Formal Security Analysis of NFC M-coupon Protocols using Casper/FDR

Ali Alshehri, Johann A. Briffa, Steve Schneider and Stephan Wesemeyer
Dept. of Computing, University of Surrey
Guildford GU2 7XH, England
Email: A.A.Alshehri@surrey.ac.uk, J.Briffa@surrey.ac.uk, S.Schneider@surrey.ac.uk, S.Wesemeyer@surrey.ac.uk

***Abstract*—Near field communication (NFC) is a standard-based, radio frequency (RF), wireless communication technology that allows data to be exchanged between devices that are less than 10 cm apart. NFC security protocols require formal security analysis before massive adoptions, in order to check whether these protocols meet its requirements and goals. In this paper we formally analyse NFC-based mobile coupon protocols using formal methods (Casper/FDR). We find an attack against the advanced protocol, and then we provide a solution that addresses the vulnerability formally.**

***Index Terms*—NFC, M-coupon, Casper, FDR, formal methods, model checking**

I. Introduction

Near Field Communication (NFC) is a technology that enables people to make payments, for example at the supermarket or the train station, just by waving their mobile at the point of sale. NFC is a standard-based radio frequency (RF) communication link technology that can be embedded into any device (computers, mobiles, PDA, TV, printers, etc.), in order to allow data to be exchanged between devices that are less than 10 cm apart [1]. NFC tends to be in mobile phones more since the majority of people already have one. NFC in mobiles can operate in three different modes determined by the application used; it can communicate with other NFC mobiles in *Peer-to-Peer mode*, or communicate with a passive RFID/NFC tag in *reader/writer mode*, or communicate with an NFC reader in *card emulation mode* [2].

The requirement for robust security in NFC in general has been emphasised in the literature [3], [4]. NFC-SEC standards [5], [6] enable two NFC devices, in Peer-to-Peer mode, to establish a secure channel. However, they do not provide entity authentication, and are not suitable for applications requiring specific security mechanisms. For any NFC application, cryptography is the ideal measure to address security requirements, such as confidentiality, integrity and availability. The NFC mobile coupon application (M-coupon) is one of the promising and popular applications [7]–[10]. The NFC M-coupon system requires secure issuing and cashing of the M-coupons, otherwise it can cause huge losses for a company [11], and damage to its reputation. Dominikus and Aigner [12] introduced NFC M-coupon protocols which allow secure issuing and cashing of electronic coupons.

On the other hand, formal security analysis has not been carried out in the NFC domain in general. Such analysis is important because implementing strong cryptographic algorithms in NFC schemes is only half of the solution. In fact, the way encryption is used between entities is the more challenging part of protocol design. It is quite difficult to establish secure cryptographic protocols even with robust cryptographic algorithms. Many attacks can be realised during the execution of the cryptographic protocols just by intercepting and replaying encrypted messages between entities, without decrypting any messages [13]. Thus, formal security analysis of NFC protocols in general, and especially the NFC M-coupon protocol, is critically important before their widespread adoption.

In our analysis we use *Communicating Sequential Processes* (CSP) [14], with its model checker *Failures Divergence Refinement* (FDR), which is proven to be an effective method in analysing the security of protocols [15]. However, modelling protocols in CSP is not a trivial task. Lowe developed *Casper* [16], a tool that allows the user to write an abstract description of a security protocol, which the tool compiles to CSP code for direct checking with the model-checker FDR2. Casper has been used to analyse many protocols [17], which confirms its capability to find vulnerabilities.

In this paper we use Casper to formally analyse the NFC M-coupon protocols proposed by Dominikus and Aigner [12]. We found an attack against the advanced protocol. We then we propose a solution to address this vulnerability. We use the same approach to formally verify that the corrected version does not suffer from the same flaw.

II. The NFC mobile coupon protocols

The NFC M-coupon system has a typical scenario, as illustrated in Figure 1. Initially, all parties will have NFC capability, in order to communicate with each other. Firstly, a user brings his NFC mobile close to an NFC issuer (smart poster, newspaper). Then an M-coupon is issued and sent to his mobile. Later, the user goes to the shop for cashing in the M-coupon to the cashier. The cashier may authenticate the user before the cashier provides the promised bonus. Only the cashier needs to have online access, whereas the issuer and the user can be offline.

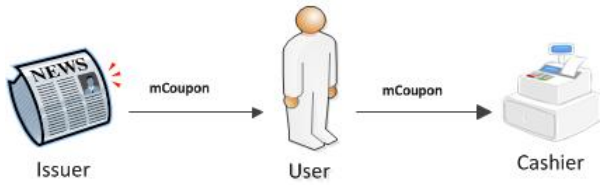


Figure 1. General NFC mobile coupon

ID(i)	Issuer ID.
ID(u)	User ID.
ID(c)	Cashier ID.
Offer	Data about the Offer .
EK	Shared key between Issuer(s) and Cashier(s)
Nu / Nu2	User's nonce (random number).
Ni	Issuer's nonce.
Nc	Cashier1's nonce.
Nc'	Cashier2's nonce.
SigU	Signature of user's Mobile.
SigC	Signature of Cashier.

Table I
PROTOCOL NOTATION

A. NFC M-coupon security requirements

Dominikus and Aigner [12] stated four security requirements for NFC M-coupon protocols:

- **No Multiple Cash-in**: An attacker shall not be able to use the same M-coupon multiple times.
- **No Unauthorized Generation**: An attacker shall not be able to issue his own M-Coupons.
- **No Manipulation**: M-Coupons shall not stay valid after a manipulation.
- **No Unauthorized Copying**: An attacker shall not be able to produce a valid copy of an M-coupon and cash it in.

Depending on the M-coupon system, the properties *No Multiple Cash-in* and *No Unauthorized Copying* can be optional.

In [12], two protocols were presented: a simple and an advanced M-coupon protocols. The advanced protocol addresses all four requirements. The simple protocol addresses the same requirements except for *No Unauthorized Copying* of the M-coupon.

Different approaches were used to address these requirements. Firstly, *No Multiple Cash-in* is addressed by establishing data bases for all M-coupons used at all cashiers. All new M-coupons can be checked online before cashing in, in order to prevent multiple cash-ins. There is no need for encryption countermeasures to meet this requirement. Secondly, *No Unauthorized Generation* is addressed by establishing long term shared keys between cashiers and issuers (symmetric authentication). Thirdly, *No Manipulation* is addressed by relying on the secrecy of the long term shared keys between cashiers and issuers: any change in the M-coupon would be detected. Finally, *No Unauthorized Copying* is addressed by embedding a user's signature inside the M-coupon.

No Multiple Cash-in is typically called protection against **Double-Spending**. Unauthorized generation and manipulation are both concerned with **Forgery Protection**. In order to be more precise, the *No Unauthorized Copying* requirement can be analysed as authentication of the M-coupon holder: **User Authentication**.

B. Protocol descriptions

The simple and the advanced protocols are shown in Figure 2, designed according to the ISO authentication standard [18]. We use the notation of Table I in describing the protocols.

The Simple M-coupon protocol

The aim of this protocol is to provide genuine M-coupons from genuine issuers. It allows users to copy their M-coupons for their family and friends. The protocol is given in Figure 2a.

Issuing phase: after the user brings his mobile close to the issuer, his mobile sends a nonce (new random number) Nu (message 1). Then, the Issuer sends the M-coupon to the user (message 2). The M-coupon contains: the issuer identity, the user's random number, the promised offer and an encrypted part. The encrypted part also contains the offer and the user's random number.

Cashing phase: the user can cash the M-coupon at any cashier has a relationship with the issuer. The user brings his mobile near the cashier and sends the M-coupon (message 3). The cashier checks three things: firstly, the validity of the M-coupon i.e. the M-coupon has not been used before (if required by the system). Secondly, the encrypted part is equal to the unencrypted part. Thirdly, the encryption key has a genuine issuer. If all these conditions are satisfied, then the bonus is given to the user (message 4).

The Advanced M-coupon protocol

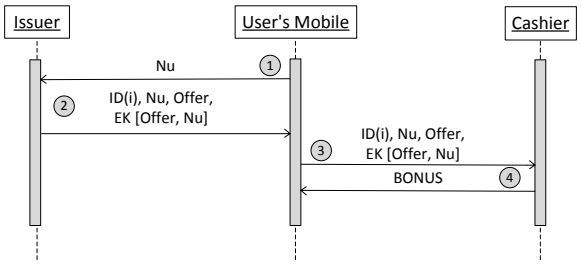
The protocol works in two phases, as shown in Figure 2b.

Issuing phase: In the advanced protocol two messages are added. After the user sends a random number (message 1), the issuer sends a random number as well (message 2), and asks the user to sign it (message 3). The issuer would not be able to verify the signature, but the signature will be included inside the encrypted part of the M-coupon (message 4). Later, the cashier will verify the signature to make sure no one can cash the M-coupon except the user who has signed it at the issuing phase.

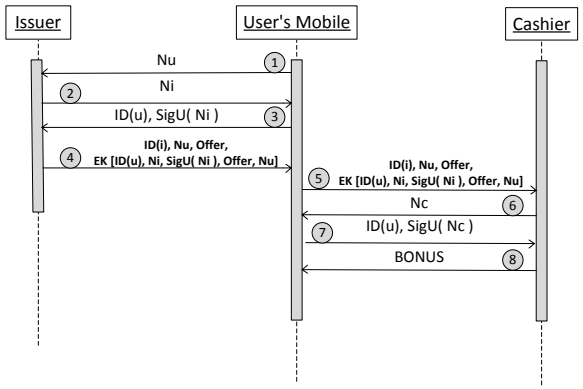
Cashing phase: The cashier verifies user's signature included inside the M-coupon (message 5), which was signed at the issuing phase. Then, a further user authentication protocol is performed in order to ensure that this is the same user. So, the cashier sends a random number (message 6). Then the user signs it (message 7) and sends it back to the cashier. If both signatures (messages 5 and 7) are confirmed, then the bonus is given to the user (message 8).

III. METHOD

Casper is a formal tool which analyses security protocols at the formal or symbolic level. Casper uses the Dolev-Yao attacker model [19]. Here the attacker has full control of the



(a) the simple M-coupon protocol



(b) the advanced M-coupon protocol

Figure 2. The simple and the advanced M-coupon protocols

network traffic, and is able to block, replay, redirect, spoof and duplicate messages. However, the attacker can only encrypt or decrypt if it holds the appropriate key, commonly known as the *perfect encryption* assumption. It tries to break the security protocol with these capabilities, using what has been provided by the participants executing their part of the protocol.

In order to model a protocol in Casper, two main sections need to be described. The first section is the protocol definition which is a general description of the protocol in terms of the participants and how they create and respond to messages. The second section is the system definition which gives the specific system runs of the protocol to be modelled, i.e. how many issuers, users and cashiers should be modelled in the analysis. Casper, or CSP, requires a specific system to be checked within the protocol, to manage the size of the state to be explored. The more simultaneous runs to be considered, the more states to explore and the longer it is to analyse. The state space grows exponentially with the number of runs, and exploration becomes infeasable quite quickly; this is known as the *state explosion problem*.

- In the simple protocol, we were able to check models containing up to two issuers, two users and two cashiers.
- In the advanced protocol we checked two models: firstly the whole protocol with one issuer, one user and one cashier due to its complexity; secondly the cashing phase with two issuers, two users and two cashiers.
  The reason we model the cashing phase is to have a more realistic model, and to have a better understanding of attacks as we will explain later.

Then we model the requirements of the protocols:

- We model **forgery protection**, which addresses the unauthorized generation and the manipulation of the M-coupon. We can analyse this with two checks; firstly, we examine the secrecy of the shared key between issuer(s) and cashier(s). Secondly, we examine the authentication between them.
- In addition, **Double-spending**, multiple cash-in, is the next requirement that needs to be checked. Multiple runs of the protocol (double-spending) are not managed by the protocol itself but by a database at the cashier, which knows all M-coupons used so far. It is an assumption built into our model that the cashiers have such a database and use it correctly. We are concerned with analysing for attacks on the protocol itself, and not attacks on the use of the database.
- Finally, **user authentication** is the last requirement, which is considered only for the advanced protocol. We can consider this by examining whether the user is authenticated to the cashier properly.

Moreover, the analysis also requires us to define the initial knowledge of the intruder. The intruder knows the following: the identities of the issuer, the user and the cashier, and the offer (since it is sent in clear). In addition the intruder has his own public/private keys and nonces, so can act as a legitimate user.

IV. Results and discussion

With respect to **forgery protection**, FDR2 did not find any attacks on either the simple or the advanced M-coupon protocols.

However, an attack was found when we examine the protocol with respect to **user authentication** in the advanced protocol. We categorise the attacks into two kinds: honest agents attack and dishonest agents attack.

A. Honest agents attack

This attack was identified when analysing the whole advanced protocol model (the first model for analysing the advanced protocol). Here, all agents behave honestly and thus play their role correctly. Figure 3 is the attack trace provided by Casper, shown in sequence diagram form. Messages from 1 to 8 show an attack on the authentication between the user and the NFC issuer at the M-coupon issuing phase. This is because neither the issuer nor the user is able to verify who they received the message from.

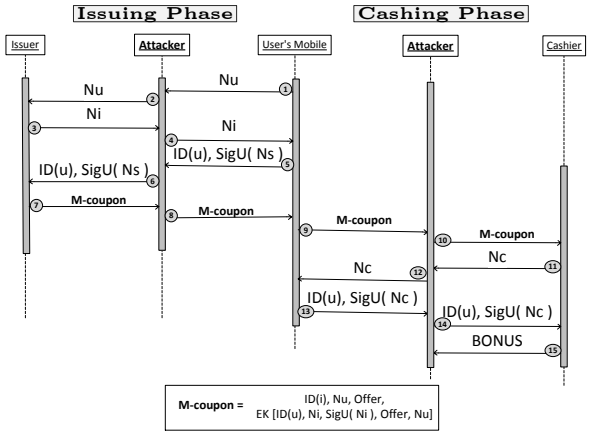


Figure 3. NFC mCoupon Attack Trace

Messages from 9 to 15 in Figure 3 is an attack on the authentication between the user and the cashier, which is the main goal of the advanced protocol.

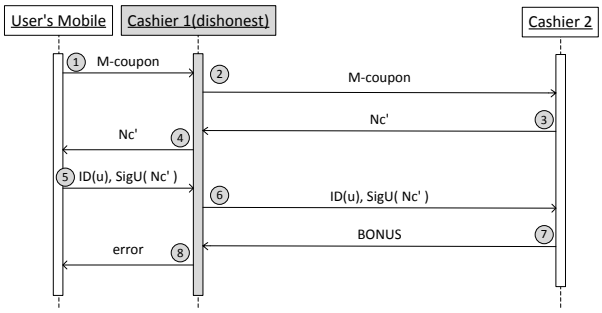
The attack works as follows: in the original protocol (Figure 2b), the cashier verifies the first signature at message 5, then the cashier needs to perform another challenge-response protocol, consisting of messages from 6 to 8, in order to prevent a replay attack. However, the signature does not show to whom this signature is going to: the identity of the verifier (Cashier) is missed. This vulnerability allows an intruder to intercept and cash a genuine M-coupon, as shown in messages from 9 to 15 in Figure 3. An intruder is able to acquire an M-coupon signed by the original customer and subsequently cash it. This makes the cashier believe he has completed a run of the protocol with the user. Later when the real user comes to cash his M-coupon, the cashier will not accept it because he thinks the user is using the M-coupon twice.

The attack in reality is relying on the possibility of the intruder performing a combination of skimming, eavesdropping and relay attacks [20], [21]. However, the relay attack is a general attack against NFC technology [21]. Therefore, if an intruder was able to relay messages between entities to exploit this vulnerability, the attack would work in any case even with a fixed secure protocol. There are some techniques for addressing NFC relay attacks, discussed in [21].

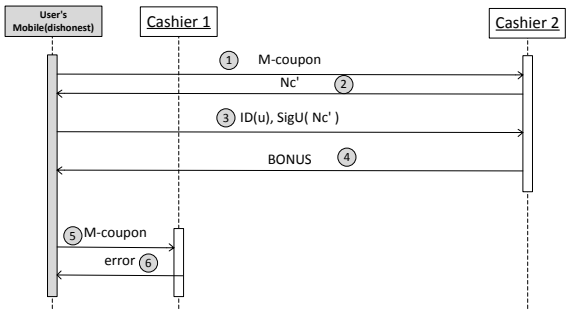
Even though authentication between the issuer and the user's mobile is not intended to be addressed by the protocol, it breaks the assumption that the M-coupon issued is linked to the user present at a particular geographical location.

B. Dishonest agent attack

There is another attack if there is a dishonest agent engaging in the protocol, as illustrated in Figure 4. Entities coloured in gray are the dishonest ones with their collaborative intruders. In Figure 4a, a dishonest cashier (Cashier 1) can forward the M-coupon to a collaborative intruder at another cashier



(a) Dishonest cashier attack



(b) Dishonest user attack

Figure 4. Attacks against the advanced M-coupon Protocol

(Cashier 2) who would be able to cash it. Then the dishonest cashier will tell the user he has already used his M-coupon at Cashier 2, and generate an error. When the user complains to the company, the investigation would show the user did indeed cash his M-coupon at Cashier 2, and will not find any evidence that the user had been circumvented. Separately to the protocol the user might be able to prove his presence at another location at the time that the M-coupon was cashed, but it would be very hard to find out who is the dishonest cashier, or to prove Cashier 1 was dishonest.

Figure 4b shows another possible attack with a dishonest user. The user can cash the M-coupon to a distant cashier (Cashier 2), then send the M-coupon to the near one (Cashier 1). Of course, Cashier 1 will reject the M-coupon since it has been used at Cashier 2, but the user will complain that he did not cash it because he is now at the location of Cashier 1.

V. SUGGESTED SOLUTION

The authors, Dominikus and Aigner, have followed the ISO authentication standard [18] when they developed their protocol, which insists on including the identity of the verifier

(Cashier) in order to address possible attacks. Nevertheless, the standard allows the omission of the identity of the verifier if the protocol is in a single direction, i.e., a client and server without server authentication, where the client is only ever authenticated to one server [18]. The NFC M-coupon protocol is not an example of such a system, as it is very likely to have many cashiers in the scheme. We recommend that the suggestion put forward by the standard, of omitting the identity of the verifier, should not be followed in this M-coupon system. An attack was found when the identity of the verifier was omitted.

In order to make this protocol more secure, there are two aspects which should be addressed; the protocol configuration, and the NFC configuration.

From the protocol's perspective, a user should include the identity of the verifier in any signature. At least the user must include the identity of a cashier in the user's authentication performed by a cashier (Figure 2b messages 5 to 8). The best way to do this is by a mutual authentication between the user's mobile and the cashier. Figure 5 is a modified version (with fewer messages) of the advanced protocol, which we designed according to the ISO authentication standard [18]. After the issuer sends a nonce to the user's mobile (message 1), the user's mobile signs the issuer's nonce and combines it with his identity and a nonce (message 2). Then, the issuer sends the M-coupon to the user (message 3). In the cashing phase, a mutual authentication is done between the user's mobile and the cashier. The cashier sends his identity and a nonce (message 4). At message 5, the user sends a signature of the M-coupon, the cashier's nonce, a new nonce generated by the user's mobile and the cashier's identity, combined with the user's mobile identity, the new nonce and the M-coupon. At this stage the user's mobile is authenticated to the cashier. The cashier can stop the protocol at this stage in case of an attack. Finally, the cashier sends the bonus and authenticates itself to the user by signing its nonce with the user's nonce, the user's identity and the bonus. At this stage the user makes sure the bonus came from the cashier.

There are two ways in which the user's mobile can communicate with a Trusted Third Party, online or offline. If the NFC M-coupon scheme is used as a part of a mobile wallet, the need for online access would be easy since the online access would be already available to other applications. If the NFC M-coupon is used as a separate application, offline authentication between the user and the cashier might be performed by exchanging certificates generated by a Trusted Third Party.

NFC configurations are very important in addressing a remaining threat: the relay attack [21]. When developing an NFC application, a user could be given the choice to start the communication directly without any confirmation as a feature of NFC (touch and go). Users should confirm any transaction in the M-coupon scheme. This will not only help to address the protocol attack, it will help to address the general NFC relay attack as well.

We formally analysed the security of this solution with Casper, and found no attacks. At the cashing phase, we were able to examine systems with up to two different users and two different cashiers.

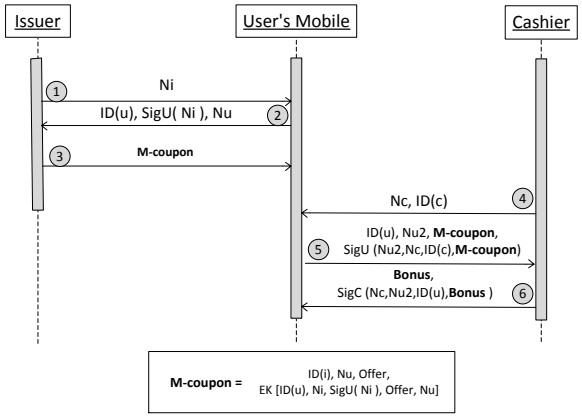


Figure 5. The Modified Advanced Protocol

VI. Conclusion

To the best of our knowledge, formal security analysis has not previously been carried out in the NFC domain. We formally analysed the security of NFC mobile coupon protocols proposed by Dominikus and Aigner [12], by using Casper. Our analysis identified an attack against the advanced protocol: that an intruder could cash an M-coupon even if he is not allowed to do so. This is because the M-coupon user's mobile generates signatures without including the identity of the cashier. We suggested a mutual authentication between the user and the cashier, and performed the same formal analysis on the resulting protocol, and identified that the attack was no longer present.

VII. Acknowledgement

This research was supported by Ministry of Higher Education in Saudi Arabia. We thank the reviewers for their helpful, constructive comments.

References

[1] K. Finkenzeller, *RFID Handbuch:Fundamentals and Applications in Contactless Smart Cards, Radio Frequency Identification and Near-Field Communication*, 3rd ed. John Wiley and Sons, Ltd., 2010.

[2] V. Coskun, K. Ok, and B. Ozdenizci, *Near Field Communication: from theory to practice*, 1st ed. John Wiley and Sons, Ltd, 2012.

[3] E. Haselsteiner and K. Breitfuß, "Security in near field communication (NFC)," in *In Proceedings of Workshop on RFID and Lightweight Crypto (RFIDSec06)*, 2006.

[4] C. Mulliner, "Vulnerability analysis and attacks on NFC-enabled mobile phones," in *ARES*, 2009, pp. 695–700.

[5] ECMA International, "NFC-SEC: NFCIP-1 Security Services and Protocol," 2010.

[6] ——, "NFC-SEC-01: NFC-SEC Cryptography Standard using ECDH and AES," 2010.

[7] Juniper Research, "Mobile coupons — ecosystem analysis and marketing channel strategy 2011-2016," Juniper Research, Tech. Rep., 2011.

[8] S. Clark. (2011) Survey: Discounts and coupons will drive adoption of mobile payments. [Online]. Available: http://www.nfcworld.com/2011/06/23/38289/survey-discounts-and-coupons-will-drive-adoption-of-mobile-payments

[9] S. C. Alliance, "Proximity mobile payments business scenarios: Research report on stakeholder perspective," Smart Card Alliance, Tech. Rep., 2008.

[10] C. Brown. (2011) "the future is NFC" says coupons.com exec. [Online]. Available: http://www.nfcworld.com/2011/03/10/36399/the-future-is-nfc-says-coupons-com-exec/

[11] T. Wolverton. (2002) Disney battles coupon goof. [Online]. Available: http://news.cnet.com/2100-1017-964831.html

[12] S. Dominikus and M. J. Aigner, "mCoupons: An Application for Near Field Communication (NFC)," in *Advanced Information Networking and Applications Workshops, 2007*, 2007, pp. 421–428.

[13] G. Lowe, "An attack on the Needham-Schroeder public-key authentication protocol," *Inf. Process. Lett.*, vol. 56, no. 3, pp. 131–133, 1995.

[14] C. A. R. Hoare, *Communicating Sequential Processes*. Prentice-Hall, 1985.

[15] P. Y. A. Ryan, S. A. Schneider, M. Goldsmith, G. Lowe, and A. W. Roscoe, *Modelling and analysis of security protocols*. Addison-Wesley-Longman, 2001.

[16] G. Lowe, "Casper: A compiler for the analysis of security protocols," *Journal of Computer Security*, vol. 6, no. 1-2, pp. 53–84, 1998.

[17] B. Donovan, P. Norris, and G. Lowe, "Analyzing a library of security protocols using Casper and FDR," in *Proceedings of the Workshop on Formal Methods and Security Protocols*, 1999, some of the Casper scripts are available here: http://web.comlab.ox.ac.uk/oucl/work/gavin.lowe/Security/Papers/prots.tar.gz.

[18] ISO/IEC, "Information technology – Security techniques – Entity authentication mechanisms – Part 3:Entity authentication using a public key algorithm," ISO, Geneva, Switzerland, 1993.

[19] D. Dolev and A. Yao, "On the security of public-key protocols," *IEEE Transactions on Information Theory*, vol. 2, no. 29, 1983.

[20] G. P. Hancke, "Practical eavesdropping and skimming attacks on high-frequency RFID tokens," *Journal of Computer Security*, vol. 19, no. 2, pp. 259–288, 2011. [Online]. Available: http://dx.doi.org/10.3233/JCS-2010-0407

[21] L. Francis, G. P. Hancke, K. Mayes, and K. Markantonakis, "Practical NFC peer-to-peer relay attack using mobile phones," in *RFIDSec*, 2010, pp. 35–49.