Formal Analysis of Comparison-Based Non-Malleability for Commitments

Ekaterina Zhuchko[13], Denis Firsov[12], and Sven Laur[3]

[1] Tallinn University of Technology, Tallinn, Estonia
ekzhuc@ttu.ee
[2] Guardtime, Tallinn, Estonia
denis.firsov@guardtime.com
[3] Tartu University
swen@math.ut.ee

1 Introduction

The field of cryptography has rapidly grown in its complexity and subsequently faced a crisis in producing correct proofs. The security guarantees for cryptographic protocols usually come in the form of pen-and-paper proofs. Formalising the intuition behind cryptographic security proofs is not a straightforward task as there are a lot of hidden assumptions and informal reasoning that can be easily overlooked by the reader. The problem of inadequate definitions in cryptography is not a new one [3]. The errors in definitions may take many years to be discovered and the impact of these errors can range from a minimal nuisance to an actual threat that can be realised as an attack in the real world. In this work, we analysed non-malleable commitment schemes and formalised our results in EasyCrypt. EasyCrypt is a proof assistant that was created to assist with the verification of security proofs for cryptographic protocols.

A commitment scheme is one of the fundamental primitives in cryptography which usually involves two parties: sender and receiver. Intuitively, we can think of the commitment as a locked box containing a message. Only the sender knows the secret key in order to unlock it and see the message. The sender can send this box to the receiver and then at a later stage give him the secret key to unlock it. A commitment scheme is a triple of algorithms $(\mathsf{KGen}, \mathsf{Commit}, \mathsf{Verify})$ and consists of two phases of interaction. The commit phase is when the sender commits to a message and sends it to receiver. The reveal stage is when the receiver can access the message and check its authenticity. The commitment scheme should also provide a number of security properties such as hiding (unable to see the message without the secret key) and binding (once the message is commited and sent to the receiver, the sender cannot change it).

The security property that we are interested in is non-malleability. It was introduced by Dolev, Dwork and Naor in 1991 [2]. The notion of non-malleability can be applied to different cryptographic primitives such as encryption, digital signatures or commitments. One important example of non-malleable commitments is its application in time-stamping [1]. The intuition behind non-malleability is that we are now in the presence of an attacker, who can attempt to maul the commitment in the protocol and then forward it to the receiver. One common motivating example where non-malleability would be needed is that of secure auction bidding. If the adversary could modify other bidders' commitments by always offering a deal that is one dollar higher i.e. $(bid+1)$, then the adversary would have an unfair advantage without needing to learn the exact amounts that others have placed.

Cryptographic proofs are often expressed in the form of games where each game is a program. These games embed a security notion within them. One way of formalising non-malleability is through the following security games [4], where $\mathcal{A}$ is any efficient adversary attacking the commitment scheme:

```
$\mathrm{GN}_0^{\mathcal{A}}$
$\mathsf{pk} \leftarrow \mathsf{KGen}$
$\mathcal{M} \leftarrow \mathcal{A}.\mathsf{init}(\mathsf{pk})$
$m \leftarrow\$ \mathcal{M}$
$(c,d) \leftarrow \mathsf{Com}_{\mathsf{pk}}(m)$
$(n, R(\cdot), \hat{c}_1, ..., \hat{c}_n) \leftarrow \mathcal{A}.\mathsf{commit}(c)$
$((\hat{m}_1, \hat{d}_1), ..., (\hat{m}_n, \hat{d}_n)) \leftarrow \mathcal{A}.\mathsf{decommit}(d)$
$\mathsf{if}\ c \in \{\hat{c}_1, ..., \hat{c}_n\}$ return $0$
$b \leftarrow \mathsf{Verify}(pk, \hat{m}_i, \hat{c}_i, \hat{d}_i)\ \forall i \in \{1 \leq i \leq n\}$
return $R(m, \hat{m}_1, ..., \hat{m}_n)$
```

```
$\mathrm{GN}_1^{\mathcal{A}}$
$\mathsf{pk} \leftarrow \mathsf{KGen}$
$\mathcal{M} \leftarrow \mathcal{A}.\mathsf{init}(\mathsf{pk})$
$m \leftarrow\$ \mathcal{M}; \bar{m} \leftarrow\$ \mathcal{M}$
$(\bar{c}, \bar{d}) \leftarrow \mathsf{Com}_{\mathsf{pk}}(\bar{m})$
$(n, R(\cdot), \hat{c}_1, ..., \hat{c}_n) \leftarrow \mathcal{A}.\mathsf{commit}(\bar{c})$
$((\hat{m}_1, \hat{d}_1), ..., (\hat{m}_n, \hat{d}_n)) \leftarrow \mathcal{A}.\mathsf{decommit}(\bar{d})$
$\mathsf{if}\ \bar{c} \in \{\hat{c}_1, ..., \hat{c}_n\}$ return $0$
$b \leftarrow \mathsf{Verify}(pk, \hat{m}_i, \hat{c}_i, \hat{d}_i)\ \forall i \in \{1 \leq i \leq n\}$
return $R(m, \hat{m}_1, ..., \hat{m}_n)$
```

In [4], the authors define the scheme to be non-malleable if the probability of distinguishing these two security games is negligible. These two games are set so that the goal of the adversary is to come up with an appropriate relation in order to win the game. This would demonstrate the adversary's ability to change the structure of a committed message and hence the adversary would be able to distinguish between the two games. In other words, the adversary would have modified a commitment and the receiver opened it, with the resulting message having no resemblance to the original committed message.

1.1 Counter example

We can build adversary $\mathcal{A}$ which breaks the non-malleability definition. For the message space we fix a uniform Boolean distribution and we fix $n = 1$. The commit and decommit functions are defined in the following way:

```
$\mathcal{A}.\mathsf{init}(\mathsf{pk})$
$\mathcal{M} \leftarrow \{\frac{1}{2}\mathsf{true}, \frac{1}{2}\mathsf{false}\}$
return $\mathcal{M}$
```

```
$\mathcal{A}.\mathsf{commit}(\mathsf{c})$
$(\hat{c}, \hat{d}) \leftarrow \mathsf{Com}_{\mathsf{pk}}(\mathsf{false})$
$R \leftarrow \lambda m_0 m_1.(m_0 = \mathsf{false}) \wedge (m_1 = \mathsf{false})$
return $(1, R, \hat{c})$
```

```
$\mathcal{A}.\mathsf{decommit}(\mathsf{d})$
if $\mathsf{Verify}(pk, \mathsf{false}, c, d)$
    return $(\mathsf{false}, \hat{d})$
else $\mathsf{fail}$
```

Let $\mathcal{A}$ be an adversary against the non-malleability games stated in the previous section. A commitment scheme C is non-malleable if for all adversaries $\mathcal{A}$, the advantage $\mathsf{Adv}_C^{\mathrm{nmo}}(\mathcal{A}) = |\Pr[GN_0^{\mathcal{A}} = 1] - \Pr[GN_1^{\mathcal{A}} = 1]|$ is negligible. When the adversary fails, he simply aborts the game since at this point he has all the relevant information to know not to continue the game. If we inline $\mathcal{A}$ into the games $GN_0^{\mathcal{A}}$ and $GN_1^{\mathcal{A}}$, we get the following cases:

- $\mathsf{Pr}[GN_0^{\mathcal{A}} : m = \mathsf{false} \wedge \mathsf{win}] = \frac{1}{2}$.
- $\mathsf{Pr}[GN_0^{\mathcal{A}} : m = \mathsf{true} \wedge \mathsf{win}] = 0$ as the relation R will not hold and the verification will also fail.
- $\mathsf{Pr}[GN_1^{\mathcal{A}} : m = \mathsf{false} \wedge \bar{m} = \mathsf{false} \wedge \mathsf{win}] = \frac{1}{4}$.
- $\mathsf{Pr}[GN_1^{\mathcal{A}} : m = \mathsf{false} \wedge \bar{m} = \mathsf{true} \wedge \mathsf{win}] = 0$ as the relation R will not hold.
- $\mathsf{Pr}[GN_1^{\mathcal{A}} : m = \mathsf{true} \wedge \bar{m} = \mathsf{false} \wedge \mathsf{win}] = 0$ as the verification will fail.
- $\mathsf{Pr}[GN_1^{\mathcal{A}} : m = \mathsf{true} \wedge \bar{m} = \mathsf{true} \wedge \mathsf{win}] = 0$ as the relation R will not hold and the verification will also fail.

In order to calculate the adversary's winning advantage, the difference of these two probabilities is taken and it is non-negligible:

$$\mathsf{Adv}_{C}^{\mathrm{nmo}}(\mathcal{A}) = |\Pr\left[GN_0^{\mathcal{A}} = 1\right] - \Pr\left[GN_1^{\mathcal{A}} = 1\right]| = \frac{1}{4}.$$

2 Results

In this work we analysed comparison-based non-malleable commitment schemes [4]. We formalised our results in EasyCrypt, a proof assistant that was created to assist with the verification of security proofs for cryptographic protocols. We prove that this definition is not satisfiable through a counter example and thus cannot be instantiated with any concrete commitment scheme. For future work, we are interested to find out whether there is a way to fix the comparison-based definition for non-malleable commitments.

3 Acknowledgments

This work was partially supported by the ESF-funded Estonian IT Academy research measure (project 2014-2020.4.05.19-0001).

References

[1] Ahto Buldas and Sven Laur, *Knowledge-binding commitments with applications in time-stamping*, Public Key Cryptography – PKC 2007 (Berlin, Heidelberg) (Tatsuaki Okamoto and Xiaoyun Wang, eds.), Springer Berlin Heidelberg, 2007, pp. 150–165.

[2] Danny Dolev, Cynthia Dwork, and Moni Naor, *Nonmalleable cryptography*, SIAM Review **45** (2003), no. 4, 727–784.

[3] Neal Koblitz and Alfred Menezes, *Critical perspectives on provable security: Fifteen years of "another look" papers*, Advances in Mathematics of Communications **13** (2019), 517–558.

[4] Sven Laur and Kaisa Nyberg, *Efficient mutual data authentication using manually authenticated strings*, International Conference on Cryptology and Network Security, Springer, 2006, pp. 90–107.