Anonymously Establishing Digital Provenance in Reseller Chains

Ben Palmer

School of Engineering and Computer Science
Victoria University Wellington,
PO Box 600, Wellington, New Zealand 6140,
Email: Ben@ecs.vuw.ac.nz
Phone: +64 21 141 4784

Keywords: E-Commerce, Distributed Systems, Security, Zero Knowledge Proofs, Anonymity.

- Thesis Title: Anonymously Establishing Digital Provenance in Reseller Chains
- Start Date: 1/8/2008

1 Introduction

Amazon, iTunes, and domain name resellers, such as GoDaddy, only exist as on-line traders with no physical stores. While Amazon sells both physical and digital items others exclusively retail digital products and services such as digital media or more abstract products such as an access ‘right’, a license, a service, or a subscription. GoDaddy sell a subscription for a domain name, while iTunes sells a variety of digital media.

Buyers usually place trust in on-line sellers because of the reputation of the company behind the seller, for example consider the trust placed in iTunes. People trust the media they purchase from iTunes is properly licensed as they recognise Apple, it’s position in the market, and history of legitimate trading. For a relatively unknown company this may present a significant barrier to entry to a market, reducing customers choice and competition.

Currently most Internet resellers use a digital certificate to prove their identity. This certificate provides information on the physical location and contact details of the reseller so if the customer has a problem they can contact the reseller at their physical address. Certification authorities that provide digital certificates do not always perform comprehensive background checks to make sure the information provided by the reseller is factual. This can leave customers with incorrect, misleading or simply out-of-date contact information. The digital certificate itself does not provide any mechanism or guarantees though which the provenance of goods might be established.

In this paper we consider the concept of digital provenance in reseller chains. The goal of this work is to provide customers with a guarantee that the digital item they have purchased from a reseller is properly licensed. By providing confidence in the digital provenance of an item, customers can be sure of a properly licensed item even when the reseller they are dealing with is untrusted. In contrast to Digital Rights Management (DRM) we are checking the actions of an untrustworthy reseller as opposed to an untrustworthy customer.

The focus of this PhD is to develop a protocol for anonymously establishing provenance for digital items and provide a detailed analysis of both the security properties and performance of the protocol. We have developed a protocol called the Tagged Transactions Protocol which uses ‘tags’ to establish provenance. If a reseller can provide a customer with a valid tag, the customer can have confidence that the reseller has sold them a properly licensed item. The tagged transaction protocol uses a Tag Generation Centre (TGC) to sign and check tags. A valid tag must be signed by the TGC and the customer can check the tag passed to it by the reseller was signed by the TGC.

The main features of the Tagged Transaction Protocol are:
1. Provides a method for customers to check they are purchasing a legitimate item, even from an untrusted reseller.
2. Provides anonymity for customers, resellers, and optional anonymity for suppliers.
3. Has mechanisms to verify the actions of the third party, the TGC, in the protocol so the TGC is not acting as a trusted third party.
4. Allows customer reselling of items. If a customer wishes to resell an item they have purchased from a reseller, they can take the role of a reseller and sell it to another customer.

2 Related Work

The Paradiso system lets customer purchase not only the songs and videos from content providers but also reseller rights (Nair et al. 2005, 2008). To prevent malicious behaviour, a Trusted Computing Module (TCM) is used to store encryption keys and to perform private key operations in secure memory. Once a reseller has bought the rights to redistribute a certain media file a certain number of times they can resell this media file without needing to interact with the supplier. The main drawback of the Paradiso system is the reliance on a TCM.

The IEEE working group P1817 has produced a document suggesting a standard which is similar to the Paradiso system where customers can resell items they have purchased as they can with physical products (P1817 2010). While the P1817 standard provides options for customers to resell content it relies on a trusted player to store cryptographic keys. The working group is still in its early stages and it will be interesting to see how this develops in the future.

Serban, Chen, Zhang, and Minsky introduce the concept of a decentralised electronic marketplace (DEM) where transactions are subject to a set of trading rules (Serban et al. 2008) implemented using a mechanism called Law Governed Interaction (LGI). In LGI, a law is formulated using an event-condition-action pattern. Apart from the agents taking part in transactions in the marketplace, there are also a set
of trusted controllers that enforce the law of the marketplace. Every agent is assigned a controller, and all messages for the marketplace are sent through the agent’s controller. Although these controllers are distributed, there is no method to verify the actions of the controllers.

Durfee and Franklin have designed and implemented a protocol using contracts for distribution chain security (Durfee & Franklin 2000). A contract is created for a digital work by the content provider or author. The contract is then sent to a trusted contract certifier who checks the contract is well formed and signs it with a digital signature. The main difference between our work and the work by Franklin and Durfee is that they concentrate on ensuring the integrity of the license in a reseller chain where as we concentrate on ensuring the license is legitimate. There work could be used in conjunction with our work on digital provenance to provide a method for resellers to alter the terms in the license they sell and still prove to the customer that the license is legitimate and consistent with the original license.

The Idemix system (Camenisch & Van Herreweghen 2002) developed by Camenisch and Van Herreweghen is an implementation of an anonymous credential system (Camenisch & Lysyanskaya 2001). Both the Tagged Transaction protocol and one-show anonymous credentials provide replay detection but there are several differences. The replay detection in anonymous credentials, when using multiple verifiers, is performed after the fact whereas in the tagged transactions protocol, replay detection is done live by the TGC. In the Tagged Transaction protocol a reseller could transfer a tag from themselves to another reseller by giving them the one time secret key used for the tag $sk_{tag,r}$ and commitment value $z$ where $a_{tag,r} = g^z \mod p$. When using anonymous credentials there is a strong incentive to prevent the transfer of credentials as this allows credential pooling. The tagged transaction protocol also provides optional supplier anonymity which is not supported by anonymous credentials. Finally, in the tagged transaction protocol the supplier is not involved in the protocol after the generation of the tag. When using anonymous credentials, the supplier would act as the issuer of the credential and would be involved in every transaction.

3 Problem Domain

The basic reseller model is shown in Figure 1. There may be several resellers between the supplier and the customer. We refer to the list of suppliers and resellers to the customer as a chain and each individual section as a link. The three roles involved are:

1. The Supplier: Suppliers are the original creators or holders of the rights for an item.
2. The Resellers: A reseller or middleman has a set of customers who purchase items from them as well as a set of suppliers that the reseller purchases items from. There may be multiple resellers between the supplier and the customer.
3. The Customer: Any party that is interested in purchasing an item produced by a supplier and sold by a reseller.

3.1 Threat Model

A malicious reseller has several ways to try and defraud both the supplier and the customer. We have informally grouped these actions in to the following three categories:

1. Impersonation. The reseller claims to be the supplier or tries to subvert the protocol to make it appear that they are the supplier. In this way the reseller may be able to take full payment for an item without ever paying their supplier while the customer believes they have a properly licensed item.
2. Counterfeiting. The reseller sells the customer an item but never buys it from the supplier. There are several ways a reseller may try and fabricate a license for an item including:
   - Fabrication. The reseller tries to fabricate a license for an item from scratch (or possibly having seen the structure of other licenses).
   - Replay. The reseller tries to sell a license they have purchased from the supplier to multiple customers.
   - Network Sniffing. The reseller sees a legitimate license being sent over the network, copies it, and sends it to a customer.
3. Information Revelation. The customer learns the identity of the supplier or one of the resellers that is not its neighbour in the chain.

While the customer will want to verify that their reseller has not committed any of the first two actions detailed above, there are also issues with identity revelation. The reseller may want the identity of the supplier and any resellers that are before it in the chain to be anonymous. If the customer knew the identity of the supplier, or a reseller that has fewer links between it and the supplier, the customer may try and go straight to the supplier effectively avoiding the ‘middleman’ that is the reseller.

4 Methodology

<table>
<thead>
<tr>
<th>Task</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Design of Protocol</td>
<td>Completed</td>
</tr>
<tr>
<td>Initial Security Analysis</td>
<td>Completed</td>
</tr>
<tr>
<td>Modelling of Protocol</td>
<td>December 2010</td>
</tr>
<tr>
<td>Theorem Proving</td>
<td>February 2011</td>
</tr>
<tr>
<td>Extensions of Protocol</td>
<td>April 2011</td>
</tr>
<tr>
<td>Thesis First Draft</td>
<td>June 2011</td>
</tr>
<tr>
<td>Completion Date</td>
<td>August 2011</td>
</tr>
</tbody>
</table>

Table 1: Methodology

The methodology for this thesis is shown in Table 1. The first two steps were the design and initial security analysis of the Tagged Transaction Protocol which are now complete. The third step in our methodology is to model the Tagged Transaction Protocol using a model checker. The FDR (Formal Systems (Europe) Ltd n.d., Roscoe 1994) model checker is able to check a CSP (Hoare 1978) model of a protocol against a CSP specification. If FDR finds any situations where the model is able to get in to a state that is not allowed by the specification, a counter example is output to allow the debugging of the protocol. By implementing a CSP model of the protocol as well as CSP specifications of our security requirements we can check the Tagged Transaction protocol
to see if there is any way the protocol can get in to an invalid state where a private key is leaked, or a reseller is able to produce tags that are accepted by the customer. To make the modelling of a protocol in CSP easier, Gavin Lowe has built the Casper (Lowe 1998) compiler that takes a protocol description and compiles it to a CSP file that can be checked by FDR. While model checking is a useful tool that is able to show problems in security protocols, the models constructed have to be finite and rather small. Ideally we would like to be able to prove our protocol secure with respect to our security requirements no matter how many parallel instances of the protocol are running at the same time. The fourth step of our methodology is to apply theorem proving to extend the specific results achieved using model checking to a more general result. We also intend to look at other application areas where the Tagged Transaction protocol can be applied to solve other existing problems. The final tasks are an initial draft of the thesis followed by the final version of the thesis to be submitted in August 2011.

5 Initial Results

Currently we have designed the Tagged Transaction protocol to fulfil our security requirements. We have also completed a basic security analysis of the protocol and are in the process of completing the model checking phase of the methodology.

5.1 Protocol Design

The tagged transaction protocol provides a mechanism for offline establishment of the provenance of a digital item with replay detection while preserving the anonymity of suppliers and resellers. We use a third party that we call a Tag Generation Centre (TGC) to generate and sign tags. We call the tagged transaction protocol offline as, once the supplier has generated a tag and sent it to the first reseller in the chain, the supplier is not involved in the checking of a tag or the generation of tags for further resellers. Replay detection provides a method to prove if a reseller has tried to replay a tag that has already been used. The witness extraction property of zero knowledge proofs of knowledge is used to detect replay of a tag. The tagged transaction protocol makes use of an anonymous communication channel between the TGC and suppliers and resellers to provide supplier and reseller anonymity. When a customer or reseller purchases an item from a reseller or the supplier they check it has a valid tag signed by the TGC.

![Diagram of the Tagged Transaction Protocol](image)

Figure 2 shows the basic steps in the tagged transaction protocol. First the supplier will register a new item $x$ with the TGC and provides a public key for the item $pk_x$ (1). If the supplier does not require anonymity, this check can be done by a human to provide a stronger check than the one at first serve registration that is used with anonymous suppliers. The supplier can then create tags for this item. A reseller will send a purchase request to the supplier for item $x$ signed with the secret key for the item $sk_x$ (2). The supplier will then request the TGC to generate a tag for the item which contains a license signed by the supplier, a one time public key for the tag generated by the reseller $pk_{tag}$ and a commitment value for a zero knowledge proof $a_{tag}$ (3). The TGC then returns the tag to the supplier (4) who sends it to the reseller (5). When a second reseller sends a purchase request for the item to the first reseller (6) the reseller requests a new tag from the TGC and the TGC and reseller take part in a zero knowledge proof of knowledge that the reseller knows $sk_{tag}$ using the commitment value $a_{tag}$ (7). If the reseller has tried to replay the tag and the TGC has used different challenge values in the zero knowledge proof, the TGC can extract the secret key for the tag $sk_{tag}$ and prove that the reseller has tried to replay a tag. The TGC then sends this new tag $tag_2$ to the first reseller (8) who sends it to the second reseller (9). Steps (6) - (9) can be repeated depending on how many resellers are in the chain between the supplier and the customer. The customer will then send a purchase request for $x$ to the last reseller in the chain which contains a public key for the tag $pk_{tag_2}$ and a commitment for a zero knowledge proof $a_{tag2}$ (10). The reseller then requests a new tag from the TGC (11). The TGC and reseller take part in a zero knowledge proof of knowledge that the reseller knows $sk_{tag2}$ using the commitment value $a_{tag2}$. The TGC returns the new tag $tag_3$ to the reseller (12) who sends it to the customer (13). When a customer or reseller wishes to verify the tag they have received from a reseller, they can check that the tag has been signed by the TGC. This shows that the tag has not been replayed as the TGC will not generate and sign a new tag if it detects replay. The verifier can then also check that the license contained in the tag has been signed by the supplier for the item.

5.2 Security Analysis

We now consider how the tagged transaction protocol fulfills our security goals. Specifically we examine if the protocol prevents impersonation and counterfeiting, and preserves the anonymity of the parties involved.

5.2.1 Impersonation

To generate a new tag for the item $x$ and pass itself off as the supplier of $x$ a malicious reseller has to provide the TGC with a request for a new tag signed by the secret key $sk_x$ of the item. If a reseller has a new tag that was not created by the supplier but is signed by the secret key for the item the reseller must have either discovered the secret key of the item $sk_x$, or altered a legitimate tag request.

To prevent a reseller from altering a legitimate tag the digital signature scheme used must be secure against adaptive chosen message attacks which is why we use the modified El Gamal signature scheme (Pointcheval & Stern 1996).

If a malicious entity was able to intercept and modify messages from the supplier to the TGC, they could intercept the registration message that contains the identity of the item $id = H(x)$ and the public key for the item $pk_x$. They could then modify the public key of the item to a public key that they know the private key for and forward this message to the TGC. To prevent this attack the registration message from the
supplier to the TGC is encrypted using the public key of the TGC to prevent a malicious entity in control of the network from being able to modify registration messages.

5.2.2 Counterfeiting

If a reseller never purchases the item from the supplier, they will not have a tag that is signed by the TGC for the item. To have a tag signed by the TGC the reseller must either have purchased the item form the supplier, or they must know the secret key of the TGC. If the secret key of the TGC \( sk_{TGC} \) became known, then any reseller could generate their own tags.

If a reseller tries to replay a tag that has already been used with high probability the TGC will detect the replay and not generate a new tag. The reseller will only be able to replay a tag if two customers choose the same one time public key for the tag, or if the TGC chooses the same challenge twice for the same tag. With a sufficiently large key space the chance of two customers choosing the same key is negligible. As the TGC saves the information on the tag and the challenge and response used in the zero knowledge proof, the TGC can make sure it does not choose the same challenge twice for the same tag.

There is a possibility that a reseller \( r_1 \) can observe a tag that was issued to another reseller \( r_2 \) that has not been used. The reseller \( r_1 \) can then try and use the tag before reseller \( r_2 \) generates a new tag from it. To be able to generate a new tag, the reseller \( r_1 \) has to prove knowledge of the one time private key \( sk_{tag,r_2} \) that was used when the tag was generated.

5.2.3 Anonymity

The anonymity of suppliers and resellers is provided by an anonymous communication channel used in the communication to the TGC (such as TOR (Dingledine et al. 2004)). Anonymity could also be provided by using a separate anonymity service (or the TGC as an anonymity service) that all communication would go through. This would prevent the leaking of the identities of the parties involved in transactions but still allow discovery of the identities in case of legal disputes.

5.3 Modelling Results

We are currently in the process of completing the modelling of the Tagged Transaction protocol in CSP using the Casper compiler. While the use of the Casper compiler can quickly and easily generate CSP files for checking using FDR, we have found the security properties that can be expressed in Casper to be quite restrictive. In our protocol the property of non-repudiation is an important property to check where Casper can only express secrecy and authentication properties. For this reason we have have created an initial model with Casper and then expanded on it by editing the CSP file directly.

We currently have two CSP models of the protocol. The first models the generation of tags by the supplier and TGC which are then sent to the customer via the reseller. This model checks for the secrecy of the private keys for the item and of the private key of the TGC. It also checks the property of impersonation where there should be no way for a customer to accept a tag that has not been initially generated by the supplier. The second model checks the property of replay where there should be no way for two customers to both accept tags where the reseller has not generated both tags from the TGC correctly.

6 Conclusions

In this brief paper we have presented the research work being done for my thesis on anonymously establishing digital provenance in reseller chains. While this is an interesting and important area, little research has been done to create protocols to achieve the security requirements we have defined. We have outlined a methodology to design and provide an in depth analysis of the security properties of the Tagged Transaction protocol.

References


