Bayesian Analysis of Secure P2P Sharing Protocols

Esther Palomar, Almudena Alcaide,
Juan M. Estevez-Tapiador, and Julio C. Hernandez-Castro

Computer Science Department – Carlos III University of Madrid
Avda. Universidad 30, 28911, Leganes, Madrid
{epalomar, aalcaide, jestevez, jcesar}@inf.uc3m.es

Abstract. Ad hoc and peer-to-peer (P2P) computing paradigms pose a number of security challenges. The deployment of classic security protocols to provide services such as node authentication, content integrity or access control, presents several difficulties, most of them due to the decentralized nature of these environments and the lack of central authorities. Even though some solutions have been already proposed, a usual problem is how to formally reasoning about their security properties. In this work, we show how Game Theory –particularly Bayesian games– can be an useful tool to analyze in a formal manner a P2P security scheme. We illustrate our approach with a secure content distribution protocol, showing how nodes can dynamically adapt their strategies to highly transient communities. In our model, some security aspects rest on the formal proof of the robustness of the distribution protocol, while other properties stem from notions such as rationality, cooperative security, beliefs, or best-response strategies.

1 Introduction

P2P and ad hoc networks are increasingly adopting robust cryptography schemes as more lightweight, trust-based solutions often lead to serious problems such as cheating, collusion and free-riding [1]. In order to solve these problems, there have been several different proposals to address security services in P2P file sharing systems [2,3]. However, the complete absence of fixed infrastructures in such scenarios represents a major difficulty when deploying fully decentralized schemes. Generally, it is not realistic to assume that services such as those provided by a Trusted Third Party (TTP, e.g. authentication and authorization) will be available in these environments. Thus, reputation-based incentive models seemed more promising at controlling the content distribution [4]. In such models, it is reasonable to assume that peers would not misbehave aimed at maximizing the utility (the expected payoff) that they derive from the system in each interaction. It is precisely in this context where notions such as *rationality* become particularly interesting. Informally, a rational peer is considered to be such that it would never behave against its own interest.

In previous work [5], we proposed a protocol to maintain content integrity based on the collaboration among a fraction of peers in the system. Moreover,

R. Meersman and Z. Tari et al. (Eds.): OTM 2007, Part II, LNCS 4804, pp. 1701–1717, 2007.

the model establishes a rational content access control by means of a challenge-response mechanism, whereby nodes may achieve good reputation and privileges. Contrary to classic trust systems where trust decisions are directly or indirectly given by nodes' past behavior, our scheme uses cryptographic proofs of work to discourage selfish behavior and to reward cooperation.

In this paper, we describe a P2P file sharing system in which nodes interact following our proposed scheme. Furthermore, we describe a formal framework to analyze some security aspects of the protocol itself, and also the dynamics created when nodes interact following the protocol description. Additionally, our formal analysis, based on Bayesian Game theory, serves to prove that our scheme offers major advantages over other existing ones. In particular, although we use reputation as an incentive which encourages both content providers and requesters to cooperate, i.e. play by the rules, the dynamics of the community are based on evaluating the requester's trustworthiness and collaboration state by means of several probability distribution functions, which give us different community profiles. This raises an interesting issue as we are able to consider non-collaborative nodes and to measure the effect they might have on the overall system performance.

1.1 Our Contribution

There are three main aspects to our contribution.

- We present an enhanced version of the cryptographic protocol introduced in [5], which provides a secure content distribution and content access control in pure P2P networks, by means of cryptographic puzzles.
- A formal framework is provided in which we analyze the entire scheme. We use such a model to give formal proofs of the protocol's rationality.
- We carry out a formal analysis of the dynamics created on a secure P2P file sharing system, where nodes interact following the steps of our proposal.

The rest of the paper is organized as follows. In Section 2, we provide a brief overview of some related work. Section 3 formally presents our secure P2P content sharing protocol. In Section 4, we first give a brief introduction to Bayesian Game Theory concepts, then present the formal model, and evaluate the dynamics of the new defined system. Finally, Section 5 concludes the paper and outlines some open issues.

2 Related Background

For readability and completeness, we first discuss some of the works on P2P security models and provide a brief introduction to Game Theory applied to P2P systems.

2.1 Cryptographic P2P Security Models

Providing efficient security services in P2P and ad hoc networks is an active researching area which prompts many challenges. Researchers have to adapt common cryptographic techniques, e.g. threshold and public key cryptography, to highly dynamic environments to ensure that even when some nodes are unavailable, others can still perform the task through the coalition of cooperating parties [6]. In particular, some works address the provision of membership control [2,7] by means of distributing the ability to sign and encrypt/decrypt.

In this context, node participation in a network has been recently addressed in [8] by adopting a game theoretical approach. From the results presented, authors conclude that even if nodes perceive a cost in sharing their resources, this may induce node participation.

Discouraging Misbehavior. Further works focus on quantifying the cost/benefit tradeoff that will lead nodes to share theirs resources, and approach secure collaboration-based systems assuming peers' rationality [9]. Particularly, micropayments protocols tend to meet fairness, assuring that providers are guaranteed to be paid, while requesters are discouraged to behave as freeloaders because they are refunded for each upload. Thus, system users are given an incentive to work together towards a common goal [10].

On the other hand, the idea of using cryptographic proof-of-work protocols to increase the cost of sending email and make sending spam unprofitable [11], may be extended to P2P networks, mainly oriented to impede Denial of Service (DoS) attacks, as well as to provide a solution for the free-riding problem [12]. So, research on this topic could lead to encourage fair content distribution using cryptographic puzzles.

2.2 Game Theory Applied to P2P Systems

The use of mathematical tools based on Game Theory to construct a coherent framework in which to design protocols, and also to provide a formal analysis of rules or protocols, has been extensively applied [13]. In fact, Game Theory has recently been used to model nodes' behavior in P2P systems [4,14]. In the latter, Golle et al. analyze free-riding situations using a model based on basic Game Theory. However, they find perturbations -e.g. users joining and leaving the system– when reaching for a Nash Equilibrium. Other Game-theoretical models for trust and reputation systems have also been introduced in [15,16]. In this paper, we will apply a Bayesian Game Theory framework, presented in [17], to analyze our security protocol. The proposed framework is general and can be applied to any protocol in which rationality is sought by design.

3 A Puzzle-Based File Sharing Protocol

This section is structured in three parts. First, we motivate the need for a non-trivial content authentication and access control protocol in highly decentralized

systems. Next we introduce some working assumptions as the main building blocks of our solution. Finally, we present our scheme based on Byzantine agreement for content authentication, and cryptographic puzzles for a secure content download.

3.1 Motivation

In many applications, it is crucial for an user to control who gets access to the contents he/she shares. This task may seem easy to be tackled by assigning permissions, and using a common access control mechanism. However, in a file sharing system contents can be soon replicated through different locations, so ensuring that each new provider will behave accordingly to another user's access policy gets more complicated (and definitely hard should a global security infrastructure be unavailable.)

A common solution consists of renouncing to any form of fine-grained access controls, and simply ask each requester to invest an effort (e.g. computational, such as solving a puzzle) just to prove that he is really interested in the file. Despite its many limitations, such a mechanism can be extremely useful to discourage non-desired behaviors in collaborative environments. For instance, consider a common file sharing scenario wherein, for each transaction, the node P which provides the service (i.e. the content) is called the *provider*, while the node R which requests the content is called *requester*. These are the two protocol parties.

To provide replication, we assume that the system always transmits and presents files encrypted using a key, K_S. This key is established by P, who maintains it in secret. Now, we may use a *trapdoor function* to supply collaborative requesters with l-bits out of the total bits of K_S. These l bits implies the following:

1. Faster key recovery process: The content decryption takes less time and fewer resources.
2. Increased reputation: Each provider maintains a history of transactions; a table, denoted T_i (e.g. for node A, T_A), containing past requesters, the corresponding desired content identifier, and the transaction's result, among other items. So, instances of the protocol imply new entries in the providers' database. Particulary, in the content access stage this information is necessary in order to reach an equilibrium between the collaboration profile of the community and the complexity of the proofs-of-work being issued.

Thus, the proposal motivates honest collaborative actions and serves as an accounting mechanism for the quality of the interactions.

3.2 Working Assumptions

Throughout this work, we will accept the following two working assumptions:

1. Anyone can verify the authenticity of a node's signature by applying a Byzantine fault tolerant public key authentication protocol presented in [6]. In this

work, Pathak and Iftode postulate that a correct authentication depends on an honest majority of a particular subgroup of the peers' community, labeled "trusted group". Thus, honest members from trusted groups are used to provide a functionality similar to that of a CA (certification authority) through a consensus procedure. Finally, successful authentication moves a peer to the trusted group, whereas encountered malicious peers are moved to the untrusted group. So, each peer includes in his/her transactions database, T_i, the identification and security label (clearance) of previously contacted peers.

2. We assume there exists a public external reputation system motivating providers and requesters to behave properly, although as we will see, for the proposed scheme, reputation does not guide the dynamics of the entire protocol.

3.3 Proposed Scheme

The scheme is structured in two main phases: content authentication and content access. We will illustrate both separately.

Notation. Here it is a summary of the notation used throughout the paper:

- N is the number of nodes in the network.
- Each node is denoted by n_i. However, when appropriate, specific nodes will be designated by capital letters: A, B, P, R, etc.
- Each node n_i has a pair of public and private keys, denoted by K_{n_i} and $K_{n_i}^{-1}$, respectively.
- m denotes the content that a specific node wishes to publish.
- A value $h(x)$, represents a cryptographic hash function applied to x.
- Let $enc_{K_{n_i}}(x)$ be the asymmetric encryption of message x using K_{n_i} as key. Similarly, we denote by $enc_{K_S}(x)$ the symmetric encryption of message x using a secret key K_S.
- Let $s_{n_i}(x)$ be n_i's signature over x, i.e.:

$$s_{n_i}(x) = enc_{K_{n_i}^{-1}}(h(x))$$

where $h(x)$ is the result of applying a cryptographic hash function to x. Finally, we denote $s_{n_i}^{n_j}(x)$, n_i's signature on x concatenated with n_j's identity, i.e.:

$$s_{n_i}^{n_j}(x) = enc_{K_{n_i}^{-1}}(n_j||h(x))$$

Content Authentication. Let A be the legitime owner of a given content m. The idea is to ensure content authenticity and integrity, using the signature of k nodes. Briefly, entity A selects k signing trusted nodes denoted by $n_0, n_1, \ldots, n_k$. Once each n_i has agreed to collaborate with A, n_i performs several verifications. This includes computing $h(m)$ and comparing it with the value contained in the received message. In order to avoid that a illegitimate user claims ownership over

1. A generates $h(m)$ and signs it: $s_A(m)$
2. For $i = 1$ to k
   (a) A sends $(m, s_A(m))$ to n_i
   (b) n_i verifies the correctness of $h(m)$ and checks S_{n_i}
   (c) n_i sends its signature: $s^A_{n_i}(m)$ to A
   (d) n_i adds the new entry to her table of signatures S_{n_i}
3. A verifies the correctness of the signatures, encrypts $h(m)$, computes K_S to encrypt m.
4. Finally A publishes:

$$s^A_{n_1}(m)||s^A_{n_2}(m)||\cdots||s^A_{n_k}(m)||enc_{K_S}(m)||enc_{K_A}\big(h(m)\big)$$

together with the identities of the participant nodes:

$$n_1, n_2, \ldots n_k, A$$

Fig. 1. Proposed content authentication scheme

1. $R \rightarrow P$: $m_1 = enc_{K_P}\big(h(m)\big), \sigma_1$
2. $P \rightarrow R$: $m_2 = enc_{K_R}(\varsigma_j), \sigma_2$
3. $R \rightarrow P$: $m_3 = enc_{K_P}(\tau_j), \sigma_3$
4. $P \rightarrow R$: $m_4 = enc_{K_R}(\omega_m), \sigma_4$

$$\begin{aligned} \text{where } \sigma_1 &= s^P_R\big(enc_{K_P}\big(h(m)\big)\big) \\ \sigma_2 &= s^R_P\big(enc_{K_R}(\varsigma_j)\big) \\ \sigma_3 &= s^P_R\big(enc_{K_P}(\tau_j)\big) \\ \text{and } \sigma_4 &= s^R_P\big(enc_{K_R}(\omega_m)\big) \end{aligned}$$

Fig. 2. Proposed content access scheme: Asking for a trapdoor

a content, each signer must keep track of her own signatures over past contents. This information is stored in a local data structure named S_{n_i} (e.g. a table with an entry for each signed content.)

Note that the signer must also check S_{n_i} and verify that no entries exist corresponding to the same content, i.e. she has not signed the same content in the past. If previous verifications succeed, n_i signs m linked to A's identity and sends the signature to A. Then, A symmetrically encrypts m, $enc_{K_S}(m)$ and publishes the content information just generated:

$$(s^A_{n_1}(m)||s^A_{n_2}(m)||\cdots||s^A_{n_k}(m)||enc_{K_S}(m)||enc_{K_A}\big(h(m)\big))$$

together with the identities of the participant nodes $(n_1, n_2, \ldots n_k, A)$.

The content authentication scheme is summarized in Fig. 1.

Content Access. Now, let a node R be a requester who requires m from a provider P. We can assume that, at this time, R is already engaged in a searching process, which typically leads to a list of sources that keep a replica of the desired content. A query result should at least return the content descriptor and the list of identities of

the source nodes. Together with each query result, R obtains providers' published information, as explained in Fig. 1–step 4. Before verifying the signatures, R must select a source according to some criterion, e.g. trust on some of the signers. Thus, R must choose between the following two options:

- R may try to get m from the tokens received in the search. For this purpose, she can only mount a brute force attack, which complexity will depend on the length of K_S.
- R may ask P for a trapdoor to access content m. For this, R must initiate a four-step protocol (see Fig. 2) in which P must also participate. Briefly, before R can reach the trapdoor (l–bits out of the secret key K_S), she has to solve a challenge issued by P. The challenge represents the proof-of-work requested for granting permission to easily access the content, and its complexity depends on content security level and the conjecture over the community's collaboration nature. In particular, conjectures are led by local evaluations of the corresponding transactions table, T_P. We further elaborate on the protocol steps in the best possible case, i.e. in which both main parties are motivated to behave correctly and to follow the protocol faithfully.

Asking for a Trapdoor. The requester R contacts the provider P using the last part of the content's information published, $enc_{K_P}\big(h(m)\big)$, and signs it (Fig. 2–message m_1). With this message m_1, P can check R's identity (implicit in the notation used) and the required content's hash. After this, P must decide whether R "deserves" the content or not. For this, P challenges R by elaborating a conjecture θ according to:

1. An estimation of R's collaborative attitude (i.e. if P believes that R is collaborative or non-collaborative.) A collaborative requester is one that does not deviate from the protocol specification.
2. The results of past interactions: past experience will allow P to dynamically define new estimates.

The underlying idea is that P will try to reduce the cost when she estimates it is highly possible to interact with non-collaborative peers of the community.

This conjecture can take the form of a numerical value, so if the computed value is higher than a given threshold, R is supposed to be collaborative and, therefore, P decides to continue with the protocol. Note that the decision of interacting with R depends on time-varying factors, such as the accumulated experience of P within this community. As a consequence, it seems reasonable to update these beliefs regularly.

If R is estimated as collaborative, P computes a challenge ς and sends it to her (message m_2). Upon receiving it, R sends back to P the corresponding response τ (message m_3). If P considers τ as correct, she sends to R the trapdoor $\omega(K_S)$ necessary to recover the key K_S (message m_4). In any case, both P and R store the result of their interaction in their tables T_P and T_R, respectively. This table has the following structure:

$$\langle n_j, h(m), \varsigma, \tau, t_i, F \rangle$$

where t_i is a timestamp and F a flag indicating if the transaction has been successful or not.

Concerning the challenge itself, there exist a number of primitives which can be used for this purpose (e.g. moderately-hard memory-bound functions [18]). The basic idea is that the verification by the challenger should be fast, but the computation by the requester has to be fairly slow.

Finally, after a huge part of the key is received, R must try, on average, 2^{n-l-1} keys –n being the number of bits of K_S–, to decipher m. Note that P may supply many different trapdoors for the same m, choosing any l bits of K_S randomly.

3.4 On the Scheme's Performance

We have informally evaluated the efficiency provided by our proposal in several stages considering the number of cryptographic operations and the complexity for each stage of the protocol [19]. Particulary, content authentication stage performs a number of hash generations, signature generations and verifications which depends on the number k of signers, plus a pair of symmetric encryptions. For example, the verification cost is $\mathcal{O}(|m|k^3 \log k)$, since $\frac{k(k+1)}{2}$ verifications are performed by k signers, and also depends on the content's length, $|m|$. Of course, this also implies that k instances of Pathak and Iftode's protocol must be executed. On the other hand, the number of transmitted messages in the content authentication process increases as the number of signers involved in each stage grows. Hence the number of transmitted messages is $\mathcal{O}(k \log k)$.

In turn, the cost of the content access stage depends on the requesters' collaborative nature. However, it is interesting to examine to what extent the usage of this kind of effort-aware access control can be applicable and reliable in real networks. For example, puzzle generation takes less than 2 minutes using AES standard with 128-bits key, but a brute force attack over the same encryption would cost more than 30 minutes. On the other hand, the cost of getting the key decreases in case of having a trapdoor, e.g. with a 32-bits trapdoor takes less than 10 minutes. First, we have considered that more than 32 hidden bits would be impracticable, e.g. with 2^{64} large number of potential keys to test, it literally takes a matter of years. Additionally, time also relies on peers' computational resources, i.e. computing power and memory size and speed. Furthermore, we have measured the computational cost for content verification in the *worst* case, i.e. no signers are known and all the verifications must be performed, as an upper bound. As content's size and the number of signers increase, the computational cost grows significatively: A content of 500MB and 20 signers takes approximately 1 hour. Note, however, that this task is carried out just once, and that content access is considerably faster.

4 Protocol Formal Analysis

Our analysis of the scheme introduced in the previous section is based on Alcaide et al's work in [17]). In particular, the protocol will be represented by a Bayesian

game. Furthermore, several Game Theory results will allow us to predict the outcome of a such a game, and therefore the outcome of the protocol execution it represents. Despite the analysis becoming more complex than by using basic Game Theory, we find it more realistic and more informative, which will enable us to make important statements about the dynamics of our system. In the rest of the paper, we will formally introduce our analysis.

4.1 Bayesian Framework

The actual model is based on describing the given protocol as a *game of Incomplete Information*, also called *Bayesian game*, and compute the moves which bring the game to an equilibrium, from where players would not want to deviate. The corresponding *protocol game*, derived from the protocol description, is intended to model all possible interactions of the protocol participants, even the potentially misbehaving actions (i.e., those different from the ones described by the protocol). Each of the parties involved in the protocol becomes a player of the protocol game, including the network.

In a Bayesian game, each player is allowed to have some private information that affects the overall game play but which is not known by others. This information is usually related to their payoff values (what players receive at the end of the game, depending on what strategies all players play). Players have initial beliefs about the type of their opponents and can update their beliefs on the basis of the actions they have played. We will briefly introduce the two most relevant concepts in Bayesian games: player's type and player's beliefs. Formal definitions of these concepts will be introduced when defining our particular system.

- *Player's type.* The type of a player determines univocally that player's payoff function, so that different types will be associated with different payoff functions. A Bayesian game is modeled by introducing Nature as a player in a game. Nature randomly chooses a type for each player according to the probability distribution across each player's type space.
- *Player's beliefs.* Each player's set of beliefs will assist them in the process of choosing the *best-response* strategies to confront their opponents.

We will only reproduce those aspects of the formal model described in [17], which are essential to the goals and scope of this paper using the same notation whenever possible. Please refer to [17] and [20] for further refinements and details. What we are presenting next is a brief description of the main concepts to support our rationality proof and to study the dynamics created when nodes interact in a file sharing P2P system, using the scheme proposed above. The following definitions formalize the aforementioned concepts for our specific system.

4.2 Players and Types

As mentioned before, each protocol participant becomes a player in the corresponding protocol game. Let provider P and requester R be the players of

the protocol game, denoted as G_{RP}, and created from the protocol description given in Section 3. We consider $\{P, R, Net\}$ as the complete set of players. However, since the network always plays following a fixed strategy (the network is considered to be reliable), we only develop specifications for players P and R.

Definition 4.1 (Players type profiles). *Let $\mathcal{T} = \mathcal{T}_P \times \mathcal{T}_R$ be the type-profile space, where $\mathcal{T}_p = \{C\}$ and $\mathcal{T}_R = \{C, NC\}$ are the type spaces for players P and R, respectively. A type C denotes a cooperative node, while NC denotes a non-cooperative one.*

In other words, in our particular instance, a provider node P has an only type, *cooperative*. By contrast, requester nodes R have two different types. We will denote by P a cooperative provider, and by R_C and R_{NC} a cooperative and a non-cooperative requester, respectively.

We consider the following probability distribution over the space $\mathcal{T}$:

$$\theta_C = prob(R_C|P) \quad \theta_{NC} = prob(R_{NC}|P)$$
$$\text{s.t. } \theta_C + \theta_{NC} = 1$$

Note that $prob(P|R_C) = prob(P|R_{NC}) = 1$.

4.3 Strategies and Beliefs

Informally, a *pure strategy* for a player E in a game G, is a complete contingency plan which describes the series of *actions* that player E would take at each possible decision point in the game G. For our specific instance we define:

Definition 4.2 (Players set of actions). *Let $A_P = \{m_2, m_4, quit_P\}$ and $A_R = \{m_1, m_3, quit_R\}$ be the sets of actions for players P and R, respectively.*

Definition 4.3 (Pure strategies for player P). *In G_{RP}, the complete set of* pure *strategies for player P, denoted as S_P, is defined as $S_P = \{s_1^P, s_2^P, s_3^P\}$, where: $s_1^P = (m_2, quit_P)$, $s_2^P = (m_2, m_4)$ and $s_3^P = (quit_P, \cdot)$.*

The first component of each tuple represents the action taken by player P at round 2 of the protocol (P's first turn to move). In a similar way, the second component represents the action taken by P at round 4 of the protocol (player P's second chance to make a move).

Definition 4.4 (Pure strategies for player R). *In G_{RP}, a pure strategy for player R is represented by a tuple $s^R = (s^{R_C}, s^{R_{NC}}) \in S^R \times S^R$, where s^{R_C} represents the strategy to follow by a node R of type* collaborative*, whereas $s^{R_{NC}}$ represents the strategy to follow by a node R of type* non-collaborative.

The set S^R is defined as: $S^R = \{s_1^R, s_2^R\}$ with $s_1^R = (m_1, quit_R)$ and $s_2^R = (m_1, m_3)$. The complete set of pure strategies for player R is then described as: $\{(s_1^R, s_1^R), (s_2^R, s_2^R), (s_1^R, s_2^R), (s_2^R, s_1^R)\}$.

In this case, the first component of each tuple represents the action player R takes at stage 1 in the protocol game, and the second describes the action at stage 3 (first and second turns for R to move).

Definition 4.5 (Strategy profile). *A strategy profile in the G_{RP} game is a vector $s = (s^R, s^P)$ of individual strategies, one for each player, where $s^R \in S^R \times S^R$ and $s^P \in S^P$.*

Note that, **specifying a strategy profile univocally determines the outcome of the game.**

The following probability distributions represent the set of beliefs each entity holds over the opponent's set of actions at each particular stage of the protocol.

At stage 2 of the protocol, P's conjecture over node R's real nature (requester R could be collaborative or non-collaborative) is represented by the following probability distribution function over $\mathcal{T}_R$:

$$\theta = prob(R_{NC}|m_1) \qquad 1 - \theta = prob(R_C|m_1)$$

Requester nodes are able to form conjectures over the provider's intention to quit the protocol at round two of the protocol. We define the following probability distribution function to represent such belief:

$$\alpha = prob(quit_P|P) \qquad 1 - \alpha = prob(m_2|P)$$

Note that, at stages three and four of the protocol, entities are *rationally* forced to follow the protocol description as they obtain a better payoff value be doing so. Hence, there is no need for opponent's nodes to form conjectures over any other kind of behavior at those steps. Rationality is therefore forcing nodes to take specific actions. We will give formal proof of this statement in further sections.

4.4 Payoff Functions

As stated before, one of the key points of Bayesian games is the fact that each type of player is associated with a different payoff function. We define the following payoff functions:

$$U_R, U_P : \mathcal{T}_R \times S_R \times S_P \to \mathbb{R}$$

Fig. 3 relates all possible payoff values obtained by players in the G_{RP} game.

In addition, we impose the following constraints:

$$r^- > r^t$$
$$p^+ > p^- > 0$$
$$0 \leqslant \omega, \theta, \alpha \leqslant 1$$

Moreover, a detailed representation in extensive form of the protocol game G_{RP} is provided in Fig. 4. Briefly, the common interpretation of an extensive form game is the following. The game can be thought of as a tree, where the edges and the vertices are associated to actions and sequences of actions, respectively. Terminal vertices are those that cannot be followed by any other actions. When a sequence of actions reaches a terminal vertex, the game ends. For each branch in the tree, the payoff value associated to its final node represents the total outcome that players R and P obtain when following such a path.

p^-	Cost for node P to elaborate a puzzle to include in m_2.
p^+	Profit for node P when completing the protocol sending the trapdoor included in m_4. This value represents the potential new business generated by well behaved providers ensuring the continuation of the current system.
r^t	Value it has for requester R to have received a puzzle. When R is non-collaborative, this value represents the reward to having misled provider P to enter the protocol, when R had no intention to send back a response.
r^+	When the requester is collaborative, this value represents the gain after receiving the trapdoor included in m_4.
r^-	Cost for player R to elaborate an answer to the P's challenge.

Fig. 3. Payoffs in the G_{RP} game

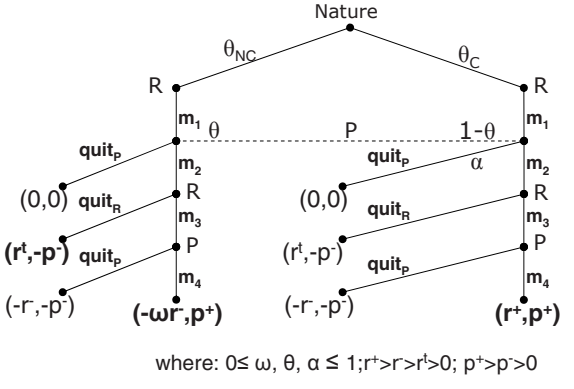

Fig. 4. G_{RP} in extensive form

4.5 Dominated Strategies and Expected Gains

In this section, we will compute the gains each player expects to obtain when following a specific strategy.

There are cases when it is possible to anticipate the moves that rational players will or will not take during the protocol game execution. All those actions for which the expected payoff is lower than the one obtained following other options are called *dominated strategies*. Dominated strategies can be eliminated from the formal analysis as self-interested rational players will never follow them. By contrast, a *dominant strategy* is such that, a rational player will always choose to follow it, as the expected gain by doing so is greater than by taking a different move. In our specific analysis of the G_{RP} protocol game, we can clearly identify one dominated strategy for player P at the final stage of the protocol game:

– *The move m_4 dominates the last round of the protocol game.* Every rational provider P, having reached stage 4 in the protocol game, will always choose to send message m_4, as by doing so the expected payoff is greater than by quitting the game. Both a reputation system and the prospect of future profitable interactions are represented by the positive value p^+.

	$s^R_1 = (m_1, quit_R)$	$s^R_2 = (m_1, m_3)$
R_C	*dominated* $EG(R_C, s^R_1, \alpha) < EG(R_C, s^R_2, \alpha)$	*dominant* $EG(R_C, s^R_1, \alpha) < EG(R_C, s^R_2, \alpha)$
R_{NC}	*dominant* $EG(R_{NC}, s^R_1, \alpha) > 0$	*dominated* $EG(R_{NC}, s^R_2, \alpha) < 0$

Fig. 5. Dominant and dominated strategies for player R

	$s^P_1 = (m_2, quit_P)$	$s^P_2 = (m_2, m_4)$	$s^P_3 = (quit_P, \bullet)$
$\theta < p^+/(p^+ + p^-)$	*dominated* $EG(P, s, \theta) < 0$	*dominant* $EG(P, s, \theta) > 0$	*dominated* $EG(P, s, \theta) = 0$
$\theta = p^+/(p^+ + p^-)$	*dominated* $EG(P, s, \theta) < 0$	$EG(P, s, \theta) = 0$	$EG(P, s, \theta) = 0$
$\theta > p^+/(p^+ + p^-)$	*dominated* $EG(P, s, \theta) < 0$	*dominated* $EG(P, s, \theta) < 0$	*dominant* $EG(P, s, \theta) > 0$

Fig. 6. Dominant and dominated strategies for player P

We can compute the *Expected payoff* values (EG),for each of the players involved and the remaining set of moves, by multiplying the probability of following a specific branch of the tree and the payoff expected at the final node.

From player R's point of view, we have the following results, summarized in Fig. 5:

$$\begin{aligned} EG(R_C, s^R_1, \alpha) &= (1-\alpha) \cdot (r^t) \\ EG(R_C, s^R_2, \alpha) &= (1-\alpha) \cdot r^+ \\ EG(R_{NC}, s^R_1, \alpha) &= (1-\alpha) \cdot r^t \\ EG(R_{NC}, s^R_2, \alpha) &= (1-\alpha) \cdot (-\omega r^-) \end{aligned} \tag{1}$$

Equations (1) let us formally reason and establish the following statements:

- Action $quit_R$ is dominated by action *send* m_3 at stage 3 of the protocol game, when R type is collaborative. At this stage in the protocol game, R is sure of P's latest move (participant rationality is public information) so choosing to send m_3 offers R a greater payoff value. Note $EG(R_C, s^R_1, \alpha) < EG(R_C, s^R_2, \alpha)$. Strategy s^R_2 is therefore a dominant strategy for R_C.
- By contrast, action $quit_R$ dominates strategy *send* m_3 at stage 3 of the protocol game and for player R, type non-collaborative. Choosing $quit_R$ offers R_{NC} a positive payoff value of $(1-\alpha) * r^t \ \forall \ 0 < \alpha < 1$, whereas choosing m_3 will only get a negative payoff value. Strategy s^R_1 is therefore a dominant strategy for player R_{NC}.

Similar calculations can be carried out from player P's point of view. For this we have:

$$\begin{aligned} EG(P, s_1^P, \theta) &= -p^- \\ EG(P, s_2^P, \theta) &= \theta \cdot (-p^-) + (1-\theta) \cdot p^+ = \\ &\quad p^+ - \theta \cdot (p^+ + p^-) \\ EG(P, s_3^P, \theta) &= 0 \end{aligned} \tag{2}$$

Equations (2) let us formally reasoning and establishing the following statements:

- Strategy s_1^P is clearly a dominated strategy for player P, as the expected payoff is negative $\forall\ 0 \leqslant \theta \leqslant 1$.
- Strategy S_2^P is a dominant strategy over S_3^P if and only if $EG(P, s_2^P, \theta) > 0 \Leftrightarrow \theta < p^+/(p^+ + p^-)$.

Fig. 6 summarizes the aforementioned results.

4.6 Evaluation

As described above, the formal model has served to formally prove that our scheme is rational: rational (self-interested) entities will always follow the steps described by our protocol.

Equilibrium. An equilibrium in the system will be represented by an equilibrium in the game. An equilibrium in the system will be a certain state which self-interested parties will not want to unilaterally move from. An equilibrium in the game is a set of strategies from which players would not want to individually deviate to obtain better payoff values.

We will consider a best-response function for each player and type. The best-response function offers players the best strategies when responding to all possible types of an opponent and all their possible strategies.

Fig. 7 (left) depicts graphically the best-response function for player P, according to the expected payoff values calculated in equations (2). Deriving data from equations (1), Fig. 7 (right) does the same for player R. The left vertical line correspond to type C, while the right vertical line correspond to type NC. The figure shows the intersection with player P's best response function. It is precisely in these intersection points, where both best-response functions cross each other, that the equilibrium is reached. Neither the provider nor the requester would want to modify their strategies unilaterally, as by doing so they would not obtain better results. Note that one of the equilibrium points is reached when all players complete all steps in the protocol. This serves to formally prove that our scheme is a rational one ([20]).

Impact of Non-collaboration. Additionally, the formal model allows us to formally identify and measure two main factors of the proposed model.

- Firstly, the system dynamics depends upon the conjecture θ that player P makes on the type of community in which it is immersed. This conjecture could vary and it can be dynamically adjusted while the system is operative.

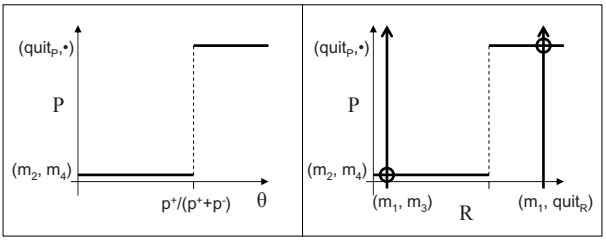

Fig. 7. (Left) Best-response function for P. (Right) Intersection of best-response functions for P and R (both R_C and R_{NC}).

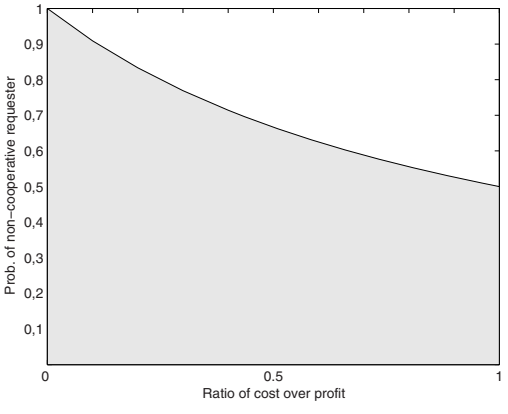

Fig. 8. Relationship between θ, p^+ and p^-

– Secondly, the ratio p^-/p^+ (cost over profit) does also influence the system behavior. Both parameters are used as control parameters for the dynamics of the system. Fig. 8 shows the relationship between the ratio cost over profit (p^-/p^+) and the threshold computed by P ($\theta < p^+/(p^+ + p^-)$) to accept the request and enter the protocol. Note that P's conjecture over the proportion of non-collaborative nodes within the community $\theta =$ Prob("R being no-cooperative") must always be lower than 0.5. P uses these calculations as a defense mechanism against communities where the number of non-cooperative nodes is greater that the number of cooperative ones.

5 Conclusion and Open Issues

Infrastructure-less networks, on which, in general, one cannot assume the existence of centralized services such as those provided by TTPs, present a challenge in terms of formalizing collaboration-based security protocols. In this paper we have analyzed the protocol game of a rational content sharing scheme modeling all possible interactions of the protocol participants. In our opinion, the system

we have outlined in this proposal offers major advantages over other existing ones. Firstly, there are no restrictions over the community new joiners, as the dynamics are not only based on reputation but also on providers local experience. Secondly, although we are assuming rational behavior, we are able to consider non-collaborative players and to measure the effect they might have on the overall system performance. Finally, we have observed that only in controlled and homogeneous community profiles (all non-cooperative, all cooperative), the system reaches an equilibrium. In future works, we will tackle two main aspects of our proposal. First, we will further elaborate on the extensions of providers' payoffs measurement and secondly, a major goal for us will be to evaluate the effects on overall system performance.

References

1. Zhu, B., Jajodia, S., Kankanhalli, M.: Building trust in peer-to-peer systems: a review. International Journal of Security and Networks 1, 103–112 (2006)
2. Narasimha, M., Tsudik, G., Yi, J.: On the utility of distributed cryptography in p2p and manets: The case of membership control. In: Proceedings of the 11th IEEE International Conference on Network Protocols, Atlanta, USA, pp. 336–345. IEEE Computer Society, Los Alamitos (2003)
3. Zhang, X., Chen, S., Sandhu, R.: Enhancing data authenticity and integrity in p2p systems. IEEE Internet Computing, 42–49 (2005)
4. Buragohain, C., Agrawal, D., Suri, S.: A game theoretic framework for incentives in p2p systems. In: Proceedings of the 3rd Int. Conf. on Peer-to-Peer Computing, Linköping, Sweden, pp. 48–56. IEEE Computer Society, Los Alamitos (2003)
5. Palomar, E., Estevez-Tapiador, J., Hernandez-Castro, J., Ribagorda, A.: A protocol for secure content distribution in pure p2p networks. In: Proceedings of the 3th Int Workshop on P2P Data Management, Security and Trust, Krakow, Poland, pp. 712–716. IEEE, Los Alamitos (2006)
6. Pathak, V., Iftode, L.: Byzantine fault tolerant public key authentication in peer-to-peer systems. Computer Networks (2006)
7. Saxena, N., Tsudik, G., Yi, J.: Admission control in peer-to-peer: Design and performance evaluation. In: Proceedings of the 1st ACM Workshop Security of Ad Hoc and Sensor Networks, Virginia, USA, pp. 104–114 (2003)
8. DaSilva, L., Srivastava, V.: Node participation in ad hoc and peer-to-peer networks: A game-theoretic formulation. In: Proceedings of the Wireless and Comm. and Networking Conf., New Orleans, USA, IEEE Computer Society, Los Alamitos (2005)
9. Shneidman, J., Parkes, D.: Rationality and self-interest in peer to peer networks. In: Proceedings of the IPTPS, pp. 139–148. Springer, Heidelberg (2003)
10. Zhang, Y., Lin, L., Huai, J.: Balancing trust and incentive in peer-to-peer collaborative system. Int. Journal of Network Security 5, 73–81 (2007)
11. Dwork, C., Naor, M.: Pricing via processing or combatting junk mail. In: Brickell, E.F. (ed.) CRYPTO 1992. LNCS, vol. 740, pp. 139–147. Springer, Heidelberg (1993)
12. Juels, A., Brainard, J.: Client puzzles: A cryptographic defense against connection depletion attacks. In: Proceedings of the Networks and Distributed Security Systems, California, USA, pp. 151–165 (1999)

13. Syverson, P.: Weakly secret bit commitment: Applications to lotteries and fair exchange. In: Proceedings of the 11th IEEE Computer Security Foundations Workshop, pp. 2–13 (1998)
14. Golle, P., Leyton-Brown, K., Mironov, I.: Incentives for sharing in peer-to-peer networks. In: Proceedings of the Conference on Electronic Commerce, pp. 14–17. ACM Press, Tampa, USA (2001)
15. Gupta, R., Somani, A.: Game theory as a tool to strategize as well as predict nodes behavior in peer-to-peer networks. In: Proceedings of the 11th Int. Conf. on Parallel and Distributed Systems, Fukuoka, Japan, pp. 244–249. IEEE Computer Society, Los Alamitos (2005)
16. Nurmi, P.: A bayesian framework for online reputation systems. In: Proceedings of the Advanced Int. Conf. on Telecomm, Guadeloupe, French Caribbean, IEEE Computer Society, Los Alamitos (2006)
17. Alcaide, A., Estevez-Tapiador, J., Castro, J.H., Ribagorda, A.: Bayesian rational exchange (to appear in Int. Journal of Information Security)
18. Abadi, M., Burrows, M., Manasse, M., Wobber, T.: Moderately hard, memory-bound functions 5, 299–327 (2005)
19. Palomar, E., Estevez-Tapiador, J., Hernandez-Castro, J., Ribagorda, A.: Certificate-based access control in pure p2p networks. In: Proceedings of the 6th International Conference on Peer-to-Peer Computing, Cambridge, UK, pp. 177–184. IEEE, Los Alamitos (2006)
20. Buttyán, L.: Building Blocks for Secure Services: Authenticated Key Transport and Rational Exchange Protocols (PhD thesis)