIc, PI_{\mathcal{Z}}, Cert(PI_{\mathcal{A}}, K^{+}_{\mathcal{A}}), SN\right\}_{S_{\mathcal{NS}(CI_c)}}$$

where $Cert(PI_{\mathcal{A}}, K^{+}_{\mathcal{A}})$ identifies the current owner of the coin and is used to verify the integrity of PAY messages, $PI_{\mathcal{Z}}$ is the peer identifier of the previous owner of c, SN is a serial number used to refresh the coin registry of the nodes coming back online in the DHT and to avoid replay attacks, and $S_{\mathcal{NS}(CI_c)}$ is the group signature generated by a sufficient number of nodes in the notary set.

When a notary $\mathcal{N}_j$ receives a forwarded PAY message from $\mathcal{P}$, $\mathcal{N}_j$ checks the integrity of PAY, and computes on a temporary updated version of $\mathcal{CR}(CI_c)$ its own share

$Share_j$ to be sent back to $\mathcal{P}$ through an SHR message. If a predefined quorum among a representative group of $\mathcal{NS}(CI_c)$ is reached, $\mathcal{P}$ computes the group signature $S_{\mathcal{NS}(CI_c)}$, updates $\mathcal{CR}(CI_c)$ and advertises it along the notary set.

2) Payment confirmation: In case the group signature $S_{\mathcal{NS}(CI_c)}$ is generated, the transaction succeeds and $\mathcal{P}$ sends back both $\mathcal{A}$ and $\mathcal{B}$ a notification message NTF containing the updated coin registry entry $\mathcal{CR}(CI_c)$. The group signature in this entry proofs the correctness of the transaction and prevents a malicious notary from arbitrarily modifying its content.

V. Evaluation

In this section we evaluate the feasibility of PRICE with respect to the security and privacy challenges defined in section II.

We assume the DHT as follows:

$$DHT = \langle K, N, R, C, id_n(\cdot), id_r(\cdot), id_c(\cdot), \rho(\cdot) \rangle$$

K is the DHT keyspace, N, R and C correspond to the set of nodes, the set of resources and the set of coins, respectively. $id_n : N \rightarrow K$, $id_r : R \rightarrow K$ and $id_c : C \rightarrow K$, denote the functions respectively associating a node, a resource, a coin to their identifier. Finally, as previously defined $\rho : K \rightarrow \{N\}$ denotes the mapping function which outputs the set peers responsible given a resource. In particular, this responsibility function determines the notary set of a coin: $\rho : id_c \rightarrow \{\mathcal{NS}(CI_c)\}$. We will call ***k-bit zone*** the subset of the id space containing all the peers whose id agrees in the high order k bits.

A. Security

Coin integrity/unforgeability The integrity or unforgeability of a coin c is guaranteed thanks to the signature S_{CGEN} of the Coin GENerator authority that generated that coin. Such a signature cannot be computed by anybody else, as the private key of the CGEN is never disclosed.

Transaction integrity The integrity of a transaction involving a coin c is represented by the integrity of the record $\mathcal{CR}(c)$ in the coin registry, and is guaranteed by the group signature $S_{\mathcal{NS}(CI_c)}$. Therefore PRICE prevents the double spending of coins. Computing such a signature requires the collusion of a sufficient number of notaries and can therefore be mitigated by increasing the minimum notaries quorum at the expense of higher computation cost and communication overload.
Moreover, to keep fresh versions of the coin registry, a serial number in every entry of the coin registry helps a notary to come back online to update his registry from the other notaries.

Identifiers integrity In PRICE, peers receive their peer identifier PI from the TIS as an output of a one-way function $h_{MK}(\cdot)$ over their real identity ID. Since the secret MK used in the keyed hash function $h_{MK}(\cdot)$ is known by the TIS only, identifiers cannot be arbitrarily computed or guessed by any user. Moreover, the account creation procedure can be repeated several times but the result always leads to the same identifier. Therefore, even though certificates can be re-issued, peer identifiers never change. This prevents any malicious user from stealing a legitimate user's identity, or from creating different identities, namely Sybils, and launch Denial of Service attacks.

B. Privacy

Data confidentiality In PRICE, PAY and NTF messages are encrypted with the recipient's public key found in its certificate signed by the TIS. The user's private and public keys are computed by the user himself at the act of the account creation, and the private key is never disclosed. In case the private key is stolen, the certificate can be re-issued.

Anonymity As the TIS and the CGEN are separate entities, nobody can link a coin identifier to a real user's identity. In fact, the TIS is the only party being able to link a peer identifier to a real identity, but it does not hold any information about that user's coins. On the other hand, the CGEN distributes welcome coins to new peers, but it does not manage identity information. In case the TIS and the CGEN services are merged, no information rather than the *initial* association between coins and users can be derived. In fact, both the TIS and the CGEN are off-line services contacted only once by each legitimate user and do not play any role neither in communication nor in data management. Perhaps, they can be built in a distributed fashion.
In the DHT, a caretaker $\mathcal{P}$ can link a coin identifier CI_c to the owner's peer identifier $PI_{\mathcal{A}}$ for all the coins $\mathcal{P}$ is responsible for. Anyway, this does not reveal the owner $\mathcal{A}$'s real identity to $\mathcal{P}$, as no information about $PI_{\mathcal{A}}$ can be retrieved from $\mathcal{A}$'s certificate $Cert(PI_{\mathcal{A}}, K^+_{\mathcal{A}})_{S_{TIS}}$.

Transaction untraceability In PRICE, the number of coins held by a user $\mathcal{A}$ and the history of all the transactions $\mathcal{A}$ did in the system is known by $\mathcal{A}$ and no one else. A single coin transaction can be traced by the notary set of that coin. However, this does not reveal anything about the other transactions of the same actor $\mathcal{A}$. Moreover, due to the security of the pseudo-random function used by the CGEN to generate a coin c, the association mapped by $\rho(\cdot)$ between the coin registry entry $\mathcal{CR}(CI_c)$ and a notary $\mathcal{N}_j \in \mathcal{NS}(CI_c)$ responsible for it is also random.

C. Performance

In this section, we provide and evaluation of the performance of PRICE in terms of latency, storage and bandwidth consumption. In the following, we will consider Kad [1] as the underlying P2P overlay.

Latency The total transaction time T for a coin c can be seen as the sum of the time T_L required to the payer $\mathcal{A}$ for looking up for a coin identifier, the time T_R for transferring the PAY message, the time T_F required for the caretaker $\mathcal{P}$ to forward PAY along the notary set, the time T_S to collect the shares in order to compute the group signature and, finally, the time T_C required for confirming the payment:

$$T = T_L + T_R + T_F + T_S + T_C \quad (1)$$

In this formula, T_R can be negligible and T_F, T_S and T_C correspond to a one-hop Round Trip Time T_{RTT} in the DHT; hence T depends on T_L and T_{RTT}.

T_{RTT} and T_L are defined as random variables and are set to the values originated from real measurements on Kad conducted in [27]. T is then evaluated with Monte Carlo techniques based on these measurements. A set of 10000 samples t_i is computed as follows: we generate 4 uniform random variables between 0 and 1, namely y_l, y_{r1}, y_{r2}, y_{r3} and sum the inverse $F^{-1}_{T_L}$ and $F^{-1}_{T_{RTT}}$ of the cumulative distributions F_{T_L}, $F_{T_{RTT}}$ at those points.

The results are shown in figure 2, and table II summarizes the main statistics. As one can see, even if 50% of transactions require less than 7.5 seconds, still 10% of them succeed in more than 12.3 seconds. Decreasing the significant contribution of T_L by the use of central indexing services may speed up the transaction time at the expense of a lower privacy protection.

Table II
TIME STATISTICS IN SECONDS FOR THE THREE MAIN DISTRIBUTIONS IN FIGURE 2

	Average	50th percentile	90th percentile
T_{RTT}	0.664	0.287	1.50
T_L	6.51	5.64	8.87
T	8.47	7.48	12.3

Storage overhead An entry in the coin registry contains the coin id, the signature of the CGEN stating this coin is valid, the current owner's certificate, the old user's peer identifier, a serial number, and finally the notary set group signature validating the correct association between the coin and its current owner. Assuming a keyspace of 128 bits, a signature length of 512 bits, public keys of 512 bits and a 32 bits integers length, an entry requires 308 Bytes.
The number of coins q every peer is responsible for, and as a consequence the size required to store the coin registry, strongly depends on the number of peers and the number of coins in a notary set. Assume the id space is divided into 2^k zones, and in each of them peers and resources agree on the k high-order bits. Assume the responsibility function $\rho(CI_c, k)$ maps a coin c to the set of peers in the k-bit zone defined by k. In this case:

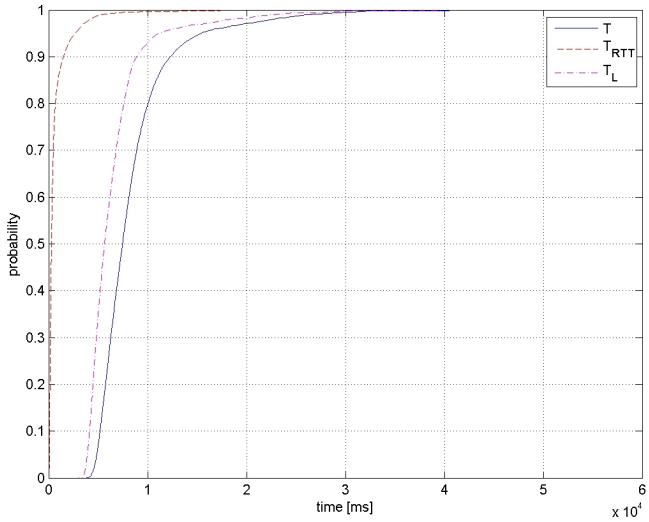

Figure 2. Total transaction time evaluation: T_{RTT} and T_L from [27], T from Monte Carlo techniques (10000 samples).

$$q = \frac{\|N\|}{2^k} w \quad (2)$$

where $\|N\|$ is the cardinality of the set N, i.e. the total number of peers in the system, and w is the number of welcome coins every peer receives at the very first join. Table III shows the size every peer should allocate, on average, to store its coin registry in a network of 5.12 millions peers[1] and where each node initially receives 100 welcome coins. When k is set to 8, then 20,000 peers populate a zone, and can act as notaries for a maximum of 2 millions of coins. Their coin registry can then reach a maximum size of 587 MB. By increasing k to 16, the number of coins every peer is responsible for decreases to 7800, leading the size of a coin registry to 2.29 MB.

Table III
COIN REGISTRY SIZE IN MB FOR DIFFERENT VALUES OF k

k [bit]	8	10	12	14	16
$\mathcal{CR}$ [MB]	587.46	146.87	36.72	9.18	2.29

Communication Bandwidth Overhead In order to evaluate the communication overhead, we first evaluate the minimum number of peers t required to compute a group signature. This threshold number should be defined according to the underlying privacy and robustness challenges: t strongly depends on the ratio m of malicious users and the online probability p of nodes. t can therefore be computed as follows:

$$t = \frac{\|N\|}{2^k} * p * m + 1 \quad (3)$$

[1]Steiner et al.[28] observed between 12 and 20 thousand active peers in one 256-th of the entire KAD id space.

Figure 3 outputs the t values with respect to different m and p values where k is set to 16. t varies between 2 and 21 where both m and p take values between 0.1 and 0.5.

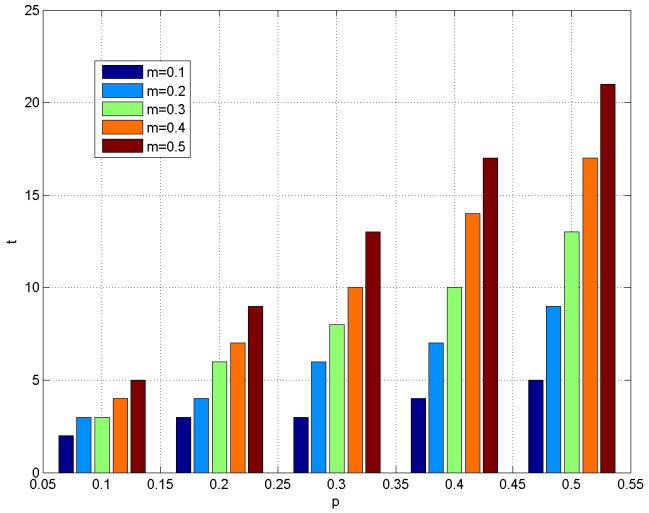

Figure 3. Evaluation of the number of notaries t to be contacted for every transaction for different online- p and misbehaving- m probabilities.

Each of these t notaries receives the PAY message forwarded by the caretaker $\mathcal{P}$, furhter computes the share of the group signature and sends it back to $\mathcal{P}$. Assuming a keyspace of 128 bits, a signature length of 512 bits, public keys of 512 bits and a 32 bits integers length, a PAY message requires 246 Bytes, while s' size is 64 Bytes. Once computed the whole group signature $S_{\mathcal{NS}(CI_c)}$, $\mathcal{P}$ sends a NTF message containing the coin registry entry $\mathcal{CR}(CI_c)$, whose size, according to the previous assumptions, is 308 Bytes. Assuming transactions occur every hour with a frequency λ, figure 4 shows that the bandwidth consumption is slightly less than 7Kbps when 100 transactions occurs every hour and 50 notaries have to agree on them.

VI. Related Work

A huge literature proposed credit-based mechanisms to stimulate cooperation in networks with the presence of selfish nodes[2].

In MANETS, credit-based incentive mechanisms were designed for enforcing the cooperation among nodes for the specific operation of *forwarding*. To achieve fairness, [22] was relying on tamper proof hardware whereas [16] defined a centralized on-line trusted entity.

PRICE does not focus on the nature of a specific operation and does not require any tamper proof hardware nor centralized entities to manage credits.

In the P2P scenario, authors in [18] proposed a micro-payment scheme where each peer is associated to a scalar

[2]i.e. nodes trying to maximize the benefits they get from the network while minimizing their contribution to it.

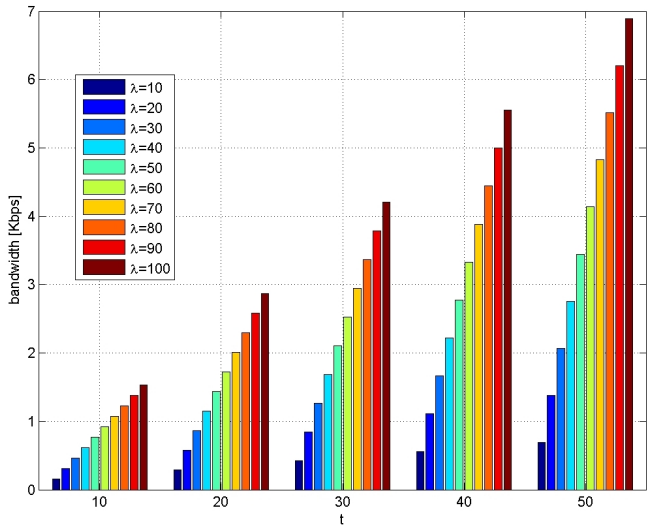

Figure 4. Evaluation of the bandwidth consumption for different transaction rates λ (per hour) and notaries to be contacted t.

value called *Karma*. A set of randomly chosen *bank set* nodes increase or decrease a peer's karma in case this peer contributes with- or consumes- resources. An atomic transaction scheme ensures fairness in the payment since the key to decrypt resources and certificate of receipt are provided simultaneously to the resource consumer and the provider respectively. When a file transfer occurs from a peer B to a peer A such a file is encrypted with a secret DES, then upon A's authorization, each member of A's bank set independently send a karma transfer request to all members of B's bank set, that in turn ask again A's bank set nodes for an acknowledgment. Once verified a majority quorum exists, B proceeds with the file transfer, and A provides B with a receipt. If B gets the receipt, A receives the key to decrypt the file.

In BitTorrent[3], a variant of "'tit-for-tat"' [29] mechanism encourages fairness in the exchange of file chunks. Such a mechanism aims at seeking pareto efficiency, meaning in this case that peers reciprocate uploading to peers which upload to them, aiming at having all the time several connections actively transferring data in both directions. In case of lack of reciprocity, a peer can temporarily refuse to upload a chunk to- , or *choke* a-, lazy peer. An optimistic unchoke mechanism, corresponding to always cooperating on the first move in prisoner's dilemma, solves the problem of discovering if current unused connections are better than the ones being used.

Criticisms against the incentive mechanisms in BitTorrent assert that its effectiveness is largely due to the altruistic behavior of a small number of altruistic nodes [30] and solutions like in [17] have been proposed to improve the

[3]http://www.bittorrent.com

overall system performance.

In Swift [17], peers exchanging file chunks are denoted as *traders* and employ a default trading strategy that is either good for them and for the network itself. Free riders are the most penalized in case of insufficient upload capacity to satisfy demand. Authors consider three strategies for traders and classify them accordingly: *paranoid*, *one-time risk-taking*, and *perioding risk-taking*. Paranoid traders are reciprocative players waiting for the reception of a valid chunk before offering to send an equal amount back, one-time risk-takers can offer free chunks to a peer never encountered before to encourage trading with the chance of receiving nothing in return, while periodic risk-takers give out free chunks periodically. Authors show that peers taking risks receive the most benefit in return, and deviating from the proposed default strategy of periodic risk-taking provides little or no advantage. Swift has then been added to the official BitTorrent client and named as *TradeTorrent*[4].

Finally, authors in [19] drew inspiration from BitTorrent to propose a P2P content distribution system based on endorsed e-cash [31] to provide accountability while preserving privacy in P2P systems. In such an approach, users can exchange files if they know the correct hashes on those files. In endorsed e-cash, users withdraw e-coins from a central bank maintaining all participants'accounts and spend them for digital content with a fair exchange protocol. In case a user gets paid, he must deposit e-coins in the bank before spending them again. A Trusted Third Party (TTP), namely the *arbiter*, is responsible for resolving disputes. Authors modify the endorsed e-cash protocol in [31] to allow the arbiter for resolving conflicts by examining a much shorter amount of data. Sybyl node creation is discouraged thanks to a mechanism in which newcomers are invited by friends and receive an initial credit from them.

PRICE extends the security an privacy features offered in [18], [17], [19] by revisiting the concept of bank account. As a main novelty of PRICE, in fact, no entity in the system can derive the total amount of credit a node currently holds, as accounts are made available for coins rather than for users. As an important consequence, there is no entity an attacker can target to discover one or more victim's account, and derive, for instance, its participation in the network.

VII. Conclusion

PRICE is a new cooperation enforcement mechanism which relies on credit-based incentives and takes advantage of the underlying DHT based P2P network to cope with security and privacy challenges. The task of coin management is distributed among several peers and in order to ensure transaction untraceability, PRICE assigns each single coin to a different set of notaries. The assignment function on the inherent functionality of a P2P network which is the DHT and the randomness of each assignment is ensured thanks to the security of the pseudo-random function used to generate the coin. The number of notaries is defined based on the ratio of malicious nodes and the average online probability and can have a direct impact on the robustness and performance of the P2P network. The communication overhead increases when more notaries are solicited for computing the threshold signature.

[4]http://mnl.cs.stonybrook.edu/project/tradetorrent/

Acknowledgment

This work has been supported by the PROSE project, grant agreement 2009 VERSO 07 03, funded by the French National Research Council (ANR).

References

[1] P. Maymounkov and D. Mazieres, "Kademlia: A Peer-to-Peer Information System Based on the XOR Metric," in *P2P-Systems*, vol. 2429, 2002, pp. 53 – 65.

[2] J. Pouwelse, P. Garbacki, D. Epema, and H. Sips, "The bittorrent p2p file-sharing system: Measurements and analysis," in *Peer-to-Peer Systems IV*, ser. Lecture Notes in Computer Science, M. Castro and R. van Renesse, Eds. Springer Berlin / Heidelberg, 2005, vol. 3640, pp. 205–216, 10.1007/11558989_19.

[3] A. Haeberlen, A. Mislove, and P. Druschel, "Glacier: Highly durable, decentralized storage despite massive correlated failures," in *Proceedings of the 2nd Symposium on Networked Systems Design and Implementation (NSDI'05)*, Boston, MA, May 2005.

[4] H. Weatherspoon, "Design and evaluation of distributed wide-area on-line archival storage systems," Ph.D. dissertation, Berkeley, CA, USA, 2006, aAI3254129.

[5] M. Rogers and S. Bhatti, "How to disappear completely: a survey of private peer-to-peer networks," in *Sustaining Privacy in Autonomous Collaborative Environments*, 2007.

[6] T. Chothia and K. Chatzikokolakis, "A survey of anonymous peer-to-peer file-sharing," in *Network-Centric Ubiquitous Systems*. Springer, pp. 744–755.

[7] L. A. Cutillo, R. Molva, and T. Strufe, "Safebook : a privacy preserving online social network leveraging on real-life trust," *IEEE Communications Magazine, Vol 47, N.12, Consumer Communications and Networking Series, December 2009*, 2009.

[8] S. Buchegger, D. Schiöberg, L. H. Vu, and A. Datta, "PeerSoN: P2P Social Networking," in *Social Network Systems*, Nürnberg, Germany, March 31 2009.

[9] A.-M. N. Mehdi Mani and N. Crespi, "What's up: P2p spontaneous social networking," in *Proceedings of PERCOM 2009, IEEE International Conference on Pervasive Computing and Communications, March 2009*, Galveston Tx, USA, March 2009.

[10] E. Adar and B. A. Huberman, "Free riding on Gnutella," *First Monday*, vol. 5, no. 10, Oct. 2000. [Online]. Available: citeseer.ist.psu.edu/article/adar00free.html

[11] J. Shneidman and D. Parkes, "Rationality and self-interest in peer to peer networks," in *Peer-to-Peer Systems II*, ser. Lecture Notes in Computer Science, M. Kaashoek and I. Stoica, Eds. Springer Berlin / Heidelberg, 2003, vol. 2735, pp. 139–148.

[12] T. Locher, P. Moor, S. Schmid, and R. Wattenhofer, "Free riding in BitTorrent is cheap," in *Fifth Workshop on Hot Topics in Networks (HotNets-V)*, Irvine, CA, US, Nov. 2006. [Online]. Available: http://www.sigcomm.org/HotNets-V/program.html

[13] S. Buchegger and J.-Y. L. Boudec, "Nodes bearing grudges: Towards routing security, fairness and robustness in mobile ad hoc networks," in *Proceedings of the 10th Euromicro Workshop on Parallel, Distributed and Network-based Processing*, 2002.

[14] P. Michiardi and R. Molva, "Core: A COllaborative REputation mechanism to enforce node cooperation in Mobile Ad Hoc Networks," in *Proceedings of IFIP Communication and Multimedia Security Conference (CMS)*, 2002.

[15] L. Buttyán and J.-P. Hubaux, "Stimulating cooperation in self-organizing mobile ad hoc networks," *ACM/Kluwer Mobile Networks and Applications (MONET*, vol. 8, pp. 579–592, 2001.

[16] S. Zhong, J. Chen, and Y. Yang, "Sprite: A Simple, Cheat-Proof, Credit-Based System for Mobile Ad-Hoc Networks," in *Proceedings of Infocom*, 2003.

[17] K. Tamilman, V. Pai, and A. Mohr, "Swift: A system with incentives for trading," in *Proceedings of Second Workshop of Economics in Peer-to-Peer Systems*, 2004,.

[18] V. Vishnumurthy, S. Chandrakumar, S. Ch, and E. G. Sirer, "Karma: A secure economic framework for peer-to-peer resource sharing," 2003.

[19] M. Belenkiy, M. Chase, C. C. Erway, J. Jannotti, A. Küpçü, A. Lysyanskaya, and E. Rachlin, "Making p2p accountable without losing privacy," in *Proceedings of the 2007 ACM workshop on Privacy in electronic society*, ser. WPES '07. New York, NY, USA: ACM, 2007, pp. 31–40.

[20] A. Satsiou and L. Tassiulas, "Reputation-based resource allocation in p2p systems of rational users," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 21, no. 4, pp. 466 –479, april 2010.

[21] P. Dewan and P. Dasgupta, "P2p reputation management using distributed identities and decentralized recommendation chains," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 22, no. 7, pp. 1000 –1013, july 2010.

[22] L. Buttyán and J.-P. Hubaux, "Enforcing service availability in mobile ad-hoc wans," in *Proceedings of the 1st ACM international symposium on Mobile ad hoc networking & computing*, ser. MobiHoc '00. Piscataway, NJ, USA: IEEE Press, 2000, pp. 87–96. [Online]. Available: http://dl.acm.org/citation.cfm?id=514151.514164

[23] D. Chaum, "Blind signatures for untraceable payments," in *CRYPTO*, 1982, pp. 199–203.

[24] ——, "Blind signature system," in *CRYPTO*, 1983, p. 153.

[25] J. R. Douceur, "The sybil attack," in *Peer-to-Peer Systems*, 2002, pp. 251 – 260.

[26] A. Singh, T.-W. Ngan, P. Druschel, and D. S. Wallach, "Eclipse attacks on overlay networks: Threats and defenses," in *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, april 2006, pp. 1 –12.

[27] M. Steiner, D. Carra, and E. W. Biersack, "Faster content access in KAD," in *Peer-to-Peer Computing*, Sep 2008.

[28] M. Steiner, T. En Najjary, and E. W. Biersack, "A global view of KAD," in *IMC 2007, ACM SIGCOMM Internet Measurement Conference, October 23-26, 2007, San Diego, USA*, 10 2007.

[29] B. Cohen, "Incentives build robustness in bittorrent?" 2003.

[30] M. Piatek, T. Isdal, T. Anderson, A. Krishnamurthy, and A. Venkataramani, "Do incentives build robustness in bittorrent?" in *In NSDI'07*, 2007.

[31] J. Camenisch, A. Lysyanskaya, and M. Meyerovich, "Endorsed e-cash," in *Security and Privacy, 2007. SP '07. IEEE Symposium on*, may 2007, pp. 101 –115.