Modern fair exchange protocol design: Dealing with complex digital items

Fabio Piva[1], **Ricardo Dahab**[1]

[1]Instituto de Computação – Universidade Estadual de Campinas (UNICAMP)
Caixa Postal 6176 – 13084-971 – Campinas – SP – Brasil

{fpiva, rdahab}@ic.unicamp.br

Abstract. *Fair exchange protocols are essential for ensuring fairness (i.e., atomicity) in exchanges concerning digital items between parties communicating through asynchronous channels. Although often regarded as generic bit streams, such items are usually complex artifacts that carry information relevant to a particular context – which may be of semantic, perceptual, legal, financial or functional nature, for instance – and reflect specific aspects that might interfere with the protocol designed for exchanging them. In this work we approach fair exchange protocol design by taking into account the intrinsic characteristics of digital items, as opposed to relying on the conventional generic bit stream assumption. Our discussion focuses on how several item properties may affect the exchange of digital items.*

1. Introduction

Fair exchange protocols were proposed [Asokan 1998] as a solution to the problem of exchanging digital items in asynchronous channels. To be considered *fair*, the protocol must ensure that, at the end of the exchange, either both participants acquire their desired items or none gain any additional information about those items. Different approaches have been taken to fair exchange protocol design, but it has been proved that true fairness can only be achieved between two participants by relying on a trusted third party (TTP, or trustee) of some kind [Pagnia and Gärtner 1999].

Since then, several fair exchange protocols have been published [Zuo and Li 2005, Payeras-Capellà et al. 2006, Avoine and Vaudenay 2004, Ray and Ray 2000]. Most of them address efficiency issues caused by the need of a TTP by reducing its participation in the protocol to a minimum; those protocols, regarded as *optimistic fair exchange protocols*, only require TTP intervention in case something goes wrong during the protocol run – such as a misbehaving party trying to cheat its counterpart or failure on the communication channel.

Even though fair exchange protocols have been widely studied, most designs still follow the same approach as Asokan's original work, which considered the exchanged items to be generic bit streams with few or no particular properties of relevance to protocol design. We believe, however, that in most current contexts items do have inherent complexity that may interfere with transactions, and exhibit characteristics that either make the exchange easier, or become obstacles for enforcing successful (fair) outcomes. Those properties are usually left aside during protocol design for the sake of "simplicity" of explanation, which often results in proposals that are inaccurate, inefficient or inadequate [Micali 2003, Zhou et al. 1999, Zuo and Li 2005, Payeras-Capellà et al. 2006] for most real-world applications – where items are far from generic bit streams. To the

extent of our knowledge, only a few works have taken into account how the nature of the exchanged items may impact transaction outcomes [Vogt 2003, Bottoni et al. 2007, Piva et al. 2009, Piva and Dahab 2011, Piva and Dahab 2013].

In this paper we discuss several of the most common properties found on digital items of interest in many currently fair exchange-focused applications – such as e-commerce, contract signing etc – and how they interact with other similarly complex items and with the protocols designed to exchange them. It is our goal to show that, by focusing on the inherent aspects for the items being exchanged – an approach to which we further refer to as *item-aware protocol design*, as opposed to the conventional *generic item protocol design* – the designer may be able to tackle context-specific problems and to avoid common protocol design pitfalls. We are aware that studies providing formal frameworks for complex tasks such as protocol design may be able to provide more-reliable foundations for further development of the state-of-the-art; regardlessly, several previous authors have been able to contribute to a better understanding of protocol design by providing useful guidelines with similarly-informal discussions on the topic [Louridas 2000, Abadi and Needham 1996, Woo and Lam 1994, Piva et al. 2009], as is the case of our contribution.

The remainder of this work is organized as follows: In Section 2 we present a collection of several common properties held by digital items, and explain how they might affect fairness. In Section 3 we provide a fair exchange-oriented discussion on how a hypothetical transaction following Asokan's model would be affected when each exchanged item presents each of those properties; we also provide a non-exhaustive list of statements that include possible advantages from which protocol designers might benefit from taking each interaction into account, as well as possible difficulties they might introduce for ensuring fairness. An example of item-aware protocol design for a hypothetical real-world scenario is presented in Section 4. We conclude our discussion in Section 5 with a few remarks and suggestions for future work on the subject.

2. Common properties of digital items

In this section we look into the items to be exchanged, in light of some specific, commonly-observed properties. As we shall discuss, the level of fairness obtained in fair exchange protocols may highly depend on the characteristics of the items being exchanged themselves, and so these properties should always be taken into account during the process of proposing a new protocol.

2.1. Idempotency/Copiability

Perhaps the most relevant difference between digital and physical items is that the first ones are easy to copy. In fact, many fair exchange protocols rely on the fact that receiving a digital item more than once is the same as receiving it once – as digital items are **idempotent** (or **copiable**) [Asokan 1998]. Under that perspective, digital items are indeed essentially sequences of bits, and thus creating a copy of a particular idempotent item makes for an identical copy of that item itself.

Although idempotency can sometimes be an advantage for protocol design – see Section 2.3 to see how copies of items can help enforce fairness – the opposite may also be true. For instance, dispute resolution becomes a hard matter when a participant is not able to return an item without possibly retaining a copy for himself – in case of a mistaken delivery, for instance. When physical items are exchanged (as in physical products

buying/selling), any mistakes can be easily undone by simply returning the wrong item in exchange for the correct one. It is, therefore, easy to address – by means of return policies – situations in which parties become unsatisfied with the outcome of a particular transaction that concerns physical items.

In exchanges concerning digital items, however, this might not always be the case. Unless the wrong item is *revocable* (see Section 2.4 for revocable items), return policies often do not apply [Amazon Legal Department 2005]. This fact requires that, in order to avoid undesired outcomes, fair exchange protocols must predict and minimize "buying a pig in a poke[1]" scenarios, in which a party is left unsatisfied with the acquired item – which can be particularly hard for *indescribable* items (see Section 2.2 for describability).

2.2. (In)Describability

Fair exchange protocols typically require that a description of each item must be publicly available (or directly delivered) to parties before the exchange takes place. Such protocols include a critical step – the *item validation* step [Piva and Dahab 2011, Piva and Dahab 2013] – in which a party is usually required to check whether the item she has received (or is about to receive) satisfies that description or not.

Such description must, however, be univocal if a party is to be assured about the outcome of the exchange. A univocal description is regarded as a set of characteristics that uniquely define an item – with no other similar item being able to entirely satisfy that particular description. When providing a univocal description of any form for a particular item is possible, we regard it as being **describable**. If only non-univocal descriptions are possible instead, the interested party might be misled into inaccurately validating an item that, while satisfying said description, is nevertheless inherently different than the one she expects to receive. We regard items that can only be described by non-univocal descriptions as being **indescribable** [Bottoni et al. 2007, Piva and Dahab 2011, Piva and Dahab 2013].

For an example of this issue, suppose that a party P is willing to obtain a picture i of the model Lena Söderberg, famous for its appearance as case of study in image processing literature. P could be satisfied with a description $desc(i)$ consisting of the following list of words: Lena Söderberg, image processing muse, model, famous, hat, PNG file, face. After engaging counterpart Q in the exchange and delivering her own item i' (possibly some sort of digital payment, for instance), P would expect to receive the file pictured in Figure 1(a), but could be surprised by the delivery of Figure 1(b) instead. One could notice that the problem could be easily solved by adding the word "color" to the description, but even in that case Figure 1(c) would still be a candidate for delivery.

It has been noted that not only pictures, but all forms of multimedia content are naturally **indescribable** – for univocal descriptions for such items are hard to obtain, if not impossible [Piva and Dahab 2011, Piva and Dahab 2013]. The radio edit or live version of given song, for instance, could be mistakenly delivered instead of the expected album version of the same song; a movie could be an unrated version, or a remake of the same story. In fact, even the most precise description of an indescribable item would still leave room

[1] "To buy a pig in poke" is an idiom associated to a scenario in which an individual, upon trying to purchase a good-quality pig in a bag, ends up with a low-quality pig because he or she did not carefully check what was in the bag before paying for it – believing the pig's previous owner's promises instead.

(a) (b) (c)

Figure 1. Three different files that show pictures of the model Lena Söderberg. Figures (a) and (b) equally satisfy any description that does not mention color properties, and Figures (a) and (c) could be mistaken even if color is mentioned – which could lead to the wrong file being delivered.

for misinterpretation [Bottoni et al. 2007, Piva and Dahab 2011, Piva and Dahab 2013]. This fact alone makes trustee-based validation unsuitable for this type of items, and complicates dispute resolution greatly – as well as fair exchange protocol design.

Stating that a univocal description of an item will be available for parties is, therefore, a dangerous assumption usually made by most fair exchange protocol designers. If one or more of the items being exchanged are indescribable, current approaches that rely on previously-obtained/public descriptions are unable to guarantee that item validation will be robust enough for allowing a party to predict the outcome of the transaction. Since indescribable items – particularly digital music and other forms of multimedia content – are currently of great interest to several e-commerce providers – which usually rely on some instantiation of fair exchange protocols – describability arises as an important issue for future research. In fact, indescribable items have only recently been identified as the protagonists of problematic exchanges and, as such, have received some attention [Bottoni et al. 2007, Piva and Dahab 2011, Piva and Dahab 2013].

2.3. Generatability

Since failure in providing a desirable outcome to all parties in exchanges concerning digital items can be rather difficult to resolve (mostly due to the idempotency property, discussed in Section 2.1), most optimistic fair exchange protocols usually rely on some level of **generatability**, which can be embedded in items. An item is said *generatable* if a party is able to obtain an equally satisfying item – possibly a copy of the intended item, or a different item which substitutes it in every aspect of interest – with the help of a TTP, provided that the affected party is able to prove her commitment to the transaction. An example of generatable item would be a signed contract by both parties of an agreement, which could have the same legal value if signed instead by both one of the parties and the TTP.

Generatable items have received a lot of attention since the proposal of fair exchange protocols. As stated in [Pagnia et al. 2003], an item is said to be *generatable* if it "*can be generated by the trustee in case the receiving party can prove that it has behaved correctly*". The strength of this generatability is defined over the possibility of success: *strong generatability* ensures that the trustee will always be able to generate the item successfully, while *weak generatability* allows failure in the generation, in case of party misbehavior; in such cases, the trustee is able to detect and provide evidence of this misbehavior to the honest party, so that external disputes may be initiated.

Although generatability is not an inherent property of digital items, it can be achieved with the help of several known techniques [Vogt 2003, Avoine and Vaudenay 2004]. In the remainder of this section we revisit a few solutions presented by Vogt et al. [Vogt 2003].

1. *Strong generatability of generic items (with active TTP):* The first approach relies on an active (online) trustee, and can be achieved by the owner party P sending the item i_P, together with description $desc(i_P)$, to the trustee; the trustee then checks the item against the description and, in case of success, stores i_P during the remaining of the exchange. The trustee also signs the provided description and returns this $SIG_{TTP}(desc(i_P))$ to P, who then uses this term as a proof to counterpart Q that i_P can be provided by the trustee if necessary. Although this approach succeeds in providing strong generatability to any describable item, it requires the trustee to keep a copy of every item for every party it is trusted by, which is completely unpractical for larger real-world applications.
2. *Strong generatability of digital signatures (offline trustee):* Another approach relies on verifiable escrow [Asokan et al. 2000] primitives, which are designed to provide strong generatability to digital signatures. This does not require an active trustee, but is restricted to signatures and thus is not straightforwardly applicable to other kinds of digital items.
3. *Weak generatability of generic items (offline trustee):* The last approach considered in [Vogt 2003] also does not require an active trustee, and works well with describable items. The trade-off is that only weak generatability is achievable[2]. In order to accomplish that, the owner P must encrypt the item i_P with the trustee's public key, and sign both this encrypted term and the item description. The obtained term $SIG_P(PU_{TTP}(i_P), desc(i_P))$ is then forwarded to P's counterpart Q, which is able to verify P's signature. In the event of a dispute, Q would provide $SIG_P(PU_{TTP}(i_P), desc(i_P))$ to the trustee, which would first check P's signature. If the check fails, the trustee considers that Q has misbehaved, since he would be able to detect the failure himself; if it succeeds, the trustee tries to decrypt $PU_{TTP}(i_P)$, and to validate the resulting item against the description $desc(i_P)$. If the validation succeeds, the trustee forwards i_P to Q and, if it fails, it considers that P has misbehaved. This method has been applied on several previously-published fair exchange protocol proposals [Ray and Ray 2001, Ray and Ray 2000].

All of these approaches allow different types of items to be made arbitrarily generatable, but share one common characteristic: they require a well-defined (i.e., univocal) description of the item, which makes them unsuitable for indescribable items. We believe that in order to embed generatability into indescribable items, it is also necessary to address indescribability issues. In that context, reversible degradation techniques [Piva and Dahab 2011, Piva and Dahab 2013] seem adequate.

2.4. Revocability

An item is said to be *revocable* if it can be invalidated by a trustee, when specific requirements are met. As with generatability, different levels of revocability may be provided. While the trustee will always succeed in making *strongly revocable* items useless

[2]In fact, "*no efficient method (i.e., without TTP interaction) is known to make arbitrary goods strongly generatable*" [Vogt 2003].

for its receiver, she may fail in revoking *weakly revocable* items; in such cases, the trustee is always sure that the receiver got or can still get the item, which can help further dispute resolution.

As generatability, it is possible to embed revocability into items, specially in particular contexts – such as digital payment applications. In fact, many electronic payment systems provide some level of revocability to electronic cheques [O'Mahony et al. 1997, Asokan et al. 1997]. The combination of generatability and revocability can be particularly constructive for fair exchange protocol design, as we shall further discuss in Section 3; for instance, a trustee may try to generate a weakly generatable item for a cheated party, and in case of failure, she may revoke the issued cheque in order to restore fairness.

2.5. Forwardability

The first proposed fair exchange protocols assumed items to be **forwardable**: An item is said to be forwardable if "P *can send the item directly to* Q*, or it can send it to the TTP; the TTP will be able to verify the correctness of the item with respect to the stated description and either store it or resend it to* Q" [Asokan 1998]. According to this original definition and the arguments presented so far in this paper, we conclude that an item is said to be forwardable if it is both *idempotent* and *describable* – which also leads us to conclude that, although simple bit strings with no further meaning, for instance, may be considered forwardable [3] (as stated in Asokan's original work), this might not always be the case for more complex digital items – which is usually the case in real-world transactions.

As with describability, assuming digital items to be forwardable may be dangerous. Several proposed fair exchange protocols for generic items follow the model established by Asokan, and also overlook this issue [Zhou et al. 2000, Garay et al. 1999, Markowitch and Kremer 2001, Markowitch and Saeednia 2002] – ultimately resulting in security flaws that can be explored by an attacker [Piva et al. 2009].

2.6. Co-dependency

In fair exchange protocols, items are usually unrelated in any way – which usually means that their values are not linked to each other. There are, however, particular transactions in which one item is only valuable to the interested party if the other item delivered in the exchange is also valuable to the counterpart. We regard these special items as being **co-dependent** from one another.

Examples of exchanges concerning co-dependent items may be found in contract signing protocols. In such contexts, parties are usually interested in obtaining their own signature, as well as the counterpart's, on some digital contract C. In the case of a two-party contract signing, for instance, P would be interested in obtaining $SIG_P(SIG_Q(C))$, while Q would desire to receive, say, $SIG_Q(SIG_P(C))$.

Since both items depend on the same basic information C to be constructed, a misbehaving party might find it particularly difficult – as opposed to exchanges involving generic items – to tamper with her own item in order to end up with a valid

[3] If an item is essentially a particular bit string with no particular complex function – as opposed to a multimedia file, for instance, which brings inherent perceptual information in it – a cryptographic hash would be enough to univocally describe it. However, bit strings with no particular function are rather rare in real-world applications for exchanging digital items.

item in exchange for nothing valuable (i.e., garbage-for-gold attacks [Piva et al. 2009]); most tampering attempts would in general result on both parties receiving invalid items $SIG_P(SIG_Q(C'))$ and $SIG_Q(SIG_P(C'))$, if so – which does not violate fairness requirements. Therefore, fair exchange protocols concerning co-dependent items might be easier to design – at least in some level – since these so-called garbage-for-gold attacks would be harder to perform (notice that this might not be the case for other attacks – such as that of a party mischievously abandoning the transaction before sending her own item and after receiving her counterpart's).

3. Notes on the interaction between properties and impacts for fair exchange

In Section 2 we discussed a few of the most relevant item properties concerning several fair exchange-related scenarios. In this section, we discuss how two-party protocol design might benefit from the interaction between two items bearing each of those properties, and try to shed some light over what could be gained or lost from the interaction between them by approaching protocol design in item-aware fashion.

For the remainder of this section, we assume that two parties P and Q wish to exchange two items i_P and i_Q. We also assume that P commits to the transaction first, giving up i_P (to which we further refer as first item) before Q gives up i_Q (to which we further refer as second item). We keep our presentation brief in order to ease further reference, basing our statements on the arguments presented so far.

i_p \ i_q		Idempotent	Indescribable	Generatable		Revocable		Forwardable	Co-dependent
				Weak	Strong	Weak	Strong		
Idempotent		1	1, 2	1, 3(a)	1, 3(b)	1	1	1, 4	[redacted]
Indescribable		1, 2	2	2, 3(a)	2, 3(b)	2	2	1, 2, 4	[redacted]
Generatable	Weak	1	2	3(a)	3(b)	7	7	1, 4	[redacted]
	Strong	1	2	3(a)	3(b)	7	7	1, 4	[redacted]
Revocable	Weak	1, 5(a)	2, 5(a)	3(a), 5(a)	3(b), 5(a)	5(a)	5(a)	1, 5(a), 4	[redacted]
	Strong	1, 5(b)	2, 5(b)	3(a), 5(b)	3(b), 5(b)	5(b)	5(b)	1, 5(b), 4	[redacted]
Forwardable		1	1, 2	1, 3(a)	1,3(b)	1	1	1, 4	[redacted]
Co-dependent		[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	6

Table 1. Interactions between item properties in optimistic fair exchange protocols (see below for details)

Table 1 shows the impact that the properties discussed in Section 2 may have on an optimistic two-party fair exchange protocol. Co-dependent items, specifically, are special items that only make sense when considered in pairs – which is why we omitted their comparison with other properties. The following statements apply to Table 1:

1. No return policies apply for idempotent items (first or second), so fair exchange protocols should be robust enough to minimize unexpected outcomes. Dispute resolution should be carefully designed.
2. Item validation is hard to achieve for indescribable items. Since strong fairness might be hard to guarantee for both the owner (if this is a first item) and the receiver (if this is a second item), as Asokan's protocols are not inherently equipped for these kinds of items, special-purpose techniques may be required as enhancements for practical deployment [Piva and Dahab 2011, Piva and Dahab 2013]. In particular, the item validation step should receive special attention during protocol design.

3. When the second item is generatable, the first party can always be assured that, in case of exceptions, the TTP might be able to help her with obtaining the desired item. Therefore, generatable items are suitable as second items in fair exchange protocols.
    (a) If the item is only weakly generatable, the TTP might fail in retrieving it for the interested party. Therefore, only weak fairness is guaranteed.
    (b) If the item is strongly generatable, however, the TTP always succeeds, provided that the interested party behaves honestly. Strong fairness is achievable with robust dispute resolution.
4. Since we claim that forwardable items are also required to be describable, those items are better-suited as second items. Item validation for describable items is often simpler to perform than for indescribable ones. However, since they are also idempotent, Statement 1 may apply – unless when omitted in Table 1.
5. Revocable items make good first items, since they can be invalidated by a TTP if something goes wrong after their delivery (such as the second item being intentionally kept by a malicious Q). This fact makes them particularly interesting for exchanges in which the second item may be problematic – such as indescribable items, for instance.
    (a) If the item is only weakly revocable, the TTP might fail in invalidating the item. Therefore, only weak fairness is guaranteed.
    (b) If the item is strongly revocable instead, the TTP always succeeds, provided that the sender behaved honestly. Strong fairness is achievable with robust dispute resolution.
6. Co-dependent items only make sense in pairs and, as such, this fact alone may help to ensure fairness to both parties. Since they have their value linked to each other, usually strong fair exchange can be accomplished even with minimalistic protocol design.
7. As discussed in Section 2.4, when the first item is generatable and the second one is revocable, accurate trustee-based dispute resolution can be implemented as a means for enforcing fairness for exceptional transactions.

In particular, item-aware protocol design focuses on how item properties would interact when two items i_P and i_Q were to be exchanged as proposed by Asokan and later authors. As we can see, when digital items are not regarded as generic objects, much information can be gained from a thorough analysis of their inherent aspects. We shall further illustrate this claim in Section 4, by providing an example of item-aware protocol design for a hypothetical real-world application.

It is important to notice that, in this approach, the order in which items are to be exchanged matters. For instance, the claim that revocable items are suitable for being released first in transactions concerning them; such items behave much like physical products, which can be returned to stores in situations where the buyer is not satisfied with the purchase. Therefore, revocability overcomes the difficulties introduced by the idempotency property shared by most digital items, which could result in a party keeping functional copies of a possibly unsatisfying item. For the same reason, embedding revocability into digital items seems to be a good solution for exchanges in which that same item is also idempotent.

Only recently effective item validation methods for indescribable items began to emerge [Bottoni et al. 2007, Piva and Dahab 2011, Piva and Dahab 2013], which makes

most previously proposed fair exchange protocols unsuitable for them – unless indescribability is addressed somehow. Even when such items are exchanged for revocable items, no guarantee can be given to the owner against possible false-positives that might occur during item validation – as exemplified in Section 2.2 and illustrated in Figure 1. In that context, a TTP would find itself in an undecidable situation: The buyer would ask for dispute resolution, claiming that he did not receive the intended item and therefore his own item should be revoked. The seller, however, would claim that she behaved honestly and delivered the item as described – and so having the payment revoked would leave her in an unfair position. Simply demanding the seller to send the expected item to the buyer would also be unfair since, in that case, the buyer would have obtained two items and paid only for one.

Embedding some level of generatability into digital items has been the most common solution for mitigating unexpected outcomes in previously proposed protocols [Ray and Ray 2000, Nenadic et al. 2005, Ateniese 1999]. Producing generatable items is only particularly useful for fair exchange, however, if the enhanced item is intended to be released last by its possessing party; in general, there shall be no practical gain in endowing a first item with generatability – provided that a robust item validation step is implemented for that particular item – which should be taken into account when designing a new protocol.

We should notice that "perfect" fair exchange (in the sense that very few fairness violations might occur due to either party misbehavior or technical faults) might be more easily achieved when a revocable first item is exchanged by a generatable second item. Protocols designed with these particular kinds of items in mind [Vogt 2003] benefit from less-complicated dispute resolution subprotocols, as well as possibly more-efficient designs regarding the number of required transactions for an average successful exchange – advantages that might be lost if generic item protocol design is used instead.

4. A practical example of non-generic protocol design for digital items

In this section we provide an illustrative example of item-aware protocol design and how the process of designing a fair exchange protocol for a hypothetical context can benefit from taking into account inherent aspects regarding the items of interest. We conduct our example under the discussion presented in Section 3. We emphasize that, rather than providing a formal framework for fair exchange protocol design, our contribution provides an alternative approach to this task that is novel in the sense that it takes into account the complexity of the items being exchanged – as opposed to the conventional, arguably oversimplified approach that regards them as generic bit streams.

4.1. Context description and relevant items' properties

We suppose the example protocol is meant for the electronic purchase/sale of some form of multimedia content (such as a digital audio file, for instance), and that the transaction is to be performed between two parties P (the buyer) and Q (the seller). Therefore, the items to be exchanged in the protocol are the digital payment i_P and the multimedia file i_Q.

We also assume the following properties apply for each item of interest: The payment i_P is **strongly revocable** – a reasonable assumption supported by several currently implemented third party digital payment systems [O'Mahony et al. 1997,

Asokan et al. 1997, Vogt 2003, Wang and Guo 2004]. As for the multimedia content i_Q being purchased, we assume it to be both **indescribable** and **idempotent**, as supported by previously published results on the topic [Bottoni et al. 2007, Piva and Dahab 2011, Piva and Dahab 2013].

For the remainder of this section, we illustrate how the statements presented in Section 3 (and summarized in Table 1) may help the hypothetical protocol designer to take advantage of – or tackle security problems that may arise from – inherent item complexity by acknowledging these three properties.

4.2. Aspects that require special attention during protocol design

As suggested in Section 3 (Statement 6), revocable items are good choices as first items in fair exchange protocols – since they provide "step back" mechanisms to the party who is committing earlier in the protocol; this is the reason behind our choice of placing the payment as the first item to be revealed in the transaction – i.e. before i_Q is delivered by the seller. For that reason, we design our example protocol so that the payment is to be performed by the buyer during an in-transaction step with the help of a trustee-provided payment system able to enforce revocability. This approach is currently implemented in many real-world e-commerce applications and widely accepted amongst many well-known content providers (both Amazon.com and iTunes Store, for instance, offer Paypal support for their buyers).

That naturally leads to the multimedia content, which is both idempotent and indescribable, the placement as second item in the protocol. By referring to Table 1 we are able to conclude, from the cells that result from the intersection between the two columns corresponding to i_Q's properties and the line corresponding to i_P's revocability, that the issues raised in Statements 1 and 2 apply to our example scenario.

Statement 1 brings to our attention the fact that return policies usually do not apply for idempotent digital items – a fact that has become common practice in real-world applications that deal with digital idempotent items [Amazon Legal Department 2005]. This creates, in our example, a context-specific requirement to take special care with how the protocol implements dispute resolution and item validation, in order to reduce the odds of a customers buying "a pig in a poke".

However, as Statement 2 emphasizes, item validation can be hard to implement for indescribable items [Bottoni et al. 2007, Piva and Dahab 2011, Piva and Dahab 2013]. For that reason, and in order to design a protocol that offers robustness against "pig in a poke" purchases, specific-context techniques for item validation may be required during protocol design. A suitable example of such technique for our instance would be the reversible degradation method [Piva and Dahab 2011, Piva and Dahab 2013], which circumvents indescribability issues by embedding some degree of generatability (see Section 2.3 for remarks on generatability) to the item – while reducing the effects of non-univocal descriptions (which, as discussed in Section 2.2, are hard to provide for indescribable items) on the transaction.

4.3. Protocol suggestion

With these requirements in mind, we provide the following protocol (illustrated in Figure 2) as a solution for our example scenario. We assume that a previous authentication step has been performed between P and Q before the illustrated transaction takes

place (which reflects the usual requirement of a buyer creating and logging into a personal account on the seller's website, for instance), in order for the transaction to take place. We also assume that the buyer has already searched seller's website for the audio file he desires to purchase, and believes it to be – based on a non-univocal description of the product – i_Q (which may be or may not be the case, since the product is indescribable [Piva and Dahab 2011, Piva and Dahab 2013]). Also, ID_{i_Q} is the product number that identifies i_Q in Q's system. Finally, $Trans\_Num$ stands for the usual transaction label that uniquely identifies this transaction.

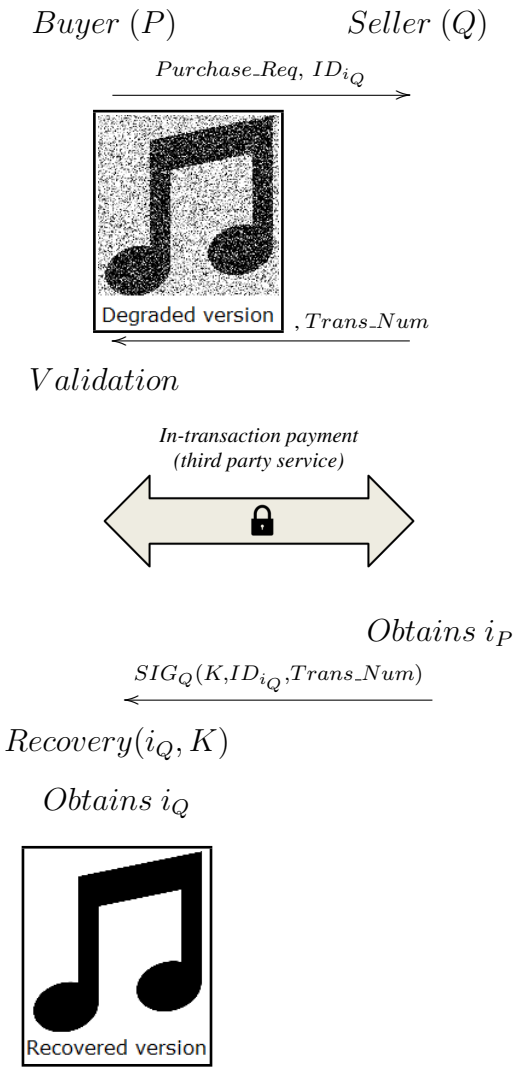

Figure 2. Example of item-aware protocol design in the context of digital audio purchase/sale.

As Figure 2 illustrates that, by acknowledging the special characteristics of the items of interest, the careful designer can more-easily focus on solving context-sensitive issues – which might be of use in the task of avoiding common protocol design pitfalls. The suggested protocol relies on a third-party provider for the payment step (thus ensuring revocability) and on the reversible degradation method as a means of circumventing indescribability (which ultimately introduces some robustness against unsatisfactory outcomes on the behalf of the buyer and the subsequent impact of no-return policies, common

to scenarios concerning idempotent digital items). Also, since some degree of generatability is also enabled by the use of reversible degradation in the validation step, dispute resolution – when required – should be easier to enforce (Statement 3).

In particular, the use of reversible degradation in this protocol allows the buyer to obtain a sufficiently degraded (i.e., worthless), but still fully-playable version of i_Q before payment – which he can then listen to in order to make sure i_Q is in fact the product he intends to pay for. If it is, the buyer proceeds with the protocol by paying for the recovering key K that will be used as input, together with the degraded version of i_Q, in the recovery process that restores i_Q to its full original quality. Further details on the reversible degradation concept can be found in [Piva and Dahab 2011, Piva and Dahab 2013].

If the degraded version brings the buyer to realize, however, that i_Q is not in fact the product he desires, he can simply abort the protocol without paying for (and thus without obtaining) K – which ensures fairness for the buyer. Because the full quality version of i_Q cannot be obtained from the degraded version without K, seller's fairness is also guaranteed. Exceptional outcomes for the transaction would include, for instance, situations in which $K' \neq K$ is delivered after payment – which would prevent the buyer from successfully recovering i_Q; but even in this scenario, dispute resolution would be simple to accomplish (since no "wrong product" – only a wrong key – had been delivered, no issues concerning no-return policies apply; the judge would be able to settle the situation either by demanding the correct K from seller, or by revoking i_P).

As a final note, we stress that the protocol illustrated in Figure 2 is not the focus of this contribution – serving instead the purpose of illustrating how the proposed item-aware approach to protocol design can be of use to modern fair exchange deployment. Therefore, we intentionally do not include any further discussion (specifically, formal proof of security) for the suggested protocol.

5. Conclusion and future work

Fair exchange protocols are suitable for exchanging digital items in a fashion that disallows one party from benefiting from either faults or misbehavior – which is by itself a difficult task. As we have discussed, we firmly believe that the current generic item approach to fair exchange introduces more problems to protocol design than it solves – a misleading oversimplification of a rather delicate, context-sensitive process. In this paper, we analyzed several inherent aspects of digital items and presented an interaction-oriented discussion on how the designer can take advantage of item characteristics, in order to simplify and improve the accuracy of such protocols through item-aware design.

By taking into account our remarks on the interactions between properties (Section 3), protocol designers may avoid disruptive effects that might undermine protocol goals when the items to be exchanged hold particular characteristics. These remarks address not only security aspects, but also quality of service and efficiency aspects, which might be of vital importance for real-world systems.

We illustrate the benefits of our discussion with an example of how protocol design can be made significantly more accurate by acknowledging such inherent aspects of items. The protocol produced through our item-aware design example includes mechanisms that provide more-accurate item validation and easier dispute resolution – thus providing robustness against specific, context-related issues posed by real-world scenarios, such as "buying a pig in a poke" and no-return policies.

We conclude by stating that we are currently unaware of any previous works concerning fair exchange protocol design that provide alternative models to the traditional generic protocol design and, as such, further research should be conducted on this subject. By further identifying interesting item properties and proposing specific item-related dispute processes and item validation techniques (i.e., the reversible degradation method), fair exchange researchers should be able to address several issues that are taken lightly in the current model. Therefore, we believe that abandoning the generic item-oriented model of fair exchange is, at least in the context of real-world applications, essential for future proposals intended as suitable solutions for practical scenarios [Piva and Dahab 2011, Piva and Dahab 2013].

References

Abadi, M. and Needham, R. (1996). Prudent engineering practice for cryptographic protocols. *IEEE Transactions on Software Engineering*, 22(1):6–15.

Amazon Legal Department (2005). Amazon MP3 Music Service: Terms of Use.

Asokan, A. (1998). *Fairness in Electronic Commerce*. PhD thesis, University of Waterloo.

Asokan, N., Janson, P., Steiner, M., and Waidner, M. (1997). The state of the art in electronic payment systems. *Computer*, 30(9):28–35.

Asokan, N., Shoup, V., and Waidner, M. (2000). Optimistic fair exchange of digital signatures. *IEEE Journal on Selected Areas in Communications*, 18(4):593–610.

Ateniese, G. (1999). Efficient verifiable encryption (and fair exchange) of digital signatures. In *CCS '99: Proceedings of the 6th ACM conference on Computer and communications security*, pages 138–146, New York, NY, USA. ACM.

Avoine, G. and Vaudenay, S. (2004). Optimistic fair exchange based on publicly verifiable secret sharing. In *Information Security and Privacy: 9th Australasian Conference, ACISP 2004*, volume 3108 of *Lecture Notes in Computer Science*, pages 74–85.

Bottoni, A., Dini, G., and Stabell-Kulø, T. (2007). A methodology for verification of digital items in fair exchange protocols with active trustee. *Electronic Commerce Research*, 7(2):143–164.

Garay, J. A., Jakobsson, M., and MacKenzie, P. D. (1999). Abuse-free optimistic contract signing. In *Proceedings of the 19th Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO '99, pages 449–466, London, UK, UK. Springer-Verlag.

Louridas, P. (2000). Some guidelines for non-repudiation protocols. *SIGCOMM Comput. Commun. Rev.*, 30(5):29–38.

Markowitch, O. and Kremer, S. (2001). An optimistic non-repudiation protocol with transparent trusted third party. In *Proceedings of the 4th International Conference on Information Security*, ISC '01, pages 363–378. Springer-Verlag.

Markowitch, O. and Saeednia, S. (2002). Optimistic Fair Exchange with Transparent Signature Recovery. pages 339–350.

Micali, S. (2003). Simple and fast optimistic protocols for fair electronic exchange. In *PODC '03: Proceedings of the twenty-second annual symposium on Principles of distributed computing*, pages 12–19, New York, NY, USA. ACM Press.

Nenadic, A., Zhang, N., Shi, Q., and Goble, C. (2005). DSA-Based Verifiable and Recoverable Encryption of Signatures and Its Application in Certified E-Goods Delivery. In *IEEE International Conference on e-Technology, e-Commerce and e-Service*, pages 94–99. IEEE Computer Society.

O'Mahony, D., Tewari, H., and Peirce, M. (1997). *Electronic Payment Systems*. Artech House, Inc., Norwood, MA, USA, 1st edition.

Pagnia, H. and Gärtner, F. C. (1999). On the impossibility of fair exchange without a trusted third party. Technical Report TUD-BS-1999-02, Darmstadt, Germany.

Pagnia, H., Vogt, H., and Gaertner, F. C. (2003). Fair Exchange. *The Computer Journal*, 46(1):55.

Payeras-Capellà, M., Ferrer-Gomila, J. L., and Huguet-Rotger, L. (2006). Achieving fairness and timeliness in a previous electronic contract signing protocol. In *ARES*, pages 717–722. IEEE Computer Society.

Piva, F. and Dahab, R. (2013). E-commerce of digital items and the problem of item validation: introducing the concept of reversible degradation. *Applicable Algebra in Engineering, Communication and Computing*, pages 1–32.

Piva, F. R. and Dahab, R. (2011). E-commerce and fair exchange: The problem of item validation. In *International Conference on Security and Cryptography (SECRYPT)*, Seville, Spain. SciTePress Digital Library.

Piva, F. R., Monteiro, J. R. M., and Dahab, R. (2009). Regarding timeliness in the context of fair exchange. In *Network and Service Security, 2009. N2S '09. International Conference on*, pages 1–6.

Ray, I. and Ray, I. (2000). An optimistic fair exchange e-commerce protocol with automated dispute resolution. In *EC-Web*, pages 84–93.

Ray, I. and Ray, I. (2001). An anonymous fair exchange e-commerce protocol. In *Proceedings of the 1st International Workshop on Internet Computing and E-Commerce*.

Vogt, H. (2003). Asynchronous optimistic fair exchange based on revocable items. In *Financial Cryptography*, pages 208–222.

Wang, H. and Guo, H. (2004). Fair payment protocols for e-commerce. *IFIP International Federation for Information Processing*.

Woo, T. Y. C. and Lam, S. S. (1994). A lesson on authentication protocol design. *Operating Systems Review*, 28(3):24–37.

Zhou, J., Deng, R. H., and Bao, F. (1999). Evolution of fair non-repudiation with ttp. In *ACISP '99: Proceedings of the 4th Australasian Conference on Information Security and Privacy*, pages 258–269, London, UK. Springer-Verlag.

Zhou, J., Deng, R. H., and Bao, F. (2000). Some remarks on a fair exchange protocol. In *Proceedings of the Third International Workshop on Practice and Theory in Public Key Cryptography: Public Key Cryptography*, PKC '00, pages 46–57, London, UK, UK. Springer-Verlag.

Zuo, M. and Li, J. (2005). Constructing fair-exchange p2p file market. In *Proceedings of the 4th International Conference on Grid and Cooperative Computing*, pages 941–946.

