An Extended Model of Rational Exchange Based on Dynamic Games of Imperfect Information

Almudena Alcaide, Juan M. Estevez-Tapiador,
Julio C. Hernandez-Castro, and Arturo Ribagorda

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

Abstract. The notion of rational exchange introduced by Syverson in 1998 is a particularly interesting alternative when an efficient scheme for fair exchange is required but the use of a trusted third party is not allowed. A rational exchange protocol cannot provide fairness, but it ensures that rational (i.e. self-interested) parties would have no reason to deviate from the protocol. Buttyán et al (2003) have recently pointed out how rationality in exchange protocols can be formalized and studied within the framework provided by Game Theory. In this paper, we identify some vulnerabilities in Syverson's protocol which were not detected by Buttyán et al's analysis. These motivate us to extend the model to consider new aspects, never formalized before when analyzing security protocols. These aspects are related to participants' reputation, protocol's robustness, and the impact that scenarios where the protocol is executed repeatedly have on the outcome of the protocol execution.

1 Introduction

It is not only the design and definition of security protocols that has been the focus of researchers in recent years. The definition of formal models to validate and verify such protocols has also been an area of intense research and development. Since the definition of the Dolev-Yao adversary model [4], many tools and techniques have been developed to prove the correctness of a protocol. Informally, a protocol is assumed to be correct when it satisfies all its goals, requirements, and properties. However, correctness does not mean that the protocol offers protection against every type of attack. Most of those validation methods and tools have been very successful at finding security flaws and providing counterexamples in many protocols. At the same time, each one of those tools has also got very significant limitations based on one single factor: the way to model unpredictable behavior; that is, how to predict the way in which a set of events, outside the protocol specifications, can subvert a protocol execution, and what the outcome would be. To attack a security protocol we only need to step out of the set of restrictions imposed by the model used to verify its properties [5]. On the other hand, without limiting the actions of each of the entities involved in a protocol the resulting model would be too wide, too difficult –if not impossible

G. Müller (Ed.): ETRICS 2006, LNCS 3995, pp. 396–408, 2006.

to manage– and, ultimately, an *undecidable* problem. There are also other types of limitations regarding existing tools and models to validate protocols, such as the constantly new pointed out properties of security services and definitions of new cryptographic primitives. Models and validation tools have to be enhanced or modified to verify compliance with the new service properties. Consider, for instance, *timeliness* and *composability.* The first one states that a security property has to be achieved in a finite amount of time, either by successful completion of the protocol or by forcing protocol termination. On the other hand, composability establishes that, given a collection of protocols executed simultaneously, it is necessary to demonstrate that no protocol in the collection will accept a message sent by another protocol in the same collection. Usually new properties such as the previous arise from new attacks and threats on breaking security services, and it is not always easy to formally define and validate them. In this context, the notion of *rational exchange* constitutes a relatively recent proposal that still poses new challenges despite its similarities with fair exchange. Before presenting our contribution, we further elaborate on this topic in the following.

1.1 Fairness and Rationality in Exchange Protocols

The problem of how to design a general procedure according to which two parties can exchange items in a *fair* manner has attracted much attention lately. Interest in this class of protocols stems from its importance in many applications where disputes among parties can occur, such as digital contract signing, certified e-mail, exchange of digital goods and payments, etc. In particular, assurance of fairness is fundamental when the exchanged items include any kind of evidences of non-repudiation, for this constitutes a key service in most of the previously mentioned applications. As a result, fair non-repudiation has experienced an explosion of proposals in recent years (see [10]) for an excellent survey).

Roughly, the property of fairness means that no party should reach the end of the protocol in a disadvantageous position, e.g. having sent her item without having received anything valuable in return. Formally, there exists no protocol according to which a number of parties can exchange items in a fair manner exclusively by themselves, and assuming that misbehaving parties can take part in the protocol. Pagnia and Gärtner provide a formal treatment of this problem in [14].As a result, the simplest protocol than can provide true fairness requires a trusted third party (TTP) in order to preserve the property during the exchange.

Recent computing paradigms, such as ad hoc and peer-to-peer networks, pose a challenge from the point of view of the security mechanisms that should be applied. In many cases, the operation of these systems is based on a complete absence of fixed infrastructures. Generally, it is not realistic to assume that services such as those provided by a TTP will be available in those environments. It is precisely in this context where notions such as *rationality* become particularly interesting. This concept, widely known to game theorists, was applied to security protocols by Syverson in 1998 [15]. Informally, a rational exchange protocol cannot provide fairness, but it ensures that rational (i.e. self-interested) parties would have no reason to deviate from the protocol as misbehaving does not result in any benefit. Since rational exchange protocols provide fewer guarantees, one would expect that they also demand fewer requirements, so they can be viewed as a trade-off between complexity and true fairness. In particular, rational exchange protocols do have the enormous advantage of not needing a trusted third party.

1.2 Overview

Syverson's rational exchange protocol was formally analyzed by Buttyán et al in [3], using their definition of rationality within a game-theoretical model proposed in the same work. However, we will see that the protocol presents significant weaknesses which were not detected by previous analysis, so a series of attacks can be successfully carried out. Our intention has been to highlight the protocol's vulnerabilities and also to extend the model in order to capture other relevant aspects involved in the protocol execution. Our proposal relies on modeling the protocol as a game of imperfect information, in which protocol participants have to establish levels of confidence based on bayesian considerations.

The rest of the paper is organized as follows. Section 2 presents the Syverson's protocol, its vulnerabilities and possible attacks, and the appropriate modifications needed to prevent them. In Section 3, we give a brief description of Buttyán et al's model and describe some enhancements. Finally, in Sections 5 and 6 we present our extended model and describe the main conclusions reached.

2 Analysis of Syverson's Rational Exchange Protocol

The scheme is illustrated in Fig. 1. A and B denote the two protocol parties, with private keys k_A^{-1} and k_B^{-1}, respectively. We assume that $item_A$ and $item_B$ are the items they would like to exchange, being $desc_{item_A}$ a description of $item_A$. (There is no equivalent description for $item_B$ because the scheme was introduced to serve as a payment protocol, in such a way that $item_B$ has the role of the payment for buying $item_A$). Moreover, $enc(k, m)$ is a symmetric encryption algorithm that encrypts message m with key k. Likewise, $sig(k_i^{-1}, m)$ provides a digital signature on m by using secret key k_i^{-1}. Finally, $w(\cdot)$ is a WSBC (Weakly Secret Bit Commitment) function [15]. For our analysis, it suffices to know that $w(x)$ keeps x secret, but it can be broken in acceptable bounds on time.

In step one, A sends B her item $item_A$ in a weakly encrypted form. Next, B sends A her item $item_B$ in return, along with acknowledgment of the first message. Finally, A sends the appropriate key k and acknowledgment of the second message.

There are two potentially critical situations which can take place: A might fail to send message m_3 or it might not send it for a long time and, as B can only disclose the encrypted $item_A$ when the payment has already taken place, A could send a forged $item_A$ and still receive payment in return. The first deterrent against A delaying sending message m_3 is that A gains nothing by doing so, except a bad reputation that could ruin its business. In the case of A sending B

$$\begin{aligned}
&A \to B : m_1 = (desc_{item_A}, enc(k, item_A), w(k), \sigma_1)\\
&B \to A : m_2 = (item_B, m_1, \sigma_2)\\
&A \to B : m_3 = (k, m_2, \sigma_3)
\end{aligned}$$

where:

$$\begin{aligned}
\sigma_1 &= sig(k_A^{-1}, (desc_{item_A}, enc(k, item_A), w(k)))\\
\sigma_2 &= sig(k_B^{-1}, (item_B, m_1))\\
\sigma_3 &= sig(k_A^{-1}, (k, m_2))
\end{aligned}$$

Fig. 1. Syverson's rational exchange protocol

the wrong $item_A$, B holds message m_3 as a proof of such misbehavior. However, an important issue arises from both of the previous statements: both participants must exchange during the protocol execution irrevocable evidences to prove the other participant's misbehavior. For example, a scheme on entity A's reputation can only be implemented when it is not possible for B to accuse A of misbehaving if A was honest, and vice versa. A fourth message could be added in which customer B acknowledges timely receipt for message m_3. Likewise, for B to be able to prove in front of an external judging entity that A sent an invalid $item_A$, B must hold irrevocable proof of such a message. The context in which this protocol might be executed has to be carefully checked, as Syverson's protocol is not always appropriate. The author identifies scenarios where the scheme could be used for: (1) If the vendor A is selling relatively low value items, so it is not worth for the customer (in terms of computational cost or the inconvenience of delay) to break the encryption to recover the item; (2) the vendor A might be selling something that might be of timely and diminishing value, such as short term investment advice or regularly changing lists of bargain items for sale; or (3) the protocol might begin one step earlier with a signed customer request for $item_A$. The vendor A can then take the chance of trading with unknown customers and refuse to service customers who repeatedly fail to pay.

2.1 Vulnerabilities

Syverson protocol, as defined by its author, presents some vulnerabilities that allow a series of attacks to be successfully carried out.

Attack I. Consider the following scenario, where $P(Q)$ means that party P acts impersonating the role of party Q:

$$\begin{aligned}
A \to B \quad &: m_1 = (desc_{item_A}, enc(k, item_A), w(k), \sigma_1)\\
B(A) \to C \quad &: m_1 = (desc_{item_A}, enc(k, item_A), w(k), \sigma_1)\\
C \to B(A) &: m_2 = (item_C, m_1, \sigma_2)
\end{aligned}$$

This attack is based on B impersonating A, sending the same message m_1 to C and receiving $item_C$ in return. B would have to quit the protocol after receiving the payment, as she has no key to send to C. Although C has paid a full price for

$item_A$, by the time that k is disclosed to C, $item_A$ would be of very little value to C. The customer C could only present message m_1 to prove A misbehaved. However, A will claim that m_1 was never intended for C and that she was not part of such a communication. Indeed, there is nothing in m_1 linking A and C as participants on the same protocol run. To overcome this attack, some amendments should be made to the structure of m_1.

Attack II. Let us suppose the following simplistic scenario: A is selling an access code to enable the viewing of a football match on a private television network. Let us suppose that A and B carried out a successful Syverson's protocol execution and that they properly exchanged the encrypted access code, $item_B$ and the corresponding key k in messages m_{11}, m_{12}, and m_{13}, respectively. The access code that B has bought from A is obviously of timely diminishing value, but B could still have time to impersonate A and sale the access code to other customers, receiving payment in return:

$$\begin{aligned} B(A) \rightarrow C \quad &: m_{21} = m_{11} = (desc_{item_A}, enc(k, item_A), w(k), \sigma_A) \\ C \rightarrow B(A) &: m_{22} = (payment_C, m_{21}, \sigma_C) \\ B(A) \rightarrow C \quad &: m_{23} = m_{13} = (k, m_{12}, \sigma_A) \end{aligned}$$

In this scenario, by the time C receives message three and realizes that there is a fraud going on, C has no evidence of such a fraud to present in front of a judge, and has got the key k to decrypt the football match access code and watch the match. However, A could claim that C is watching a program without a license and take action against her. If the number of reselling codes is large, the scale of the fraud would make it impractical to pursue each one of the individuals watching the match with no license. Furthermore, trying to trace back the origin of such messages would be practically impossible. Again, to address this problem the content of message one should be amended.

Attack III. If a vendor sends the customer a message m_1 containing garbage (i.e, a ciphertext which does not correspond with the actual $item_A$), the vendor is indeed providing the customer with evidence of such a form of cheating. Message m_1 could be presented to a judge and the vendor would be charged with the appropriate penalty. Such a penalty could greatly exceed the value of the goods, so the vendor is completely discouraged from performing such a scheme. However, the vendor could not be sued and penalized twice for the same offense and, on these terms, a vendor A could carry on sending the forged message m_1 to many others customers, receiving payments in return. These new angry customers would only have message m_1 to inculpate vendor A. Vendor A would claim that she never sent m_1 to them and that they must have got it from the first resentful customer. As a matter of fact, there will be nothing in m_1 to prove that A is reselling the same forged message all over again.

2.2 Fixing the Protocol

Even though the attacks described above correspond to simple deviations from the protocol description, they represent real threats to parties using the scheme to exchange their items. In e-commerce transactions, neither vendor A nor customer B would want to take the risk of being cheated on. However, the previous attacks can be avoided if a better cryptographic evidence is constructed. This can be done in many ways. Probably the easiest one is just by including the identity of B in m_1, thus linking the message with its intended receiver[1]:

$$A \rightarrow B : m_1 = (\mathbf{B}, desc_{item_A}, enc(k, item_A), w(k), \sigma_1)$$

where:

$$\sigma_1 = sig(k_A^{-1}, (\mathbf{B}, desc_{item_A}, enc(k, item_A), w(k)))$$

This modification suffices to prevent attacks one to three.

3 Buttyán et al's Model: Game Theory and Protocol Games

Syverson's protocol was analyzed by Buttyán et al in [3]. For readability and completeness, we first provide a brief introduction to the game-theoretical model of rational exchange introduced by Buttyán et al. Please refer to [3] for further details. Where possible, we have adopted the same notation as used in [3].

3.1 Protocol Games

The protocol game of an exchange protocol is intended to model all possible interactions of the protocol participants, even the potentially misbehaving actions (i.e., those different from the prescribed by the protocol). A protocol game is constructed from the protocol description. Each of the parties involved in the protocol becomes a player of the protocol game, including the network. Every participant, apart from the network, has strategies to quit, to do nothing, to send a message following the protocol steps, or to send a message deviating from the steps of the original protocol. Each player can send messages which have been defined as *compatible* with the protocol, i.e., messages which are within the context of the protocol. The set of messages compatible with a protocol is formally defined within Buttyán et al's model. Participants can alter the order in which those messages are sent. When the protocol game is over, every participant can assess the profit or the loss they have incurred in, by using a payoff function. Informally, a two-party rational exchange protocol is an exchange protocol in which both main parties are motivated to behave correctly and to follow the protocol faithfully. If one of the parties deviates from the protocol, then she may bring the other, correctly behaving party in a disadvantageous situation, but she cannot gain any advantages by her misbehavior. Buttyán et al define the concept of rationality in terms of a Nash equilibrium in the protocol game.

[1] As usual, we assume that A's identity is implicit in m_1, since the message contains A's signature. If this was not the case, then we must include it explicitly to avoid a different class of attacks.

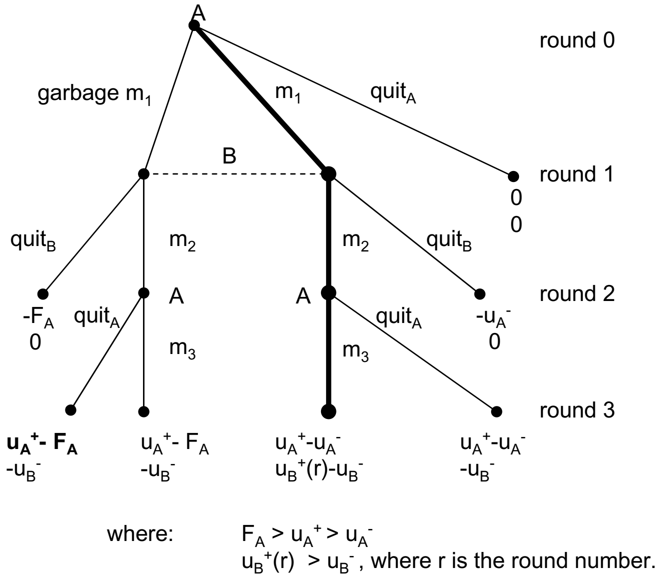


Fig. 2. Partial representation of Syverson's protocol game in extensive form

It is required that the strategies that correspond to the behavior described by the protocol form a Nash equilibrium in the protocol game and that no other Nash equilibrium is strongly preferable for any of the participants.

In Fig. 2, we have represented Syverson's protocol game in an extensive form. The tree represents the different moves each participant can make and all the different possible outcomes. The vectors assigned to each terminal node represent the outcome for A (first value) and B (second value) when A and B follow the path of strategies to finish the protocol at that end. These values are given by Buttyán et al in their model.

The values u_A^+ and u_B^+ denote the values that $item_B$ and $item_A$ are worth to parties A and B respectively. In a similar way, the values u_A^- and u_B^- denote the values that $item_A$ and $item_B$ are worth to A and B, respectively. The value of F_A represents the penalty A has to pay when proved to be the author and sender of a forged message m_1. Note that this is possible only when the enhancement described in section 2.2 is added to the protocol description. The model designed by Buttyán et al assume that evidences are well constructed and fails to reflect the actual content of message m_1, as it is described in the protocol.

We have highlighted in the tree the *strategy profile* for A and B which would result in a rational exchange of items $item_A$ and $item_B$. As noted by Buttyán et al's model this profile constitutes a Nash Equilibrium so, by definition, neither of the players would want to deviate from it. Therefore Buttyán's model serves to formally define rationality and to prove that Syverson's enhanced protocol is a rational exchange protocol. However, as we can see in the diagram, A is not motivated to be fair to B in the last round of the protocol. A could threat B to execute $quit_A$ or to get delayed in sending m_3 to B. This is a feasible threat. In the following section we will see the consequences this threat has on B's behavior and on the equilibrium previously found.

4 Discussion

In this section we intend to analyze new aspects of the protocol, extending the model described above and introducing new parameters into the extensive form of the protocol game.

4.1 Other Nash Equilibriums in Repeated Executions

As we mentioned before, A is not motivated to be fair to B in the last round of the protocol. Therefore, A could threaten B to quit or to delay sending m_3 to B. B would then be safer quitting the protocol before round 2 and aborting the exchange. The best response that A can give to B's quit strategy is to quit as well. Therefore strategies $(alwaysquit_A, alwaysquit_B)$ also form a Nash Equilibrium for the protocol game described in Fig. 2. In order to solve this issue, A should be given some kind of incentive to be fair to B in the last round of the protocol. This incentive may have the form of a "reputation factor", surely managed by external parties, which will be made public. This reputation factor will give entity B the means to place an appropriate level of confidence in entity A. In the extended model which we propose, participant B will be forced to form a conjecture or set of beliefs over A's behavior based on its reputation factor (this implies repeated scenarios) or similar (other criteria for first time executions). This way, our model will capture the uncertainty B has over A's behavior at the last step in the protocol. A certain value δ will define the probability that entity B assigns to the event of A sending m3 at round three according to the protocol description. Consequentially, the value $(1-\delta)$ will determine the probability of A getting delayed in sending m_3 or not sending it at all. The fact that, most likely, entity A will be using Syverson's protocol to interact with a variety of entities B, in various occasions (repeated executions) can also help B in adjusting the value for δ.

4.2 Reputation Factors

Any given entity wishing to participate in a security protocol must place a degree of confidence in the protocol design as it is not possible to model and anticipate all malicious protocol attacks. Although a well verified and validated protocol will offer the participants enough guarantees to preserve security, nevertheless any given protocol carries a reputation factor. In the case of Syverson protocol, B is the entity taking a greater risk so B must be sure that A will behave according to the protocol description and B must also be confident that the protocol is well designed so A cannot deviate without being noticed. Our extended model will capture the level of uncertainty that participant B holds over the robustness of the Syverson protocol. A certain probability α can be considered as the level of confidence a customer B has in the protocol's design. It represents the possibility that a forged message sent by A could actually be part of the protocol execution, breaking the rationality property and enabling A to finish the protocol in a advantageous position. It establishes the fact that the protocol could present unknown vulnerabilities identified only by A. It is assumed that the kind of forged message A could send is different from gm_1 (garbage m_1) in Fig. 2, because gm_1 will always be detected and penalized. This fact would be represented as a new branch in the tree, labelled as *unpredictable* gm_1.

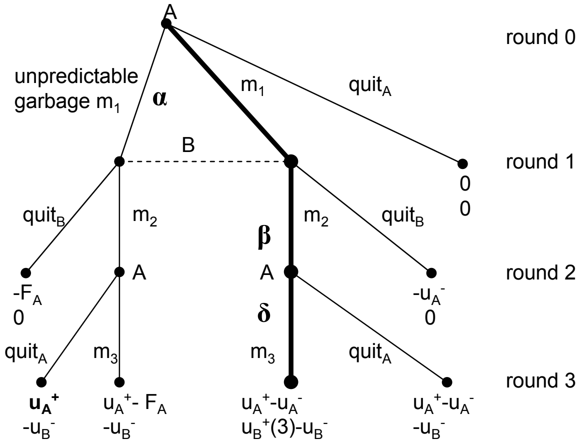
4.3 Extended Local History Records

It is specified in Buttyán et al's model that each player creates a history record of all the events that were generated by her and the round number of their generation. Possible entries in the history record file of protocol participant A would be $\mathsf{send}(m_1, party_B)$ or $\mathsf{rcv}(m_2)$, in round r. Based on the entries stored in this record each player is allowed, or not allowed, to send a particular message. For instance, a valid digital signature s_A, can only be generated by A, therefore, B can send a message containing s_A iff B received a message containing s_A *earlier and during the current protocol execution.* Indeed, as the model was defined, this history record is newly created for each protocol run so information received in previous protocol runs is discarded at the end of the execution. It is precisely this aspect of the model that hides the protocol vulnerabilities described in section 2.1. Any participant of the protocol could have compatible messages from previous runs and will be able to use them. Buttyán et al's model, by discharging old messages, fails to detect attacks as those described in section 2.1.

Furthermore, the protocol participants will have to be trained to identify such fraud messages and discharge them. Within Buttyáns model, each participant is given a program π_i to execute at each step in the protocol execution. The logic of these programs must be extended to include various tests to verify whether evidences are properly constructed [7] and the appropriate mechanisms designed to be able to reject old or/and forged messages. The protocol design and verification processes have to guarantee the participants some essential required security properties.

4.4 Entity A's Conjecture

In a similar way, A will be asked to conjecture about B's behavior at round two of the protocol. B could, at step two, continue or quit. A will assign a certain probability β to the event of B sending m_2 at round two of the protocol. Therefore (1-β) will be the probability of B misbehaving at round two by quitting the execution. Note that A can always verify the freshness of m_2 as it is an irrevocable receipt and an irrefutable proof of origin of m_1. B cannot cheat sending the wrong m_2 as this will always be detected and punished. For simplification, we have omitted this path in the tree.



Where B’s conjectures are:

α = prob{A is able to misbehave at round one}

δ = prob{A sends m3 at round three of the protocol}

and A’s conjectures are:

β =prob{B sends m2 at round two of the protocol}

Fig. 3. Partial representation of Syverson’s protocol game in extensive form

5 Extended Model Based on Dynamic Games of Imperfect Information

By considering the protocol as a game of imperfect information, we are forcing both entities, A and B, to form conjectures about each other, and also about the correctness and *robustness* of the protocol. Those conjectures will be represented by probabilities α, β, and δ introduced before. See Fig. 3 for a partial extensive-form representation, of the Imperfect Information Protocol Game. Fig. 3 extends Fig. 2 showing a completely new scenario. In 3 there exists the possibility that A could send B a forged message, for which A would obtain message m_2 in return and for which entity A will not be fined or penalized. This would only be possible by stepping outside the previous model and assuming that there still are vulnerabilities in the protocol design. A could well identify such flaws and try to take advantage of them. However, it is not always clear that there still exist vulnerabilities and that entity A could recognize them. So the uncertainty B holds over the protocol correctness can be captured and modelled by this new branch in the tree. We have omitted the other gm_1 path to simplify the analysis of this new aspect. The following calculations will establish the criteria for A and B to be participants of the protocol, and they will also help to define different equilibria for the different values of α, β and δ, from which neither of the two entities will want to deviate.

Entity B can formulate the following considerations for each one of the possible strategies:

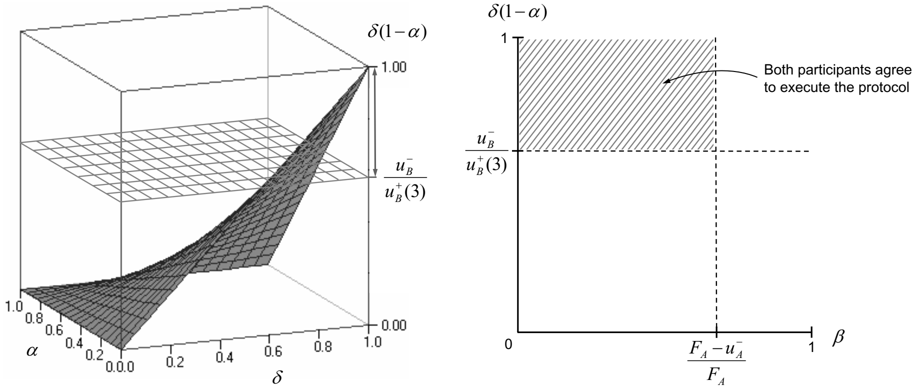


Fig. 4. Graphic representation of Nash equilibrium space

$$\begin{aligned} EG(quit_B) &= 0 \\ EG(m_2) &= \alpha * [(1-\delta) * (-u_B^-) + \delta * (-u_B^-)] + \\ &\quad (1-\alpha) * [\delta * (u_B^+(3) - u_B^-) + (1-\delta) * (-u_B^-)] = \\ &\quad u_B^+(3) * \delta * (1-\alpha) - u_B^- \end{aligned} \tag{1}$$

Where $EG(m)$ represents the expected gain after message m. Note that:

$$EG(m_2) \geq EG(quit_B) \Leftrightarrow u_B^+(3) * \delta * (1-\alpha) - u_B^- \geq 0$$

Therefore:

$$EG(m_2) \geq EG(quit_B) \Leftrightarrow \delta * (1-\alpha) \geq u_B^- / u_B^+(3) \tag{2}$$

The graph shown in Fig. 4 (left) represents the function $\delta*(1-\alpha)$. For all those values α and δ for which the graph is over the value $u_B^-/u_B^+(3)$, the best strategy for B would be to carry out the exchange and follow the protocol description. Below that line, B's best strategy is to quit, as the expected gain value would be less than zero. In a similar way, A can formulate the following considerations: For each one of the possible strategies that A can follow, the expected gains would be (ugm_1 stands for unpredictable garbage m_1):

$$\begin{aligned} EG(m_1, quit_A) &= \beta * (u_A^+ - u_A^-) + (1-\beta) * (-u_A^-) = \beta * u_A^+ - u_A^- \\ EG(m_1, m_3) &= \beta * (u_A^+ - u_A^-) + (1-\beta) * (-u_A^-) = \beta * u_A^+ - u_A^- \\ EG(ugm_1, quit_A) &= \beta * (u_A^+) + (1-\beta) * (-F_A) = \beta * (F_A + u_A^+) - F_A \\ EG(ugm_1, m3) &= \beta * (u_A^+ - F_A) + (1-\beta) * (-F_A) = \beta * u_A^+ - F_A \end{aligned}$$

Note that the omitted strategies (ugm_1, m_3) and $(ugm_1, quit_A)$ do not affect the following rationale, as they are strictly dominated strategies where the payoff function $(\beta * u_A^+ - F_A)$ is always less than zero. Furthermore, the strategy (ugm_1, m_3) is also a strictly dominated strategy with a payoff value below zero. The strategy $(ugm_1, quit_A)$ plays an important role, as there will be a threshold value for β to establish whether A, having the opportunity to attack the protocol, would take the risk to be detected at the first step of the protocol.

From previous expressions, we obtain that:

$$EG(ugm_1, quit_A) \geq EG(m_1, m_3) \Leftrightarrow \beta \leq (F_A - u_A^-)/F_A \tag{3}$$

Fig. 4 (right) shows the intersection between the space of values for α and δ from Fig. 4 (left) and the new threshold for A's conjecture β. The shadowed area represents the equilibrium space. There are infinite Nash equilibriums depending on the different conjectures, all of them formed by mixed strategies. The values of α, β and δ will be regulated by the public reputation factor, so they will be in favor of the exchange or not *at the same time*. This is: if A's reputation is not too good or if it is the first time A participates in an exchange, B will show a high level of distrust, but A will be aware of this and will adjust the value of β accordingly. When considering a repeated scenario, the total profit for participants A and B is calculated adding all the profits obtained at each one of the protocol executions. Misbehaving will then have a global impact on the total expected gains. The values for α, β, and δ serve to formally model such an effect.

6 Conclusions

In this paper, we have studied new aspects of Syverson's protocol. First, we analyzed the evidence tokens constructed during the protocol execution, which were meant to preserve rationality in case of misbehavior. We found some vulnerabilities related to those, and provided an enhancement which overcame the problems. Secondly, we formally considered the most common context for Syverson's protocol, which is based on repeated scenarios. These were formally taken into account when studying the participants' behavior. In repeated scenarios, participants care about their reputation, so it is possible to analyze part of their future conduct based on such a factor. Finally, we studied a new aspect never modelled before: The uncertainty over the protocol's robustness and the impact this has on the participants behavior.

Our model brings into consideration Syverson's protocol reputation when assessing the risk undertaken when it is executed. We have taken Buttyán et al's model, based on game theory, and extended it to add three new parameters which serve to analyze the new aforementioned aspects. A completely new space of Nash equilibrium has emerged as a result.

References

1. L. Buttyán, J.P. Hubaux. "Rational exchange– a formal model based on game theory". In *Proceedings of the 2nd International Workshop on Electronic Commerce*, LNCS Vol. 2232, p. 114. November 2001. Springer-Verlag.
2. L. Buttyán, J.P. Hubaux. "A formal Analysis of Syverson's Rational Exchange protocol". In *Proceedings of the 15th IEEE Computer Security Foundations Workshop*, pp. 181–193, June 2002.

3. L. Buttyán, J.P. Hubaux, S. Čapkun. “A Formal Model of Rational Exchange and Its Application to the Analysis of Syverson’s Protocol”. *Journal of Computer Security*, Vol. 12, Issue 3/4, 2004, pp. 551–588. IOS Press, 2004.
4. D. Dolev and A. Yao. “On the security of public-key protocols”. *IEEE Transactions on Information Theory*, Vol. 29, pp. 198–208, 1983.
5. D.E. Denning. “The limits of Formal Security Models”. National Computer System Security Award Acceptance Speech, 1999.
6. R. Gibbons. *Game Theory for Applied Economists.* Princeton University Press, 1992.
7. [ISO/IEC 13888-3] Information Security. Security Techniques. Non Repudiation. 1997.
8. M. Jakobson. “Ripping coins for a fair exchange”. In *EUROCRYPT’95*, LNCS Vol. 921, p. 220. May 1995. Springer-Verlag.
9. S. Kremer and J.F. Raskin. “A game Approach to the Verification of Exchange Protocols”. In *Proceedings of the 1st Workshop on Issues in the Theory of Security*, July 2000.
10. S. Kremer, O. Markowitch, and J. Zhou. “An intensive survey of fair non-repudiation protocols”. *Computer Communications*, 25(17):1606–1621. Elsevier, 2002.
11. S. Kremer. “Formal Analysis of Optimistic Fair Exchange Protocol. PhD Thesis. UniversitLibre de Bruxelles. Facultde Sciences. 2003-04.
12. R.M. Needham. “The changing environment for security protocols”. *IEEE Network*, Vol. 11, No. 3, pp. 12–15, May-June 1997.
13. Petteri Nurmi. “A framework for online reputation systems”. Department of Computer Science, University of Helsinki. March 2005.
14. H. Pagnia and F.C. Gärtner. “On the impossibility of fair exchange without a trusted third party”. Darmstadt University of Technology, Department of Computer Science. Technical Report TUD-BS-1999-02. March 1999.
15. P. Syverson. “Weakly secret bit commitment: Applications to lotteries and fair exchange”. In *Proceedings of the 11th IEEE Computer Security Foundations Workshop*, pp. 2–13, 1998.