Formal Security Analysis of Near Field Communication using Model Checking

Nikolaos Alexiou[a], Stylianos Basagiannis[b,*], Sophia Petridou[c]

[a]*School of Electrical Engineering, KTH Royal Institute of Technology, Stockholm, Sweden*
[b]*Research Centre of United Technologies, Cork, Ireland*
[c]*Department of Applied Informatics, University of Macedonia, Thessaloniki, Greece*

Abstract

Near Field Communication (NFC) is a short-range wireless communication technology envisioned to support a large gamut of smart-device applications, such as payment and ticketing. Although, two NFC devices need to be in close proximity to communicate (up to 10 *cm*), adversaries can use a fast and transparent communication channel to relay data and, thus, force an NFC link between two distant victims. Since relay attacks can bypass the NFC requirement for short-range communication cheaply and easily, it is important to evaluate security of NFC applications. In this work, we present a general framework that exploits formal analysis and especially model checking as a means of verifying resiliency of NFC protocol against relay attacks. Towards this goal, we built a Continuous-Time Markov Chain (CTMC) model using the PRISM model checker. Firstly, we took into account NFC protocol parameters and, then, we enhanced our model with networking parameters, which include both mobile environment and security-aware characteristics. Combining NFC specifications with an adversary's characteristics, we produced the relay attack model, which is used for extracting our security analysis results. Through these results, we can explain how a relay attack could be prevented and discuss potential countermeasures.

Keywords: Near Field Communication, probabilistic model checking, relay attack, security analysis, wireless communication.

1. Introduction

Contactless radio communications, such as Near Field Communication (NFC), have become popular short-range solutions for establishing secure ad-hoc connections. NFC allows low data rate links of 106, 212 and 424 *Kbps* to transfer data over short distance (up to 10 *cm*) [1, 2]. Due to its simplicity, NFC technology is a suitable candidate for an increasing number of applications including

*Corresponding author
Email addresses: alexiou@kth.se (Nikolaos Alexiou), basagis@utrc.utc.com (Stylianos Basagiannis), spetrido@uom.gr (Sophia Petridou)

mobile payments, e-ticketing, access control systems and in-vehicle communications [3, 4, 5, 6]. The integration of NFC in smartphones in particular, transforms user devices into mobile wallets [7, 8] and carriers of authentication and authorization proof that is exchanged via short-range NFC channels.

Just as in typical RFID communications, two NFC-enabled devices can be paired in peer-to-peer mode or, alternatively, operate in card emulation mode for mobile-to-infrastructure communications. However, NFC systems are susceptible to attacks leaving the security an open issue [5]. In particular, relay attacks are easy to deploy and pose a serious threat for security of NFC systems, as well as for the acceptability of the technology. During a relay attack, the adversary acts as a transparent intermediary between two distant victim devices, i.e. an NFC reader and an NFC target, and maliciously *forces* an NFC link between them. This is achieved using a fast and transparent relay channel that connects the two victim NFC devices, which eventually believe they are in close proximity and can communicate directly with each other. The attack leverages on the absence of localization evidence of the NFC protocol, as well as on the fast relaying property of the adversarial channel, that eliminates the distance between the victim devices.

A prevalent countermeasure for relay attacks against NFC systems is the distance-bounding protocols [9, 10]. In a nutshell, they provide guarantees regarding the maximum distance between two communicating devices. Therefore, they prevent an attacker from faking the close-proximity property that is necessary to launch the relay attack. However, distance bounding has not yet been adopted to secure real world NFC systems, or is envisioned to do so in the near future. Although possible reasons are out of scope of this paper, we briefly mention the following: (i) Secure and reliable hardware implementations that can guarantee the tight timing constraints required by distance-bounding protocols, come with additional cost. Cost of several hardware implementations has not been defined yet [10]. However, the technology's acceptability to new applications can be slowed down or, even worse, hindered by additional costs. (ii) Software-only implementations of distance-bounding protocols sound promising, but suffer from a series of important issues like reliability and efficiency, not only in real world scenarios but also in lab environments. Thus even if possible, software-only distance-bounding solutions will need extensive work in order to be adopted [11].

It is therefore obvious, that we need methods to analyze and evaluate security of real world NFC systems. Formal analysis techniques constitute the perfect candidate [12, 13], since they can be applied to analyze the security of systems in a rigorous manner. Model checking techniques offer the additional advantage of automated rigorous analysis, which is beneficial for proprietary systems and new communication technologies [14], like NFC. In this paper, we propose a general framework that exploits formal analysis and especially probabilistic model checking to analyze security of NFC protocol against relay attacks. Towards this goal, we firstly built a highly configurable Continuous-Time Markov Chain (CTMC) model which takes NFC protocol specifications into account using the PRISM model checker environment [15]. Then, we enhanced our ba-

sic model with networking parameters, including both mobile environment and security-aware characteristics. In this way, we succeeded in combining an NFC transaction with the interference of an adversary, in order to construct our relay attack model. Based on this model, we extracted security analysis results, which can be exploited during protocol's design and implementation to evaluate the probability of a relay attack for a variety of protocol and adversary's characteristics and, thus, to explain how the attack could be prevented and to discuss potential countermeasures.

The remainder of this paper is organized as follows. Related work on relay attacks against NFC and the novelty of the proposed analysis are discussed in Section 2. Section 3 provides the NFC protocol specifications and describes the relay attack. Section 4 is a brief introduction to the probabilistic model checking preliminaries. The relay attack model along with security analysis details are presented in Section 5, while in Section 6 the results of the analysis are discussed. In Section 7 we set a market-wise NFC discussion and, finally, in Section 8 we conclude with remarks of the presented work as well as future directions.

2. Related Work

There is currently a major push for adopting NFC technology and its security guaranties in general purpose mobile devices [16, 17]. At the same time, proliferation of NFC technology has also attracted the attention of malicious users. To address these challenges the current bibliography focuses on analysis and techniques related to the resiliency of NFC to attacks.

In 2005, Kfir and Wool studied relay attacks on contactless smartcard communications focusing on operating ranges issues [18]. They highlighted the fact that the nominal range of 10 cm between the reader and the target can be circumvented by exploiting the attackers' hardware, consisting of a proxy and a mole. In practice, they showed that an extension of 50 m is feasible in the reader-to-proxy range, while mole-to-target range can be also extended up to $40 - 50$ cm. This entails that range limitations imposed by ISO/IEC 14443 standard can be overcome increasing the attackers' probabilities.

The same year, Hancke designed a low-cost system and executed a relay attack up to a distance of 50 m connecting a proxy and a mole through an UHF antenna [19]. His implementation was simple and cheap and introduced a small delay of $15 - 20$ μs, which is possible because the communication is relayed as analog data. The alternative approach of encoding/decoding and buffering the data packets requires additional processing time causing longer delay.

Recently, Issovits and Hutter presented a practical relay attack which exploits a number of mechanisms of the ISO/IEC 14443 standard, i.e. the Frame Waiting Time (FWT), the Negative Acknowledges (NAKs) recovery functionality and the Waiting Time eXtensions (WTXs) [20]. More specifically, they used an RFID-tag emulator with programming capabilities and a Nokia 6212 NFC mobile phone as proxy and mole correspondingly and a Bluetooth link between them as a relay channel. The legitimate parties, i.e. an ISO/IEC

14443 compliant reader and tag, exchange information using RFID links. The proposed attack reaches average delays of 85.3 ms. However, although they exploit protocol mechanisms for their attack and they propose some protection measures compliant with the standard, their approach is protocol dependable and restricted to their specific attack scenario. Moreover, the proposed countermeasures of checking the transmission parameters is not proven rigorously.

The practicability and complexity of relay attacks has greatly facilitated due to the availability of NFC-enabled mobile phones. Francis et al. [21] showed that it is possible to relay NFC signals over Bluetooth using two mobile phones. They further revealed that, with the introduction of software card emulation in some smart phones, it is even possible to relay contactless credit card and electronic passport transactions between two NFC mobile phones [11]. A Nokia 6131 NFC phone was configured as a proxy-reader between the smartcard and the relay channel, whilst a Blackberry 9900 phone is chosen to implement the proxy-token between the relay channel and the NFC-reader. Roland [22] further analyzed the security implications of software card emulation.

In current bibliography, the main countermeasure against relay attacks on RFID and consequently NFC systems is distance-bounding protocols [9, 10]. Their idea is to verify the proximity of two parties based on Round-Trip-Time (RTT) of cryptographic challenge-response pairs [11, 23]. According to [24], the distance-bounding protocols proposed over the last years roughly fall into two categories: those based on the Brands and Chaum protocol [23] and those based on the Hancke and Kuhn protocol [9]. Although distance-bounding is a well-researched area of security, the approach in most works is mostly informal [25]. A major issue is their hardware implementation which either ignores low-level implementation details, e.g. physical layer of the communication channel [26], or comes with additional cost [10]. According to [26], the conventional RF channels have been shown inadequate for secure distance bounding implementations, which should rather require special communication channels to facilitate accurate and secure distance estimates. Much progress has been made on practical distance bounding implementations for smart tokens [10, 27], but the integration of such channels into NFC-enabled devices has not been an industry priority [11].

2.1. Contribution

The authors in the aforementioned bibliography focused on proving practically the vulnerability of NFC systems against relay attacks. But, their studies remain dependable and therefore restricted to specific attack scenarios. Purpose of this paper is to address the security issues of NFC technology in a more general framework.

Our main idea (first presented at IWCMC2014 [28]) can be generalized according to Fig. 1 and described as follows. First, we built a highly configurable CTMC model, namely CTMC-NFC model, which takes NFC protocol specifications into account. For example, two core protocol parameters that affect the probability of an adversary to successfully launch a relay attack is the time-out during the data exchange protocol of NFC, since relaying packets causes

Figure 1: The proposed security analysis: an abstract representation

delays, and the payload size of the data transmitted, since additional data demand greater transmitting time and thus lead to delays. Intuitively, strict time constraints and high data volume could reduce this probability. Our analysis provides probabilistic results which support this intuition. Second, we augmented the CTMC-NFC model with networking parameters, including mobile environment and security-aware characteristics. The first ones refer to the data rate and the packet error rate over the NFC communication channel, while the latter ones express the adversarial strength as well as the quality of the adversarial channel. Combining an NFC transaction along with the interference of an adversary that establishes a fast and transparent relay channel, used to transfer the NFC messages from the victim Reader to the victim Target, and vice versa, we implemented the relay attack model. Finally, we used the tool of probabilistic model checking as a means of performing our automated security analysis and extracting our quantitative results to evaluate the attack probability for a variety of protocol and adversary's characteristics.

Towards the goal of proposing a general framework, the current work conducts a thorough security analysis of real world NFC systems providing the following contributions:

(i) it exploits formal methods and especially model checking as a means of analyzing and evaluating real world NFC systems. Formal analysis techniques are a perfect candidate, since they can be applied to analyze systems' security in a rigorous manner [13, 29, 30]. Additionally, model checking techniques offer the advantage of automated analysis, which is beneficial for proprietary systems and communication technologies such as NFC.

(ii) it incorporates NFC specifications, e.g. timeout during the data exchange protocol, payload size of the data transmitted, data rate and packet error rate over the NFC communication channel, in probabilistic model checking along with security-aware characteristics, e.g., the adversarial strength and the quality of the adversarial channel, in order to build a highly configurable CTMC model. In this way we succeed in proposing an approach which is general enough to verify the resilience of NFC protocols against

relay attacks. To the best of our knowledge, this is the first time that a model checking framework for relay attacks against NFC protocol has been developed in the PRISM model checker environment [15].

(iii) it provides a study that could be exploited by protocol's designers towards NFC improvements according to the security demands of services and applications used e.g. mobile payment, e-passport transaction, device pairing.

(iv) it proves rigorously the impact of the countermeasures proposed in the literature. More specifically, quantitative results can be extracted in order to provide evidence about the relevance between protocol parameters' values, e.g. timeout during the data exchange protocol, and the probability of a successful attack. This type of results could be also used by the protocol's designers to determine thresholds of the timeout parameter for different NFC applications.

3. Near Field Communication

NFC allows contactless communications between smart NFC-enabled devices in close proximity. NFC shares many similarities with RFID and offers a suite of protocols based on the ISO-14443 [31], which are standardized in [1] and [2]. NFC operates at a radio frequency of $f_c = 13.65\ MHz$, while offers three options for low data rate communications, namely $106, 212$, and $424\ Kbps$. In contrast to RFID, NFC supports short-range communications, typically up to 10 cm, and additionally bi-directional communication between devices. In summary, NFC supports two communication modes:

Passive Mode: In passive mode, NFC communication takes place between the *Initiator*, which is the active device sending data, and the *Target*, which is the passive/receiving device. During the NFC transaction, the Initiator's RF field is activated and the Target responds using a load modulation scheme. NFC in passive mode is commonly referred to as Reader/Writer and Card Emulation modes. Examples of passive mode NFC are a smartphone interacting with an NFC tag, or a smartphone interacting with a ticket validator. In the first case the smartphone plays the role of the Initiator, while in the latter case the smartphone acts in card emulation mode using the reader's RF field. The NFC in passive mode enables several real-life applications, such as contactless payments and e-ticketing [5].

Peer-to-Peer Mode: In contrast to passive mode, NFC in active mode allows two communicating devices to use their own RF field to transmit data. Both devices switch on their RF field when transmitting data in Initiator mode and sense the medium for the second device's RF field when in Target mode. The two devices switch between the Initiator and Target modes, eventually indulging in a peer-to-peer mode communication scheme. Active mode NFC can support application that involve device pairing [1, 2].

	Preamble	SYNC	Length	Payload	CRC
Size	48 bits min.	16 bits	8 bits	n 8-bit-bytes	16 bits

Table 1: Frame format fields and size

3.1. NFC Protocol

In this section we provide a thorough overview of passive mode NFC in 212 and 424 $Kbps$ data rate. More details regarding NFC specifications can be founded in [1, 2]. For the rest of the paper, we use the term Reader to refer to the Initiator of an NFC transaction. Table 1 shows the format of an NFC frame and the respective size for each of the frame's fields. The preamble consists of at least 48 logical zero bits and serves as the prologue of the frame. The SYNC field is 2 bytes long and the length is set equal to the number of bytes to be transmitted in payload plus one. The payload consists of n 8-bit-bytes of data, where n is indicated by the number of data bytes. Finally the CRC is a 2 bytes value attached to the end of the frame. When a frame is received a delay period of $8 \times 64 / f_c$ μsec, before the next frame is sent, is defined in the standard.

Overall, the NFC protocol comprises three phases: (i) RF collision avoidance, (ii) Initialization Single Device Detection, and (iii) Transport Protocol.

RF collision avoidance: The RF collision avoidance scheme is used to prevent collisions between nearby Readers having their RFs enabled in parallel. The Reader senses the medium continuously for a time period of $T_{IDT} + n \times T_{RFW}$ μs, where $T_{IDT} > 4096$, $T_{RFW} = 512/f_c$ is the RF waiting time, n is a randomly generated integer ($0 \leq n \leq 3$) and f_c corresponds to the radio frequency (13.56 MHz). The Reader enables its own RF field if no other RF field is detected.

Device Detection: Following Collision avoidance, Device Detection allows a Reader to detect NFC-enabled devices. For NFC in passive mode and data rate communications of 212 and 424 $Kbps$, a Reader can support up to 16 Targets in parallel using time-slotted device detection. In a nutshell, the Reader uses up to 16 time slots of duration T_s each, where T_s is $256 \times 64 / f_c$ μs. The Reader then probes nearby Targets by broadcasting a polling request packet. Each Target selects a random identifier R corresponding to a particular time-slot and, then, replies to the Reader during the time-slot that corresponds to R. In practice only one card can be supported, e.g., for e-ticketing applications.

Transport Protocol: Eventually, once the Reader selects a nearby Target, it starts data transmission using the Transport Protocol. During protocol activation, the Reader and the Target negotiate communication specific parameters, such as the expected *timeouts* during the data exchange protocol. The NFC protocol defines the range of acceptable timeouts from 302 μs to 4949 ms. During data exchange, the Target acknowledges each successful packet reception and replies to the Reader with its own data packets. A deactivation sequence of the Target is used to successfully finalize the protocol.

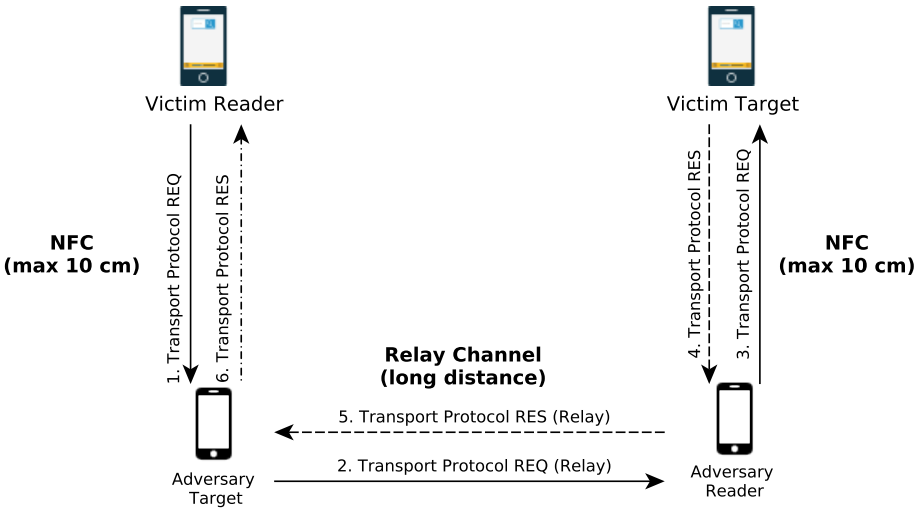

Figure 2: NFC relay attack setup

3.2. NFC Relay Attacks

Relay attacks against NFC are a typical example of man-in-the-middle attacks [32] and fit the Grand Master Chess problem described by Conway in 1976 [33]. In this context, a malicious player, who does not know the rules of Chess, could win against one of two grand masters by challenging them to a postal game. The player would simply forward the moves originating from one grand master to the other and vice versa. Although each grand master thinks that they are engaging the player they are essentially playing against each other.

Similarly, during a relay attack the adversary can remain agnostic of packet contents, protocol specifications or cryptography used, and relay all packets exchanged between the legitimate devices to achieve his malicious goals. For example, an attacker can circumvent an authentication protocol by simply relaying a challenge to a legitimate token, which will provide him with the correct response, which can then be relayed back to the verifier [11, 21]. In practice, the adversary needs to deceive the victim devices, namely the NFC Reader and Target, into believing that are close enough to establish an NFC communication link, while they are not. Since the NFC protocol poses timing only constrains to successfully complete a transaction, he can bypass them by establishing a fast and transparent relay channel, which is used to repeat the NFC messages between the victim devices. Using this channel the adversary minimizes communication delays caused by relaying NFC packets and essentially *helps*, as an intermediary, the victim devices to complete their transaction. The incentives of the attack are dependent on the NFC application and include obtaining unauthorized access to a building or buying digital goods illegally without the

consensus of the victim [6].

As depicted in Fig. 2, to launch a relay attack, the adversary needs two NFC-enabled devices that emulate an NFC Reader and a Target and are placed near, e.g., up to a distance of 10 *cm*, a victim NFC Target and Reader, respectively. Then, the adversary presents the Adversarial Target, i.e., AT, at the Victim Reader, i.e., VR to trigger an NFC transaction. AT has a preset communication link with the Adversarial Reader, i.e., AR, which acts as the relay channel and can be over a long distance. AR is simultaneously presented to the Victim Target, i.e., VT and emulates an NFC reader, simply relaying all messages received over the relay channel. The reverse procedure is followed for all VT responses. Eventually, if all messages are successfully relayed, the victim devices will complete an NFC transaction that would provide the adversary with the end result of the attack (e.g., to illegally buy digital goods).

4. Preliminaries of Probabilistic Model Checking

Model checking is a formal verification technique based on rigorous model definitions of systems, in order to discover errors, flaws or unexpected behavior in systems, protocols and hardware. Probabilistic model checking is a formal verification technique for the verification of systems that exhibit probabilistic behavior and tries to determine the probability of a model M satisfying a property *prop*. The probabilistic model checker PRISM [15] supports four types of stochastic models based on Markov Chains, namely: (i) Markov Decision Processes (MDPs), (ii) Discrete-Time Markov Chains (DTMCs), (iii) Continuous-Time Markov Chains (CTMCs) and (iv) Probabilistic Timed Automata (PTA). The proposed model follows the analysis of [34] and uses the CTMCs to verify the resiliency of the NFC protocol against relay attacks in continuous time.

A CTMC is the tuple (S, s_{init}, R, L), where:

- S is a finite set of states
- $s_{init} \in S$ is the set of initial states
- $R : S \times S \to \mathbb{R}$ is the transition rate matrix
- $L : S \to 2^{AP}$ is a labeling with atomic propositions

The transition rate matrix R assigns transition rates to each pair of states in S, which are then used as input to the exponential distribution. A transition from state s to s' can only occur if and only if $R(s, s') > 0$. Time spent at state s follows the exponential distribution and the probability of the transition being triggered within t time units is calculated as $(1 - e^{-R(s,s') \times t})$. Typically, more than one transitions from state s may occur in parallel, which is known as a *race condition*. The first transition to be triggered from s determines the next state of the CTMC. Time spent at s, before a transition, is exponentially distributed with rate $E(s) \stackrel{def}{=} \sum_{s' \in S} R(s, s')$.

$E(s)$ is defined as the exit rate of state s. The actual probability of reaching state s' from state s independently of time can be calculated using the embedded DTMC $emb(C) = (S, s_{init}, P_{emb}(C), L)$, where:

- S is a finite set of states

- $s_{init} \in S$ is the set of initial states
- $L : S \to 2^{AP}$ is a labeling with atomic propositions

$P^{emb(C)}(s, s')$ is calculated using $E(s) = \Sigma_{s' \in S} R(s, s')$ as follows:

$$P^{emb(C)}(s, s') = \begin{cases} R(s, s')/E(s) & \text{if } E_s > 0 \\ 1 & \text{if } E_s = 0 \text{ and } s = s' \\ 0 & \text{otherwise} \end{cases}$$

Properties describing expected model's behaviour are defined to perform model checking. In PRISM, properties are defined in a superset of the several temporal logics and more specifically the (i) Probabilistic Computation Tree Logic (PCTL), (ii) the Continuous Stochastic Logic (CSL), (iii) the Linear Temporal Logic (LTL), and (iv) PCTL*. For CTMCs, properties are expressed in CSL in the following syntax:

$$\Phi ::= \ true \mid \alpha \mid \neg\Phi \mid \Phi \wedge \Phi \mid P_{\sim p}[\varphi] \mid S_{\sim p}[\Phi$$

$$\varphi ::= X\Phi \mid \Phi \ U^I \ \Phi$$

where α is an atomic proposition, $\sim \in \{<, >, \leqslant, \geqslant\}$, $p \in [0, 1]$ and I is an interval of $R_{\geqslant 0}$. $P_p[\varphi]$ denotes the probability that the path formula φ being satisfied given the probability bound p. As with PCTL, it is straightforward to derive CSL operators for F and X denoting "eventually" and "next" respectively [34].

A PRISM model is a collection of modules that are active in parallel. Each module, in turn, comprises of a set of local variables and labeled actions (e.g., model transitions between states). Through the actions, the variables are updated according to the specifications of the modeled system (e.g., the protocol), which defines the state of the module. Eventually, the global state of the model is built upon the individual module states at each point in time. Each module action has two parts, the guard and the update actions:

$$[L] \ guard \Rightarrow R : u_1 + ... + u_n;$$

where L is the label naming the model transition, *guard* is set of prerequisites to trigger the command (e.g., variables values), R is the rate of the command if the *guard*'s conditions are met, and u_i is an update executed by the command. We express our properties in the PRISM model checker using the $P = ?[F \ \varphi]$, which gives a numerical estimation of the probability that the model "eventually" satisfies φ.

5. Prism Model of NFC Relay Attacks

We model a relay attack against passive mode NFC operating at 212 and 424 *Kbps*, as defined in [1, 2], and we study the indicative use case of a request and the corresponding response packet, which are exchanged between the Reader and the Target during the NFC transport protocol. Request packet is also acknowledged by the Target as mentioned in Section 3.1. We consider a relay attack successful when all communication packets (i.e., the NFC protocol

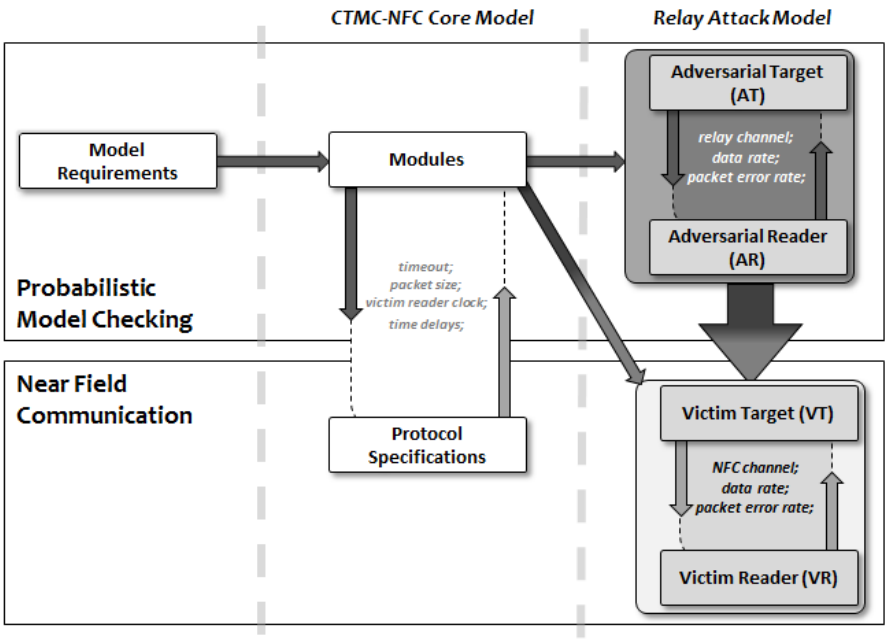

Figure 3: The NFC relay attack analysis using probabilistic model checking

packets and the data packets) are successfully relayed by the adversary. Our CTMC model is minimal in the sense that the NFC protocol is modelled with enough detail to demonstrate the attack. Meanwhile, our model is highly configurable and thus may be used to verify resilience of other NFC protocol types and modes, e.g., active mode NFC, as well as more sophisticated NFC protocols against relay attacks.

Fig. 3 provides a graphical representation of our NFC relay attack analysis exploiting probabilistic model checking. As it has already been mentioned in Section 2.1, we start building a core model, namely the CTMC-NFC model, which takes into account the NFC protocol specifications. According to Table 2, two basic protocol parameters that affect the probability of an adversary to successfully launch a relay attack is the *timeout* during the NFC transport protocol and the *size* of the data transmitted. Both the range of *timeout* values and the *size* of the NFC packets, used in our model, are derived by the NFC standards [1, 2]. We particularly focus on the low end values of *timeout* range, since strict time constraints could reduce the probability of an attack and, thus, they serve as a countermeasure. Regarding the *size* of packets exchanged during the data transport protocol, it varies depending on the NFC application. Section 6 describes two experimental setups, namely the low- and high-data volume scenarios, which also exhibit the impact of packet *size* into reducing the probability of an attack. In addition, we use the *cl* parameter to denote the

	Symbol	Description
Protocol specifications	*timeout*	Transport protocol timeout
	size	Packet size
	cl	Victim reader clock
	del	Time delays
Networking parameters	*er_rch*	Relay channel packet error rate
	er_nfc	NFC packet error rate
	dr_rch	Relay channel data rate
	dr_nfc	NFC data rate

Table 2: CTMC-NFC notation

clock at the VR and to count the time. The *del* parameter consider all possible protocol delays that can occur during an attack, e.g. packet receiving delays, packet preparation delays, and not only the transmission ones. However, it can be adjusted according to the type of NFC application and hardware used. This makes our PRISM model configurable and realistic. For our results, we set *del* to a low value, namely 100 μs, in order to verify the best case scenario for the adversary, where delay is negligible.

At a second step, we enhanced our core CTMC-NFC model with networking parameters, including mobile environment and security-aware characteristics. In this way, we succeeded in combining an NFC transaction with the interference of an adversary, in order to construct our relay attack model. The latter comprises four (4) modules, one for each of the devices involved in the attack scenario:

(i) **VR:** The Victim Reader VR is the verifier during the NFC transaction. Therefore, it is the victim of the NFC relay attack, since the adversary tries to maliciously authenticate a Target to the VR.

(ii) **AT:** The Adversarial Target AT is used by the adversary to communicate with the VR over an NFC link and with the AR through the fast relay channel.

(iii) **AR:** The Adversarial Reader AR is the reader operated by the adversary (e.g., a smartphone) which initiates the NFC transaction with the VR. AR communicates over an NFC link with the VT and through the fast relay channel with the AT.

(iv) **VT:** The Victim Target VT communicates over an NFC link with the AR and has a virtual NFC connection with the VR as a result of the relay attack.

The aforementioned modules are associated with the networking parameters, listed on Table 2. More specifically, the modules VR and VT are deceived into believing that communicate with each other over an NFC link, while they

actually use an NFC link with AT and AR, respectively. Thus, they make use of the mobile environment parameters which refer to the data rate over an NFC channel, i.e., dr_nfc, and the packet error rate during the NFC communication, i.e., er_nfc, for which we use the typical value of 10^{-8}. On the other hand, the modules AT and AR communicate over a fast and transparent relay channel, for which we use the security-aware parameters dr_rch in $Kbps$ and er_rch to define the data rate speed and the packet error rate, correspondingly. The dr_rch parameter serves as an indicator of adversarial strength, while er_rch is an indicator of the adversarial channel quality.

Using the parameters $size$ and dr_nfc, we model the time required to complete a packet transmission. As an example, the time needed to transmit an NFC message of n bits using an NFC channel of dr_nfc $Kbps$ can be modeled as n/dr_nfc. The above is exploited in PRISM to model the rate, i.e., dr_nfc/n, to complete a packet transfer, as described in Section 4. Our model expresses a simple but realistic two-step communication, where one packet from VR to VT constitutes the NFC request (REQ) and a second is the corresponding response (RES) from VT to VR, as depicted in Fig. 2. Diffie–Hellman key exchange is such a communication scenario [35]. Indicative examples of modeling basic actions, such as NFC packet transmission, timeout and packet relay, using the parameters discussed and listed on Table 2 follow.

Listing 1: NFC packet transmission modeling

```
1 [Tr_Prot_Req] (state_vr = 5) ->
2 (dr_nfc / size) * (1 - er_nfc) : (state_vr' = 6);
3
4 [er_Tr_Prot_Req] (state_vr = 5) ->
5 (dr_nfc / size) * er_nfc : (er_state_vr' = true);
```

Listing 1 shows how the transmission of a transport protocol packet is modeled in PRISM, where $dr\_nfc/size$ defines the transmission rate, as explained above. This rate is increasing as the communication data rate increases, and drops when the size of the transmitted packet increases. Tr_Prot_Req and $Tr\_Prot\_Req\_error$ are used to create PRISM labeled transitions and differentiate the successful and non-successful transmission of a packet due to communication errors, which occur with a rate defined by er_nfc. The modeling of networking and protocol specific parameters is done in one model transition in order to keep model size to a minimum. Finally, $state\_vr$ defines the state of the VR according to the protocol step.

Listing 2: Timeout modeling

```
1 [clock] (state_vr = 7) & (cl + 1 < timeout) ->
2 tstep : (cl' = cl + 1);
3
4 [clock] (state_vr = 7) & (cl + 1 >= timeout) ->
5 tstep : (error_state_vr' = true);
```

Listing 2 shows how time is measured at the VR when the transport protocol data packet is sent. Each time unit corresponds to 1 μs. While the clock value is below the defined *timeout*, the protocol continues without reaching a fail state. If the *timeout* is reached then the VR transits to the fail state and the attacker fails to complete the attack.

Listing 3: Packet relaying modeling

```
1 [Tr_Prot_Rel] state_ac=10 ->
2 (dr_rch / size_rel) * (1 - er_rch) : (state_ac' = 11);
3
4 [Tr_Prot_Rel_error] state_ac = 10 ->
5 (dr_rch / size_rel) * er_rch : (error_state_ac' = true);
```

Finally, Listing 3 shows the packet relaying modeling. The approach is similar to that of Listing 1, but it uses the networking parameters of the relay channel, i.e, *dr_rch* and *er_rch*. Transitions in Listings 2 and 3 are enabled in parallel and are involved in a race condition as explained in [34]. In brief the transition that is enabled first by its corresponding rate, defines the next state of the model, which in our case can be an increased value of the clock or a packet transmission.

Overall, once our relay attack model is implemented, we perform the automated security analysis based on the parameters that affect the probability of a successful relay attack. From the above description, it entails that these parameters are: *timeout* during the transport protocol, the *size* of data transmitted, the adversarial strength expressed by the *dr_rch* and the quality of the adversarial channel expressed by the *er_rch*. Among these parameters, the adversary is *timeout* agnostic, whilst he has the control over data and packet error rate of relay channel. Both parameters express his capability of fast and efficiently relaying packets and as it is mentioned they indicate his strength and his channel quality, correspondingly. The latter, however, could be tuned using friendly jamming techniques leading to significant deterioration of channel's quality and subsequently limiting the attacker's probability of a successful relay [36, 37]. On the other hand, the NFC end-user cannot choose neither the *timeout* value out of a range nor the packets' *size*. Thus, the probabilities derived by the proposed analysis can be mainly exploited by the protocol's designers to carefully lunch NFC configurations according to the security demands of services and applications used e.g. mobile payment, e-passport transaction, device pairing.

6. Model Checking Results

We study the impact of the parameters presented in Table 2 on the successfulness of relay attacks against the NFC protocol. As it has been mentioned in Section 5, our analysis can be fit to a simple but realistic two-step NFC communication protocol [35] that involves transmission of two packets during the data transport: a request (REQ) sent by the Reader and a response (RES) sent by the Target (Fig. 2). This way we balance between a practical but low-complexity

timeout	States	Transitions	Iterations	Time (s)
500	72840	182263	2428	9.3
2500	218840	572263	2554	30.5
5000	401340	1059763	5054	109.2

Table 3: Relay attack model's state space

model, since the proposed CTMC-NFC can be extended to include additional exchanged data packets, at the cost of model size and time. This is a well known issue in model checking that can be tackled with more computational resources and further abstract modeling, which reduce the state space [34]. Our future plans include both considerations.

Our results show that *timeout* is a decisive parameter that can be used to thwart the attack under certain conditions. At the same time, our analysis reveals that even when strict timing constraints apply, strong adversaries with high quality and fast relay channels can launch a successful relay attack with high probability. We use the relay channel data rate dr_rch to measure adversarial strength and packet error rates er_nfc and er_rch to define quality of communication over the NFC and relay channels. We consider passive mode NFC communication with dr_nfc at 212 and 424 $Kbps$. To study the impact of $size$ of NFC transport protocol packets we use two experimental setups, both complied with the NFC standards [1, 2]. The first *high-data* setup considers 255 and 160 bits of data packets transmitted by VT to VR and VR to VT respectively. Additionally, packets of 80 bits from both the VR and VT are used in the *low-data* setup. It is worthwhile to mention that our probabilistic model is highly configurable and can support a variety of setups regarding the size of data transmitted. This makes it also applicable in a variety of NFC application scenarios. Overall, we restrict ourselves to discussing Quality of Service aspects related to the parameters used for the analysis, since they differ greatly from application to application.

The size of our prism model in terms of number of states, transitions, iterations and model checking time is given in Table 3. The parameter that affects the model size is *timeout*. The rest of parameters are important in the sense that they define the transitions' rate, but they do not affect the final state space. This entails that further configuration of our model driven by $size$, dr_nfc and dr_rch is possible without affecting the state space. Model checking time presented in Table 3 is indicative and can be reduced when stronger machines are used. For our results we used an Intel(R) Core(TM) i5-2410M, 2.30 GHz processor.

Fig. 4 shows the probability of a successful relay attack for the low-data volume setup, using a range of *timeout* values, derived by the NFC standards [1, 2], and significantly low packet error probability ($er\_rch = 10^{-8}$), which is a best case scenario for the adversary [38]. dr_rch denotes the adversarial strength ex-

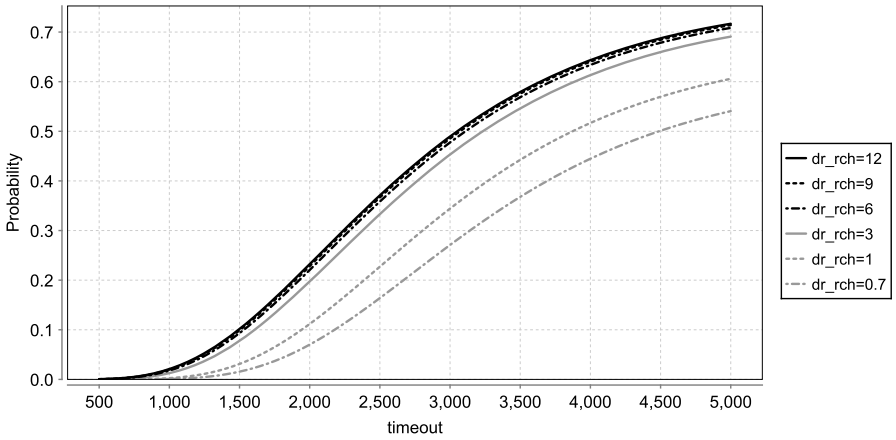

Figure 4: Probability of successful attack vs *timeout* for different relay channel data rates (*dr_rch*); low-data volume at 212 *Kbps*

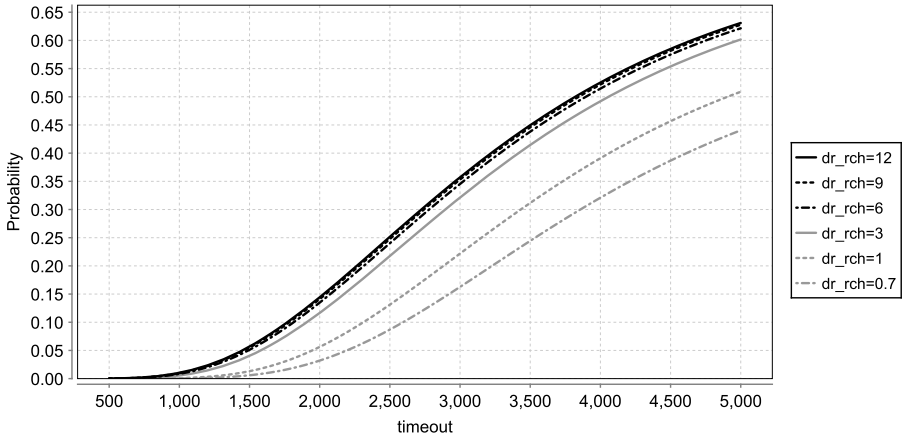

Figure 5: Probability of successful attack vs *timeout* for different relay channel data rates (*dr_rch*); high-data volume at 212 *Kbps*

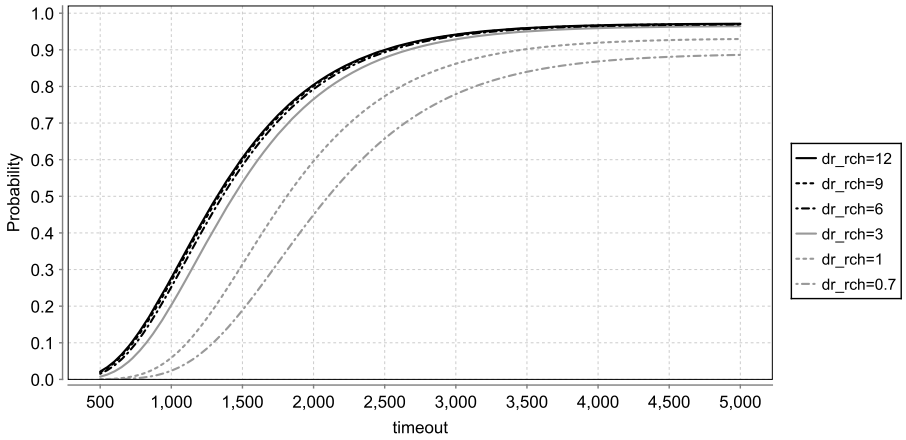

Figure 6: Probability of successful attack vs *timeout* for different relay channel data rates (*dr_rch*); low-data volume at 424 *Kbps*

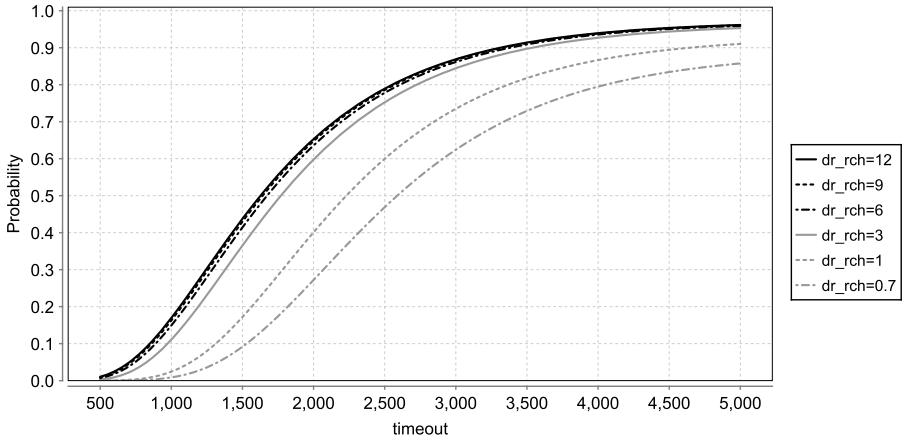

Figure 7: Probability of successful attack vs *timeout* for different relay channel data rates (*dr_rch*); high-data volume at 424 *Kbps*

pressed by the transfer speed of the adversarial relay channel in $Mbps$. We observe that strong adversaries with $dr\_rch > 3$ $Mbps$ have a significantly higher probability of performing a successful relay attack, which is above 60% for $timeout > 4.5$ ms. However, it is also clear that even weaker adversaries with slower relay channels ($dr\_rch = 0.7$ or 1 $Mbps$) exhibit a very high probability of successfully launching the relay attack when $timeout$ is high. On the other hand, the probability of a successful attack minimizes as $timeout$ reaches at 0.5 ms. For example, when $timeout < 1.5$ ms the probability of a successful attack for all adversarial strengths is less than 12%.

Fig. 5 follows the analysis of Fig. 4 for the high-data volume setup. The model checking results are similar with those of Fig. 4, but as expected, the increased volume of data that have to be relayed poses an additional challenge for all adversarial strengths. Comparing the low and high data cases, we pinpoint that for $dr\_rch = 3$ $Mbps$ in the high-data setup the probability of successful attack is approximately the same with the low-data case for $dr\_rch = 1$ $Mbps$, when $timeout = 5$ ms. In other words, an adversary has to use a faster relay channel in the high-data case, in order to retain the same probability of success as in the low-data case, since bigger payload sizes naturally demand longer time to be relayed.

In Fig. 6 and 7 we study the impact of dr_rch on the attack, when $dr\_nfc = 424$ $Kbps$. We observe that the probability of successful attack is higher when a faster relay channel is used. Even for $dr\_rch = 0.7$ $Mbps$ the probability of a successful attack is higher than 80% in both setups. The adversary essentially takes advantage of the faster dr_nfc to succeed the attack, since the packets can now be relayed faster. Moreover, we observe that the probability of a successful attack reaches close to a maximum value faster when $dr\_nfc = 424$ $Kbps$ compared to $dr\_nfc = 212$ $Kbps$, especially for the low-data setup. Indicatively, in the low-data setup when $dr\_nfc = 212$ $Kbps$ and $dr\_rch = 12$ $Mbps$, the probability reaches close to 60% when $timeout > 3.5$ ms. Changing only $dr\_nfc = 424$ $Kbps$ the probability reaches 60% in just 1.5 ms.

Fig. 8 and 9 demonstrate the probability of a successful relay attack when strict $timeout$ is defined, i.e., $timeout = 500, 750$ and 1000 μs. In a sense, these are worst case scenarios for the adversary since only a limited time window is available to relay the NFC packets. For both data volume setups, low packet error rate, e.g., $er\_rch = 10^{-8}$, and $dr\_nfc = 212$ $Kbps$, we observe that the probability of an attack decreases dramatically for stricter timeouts, even if the adversary uses fast relay channels. For the low-data case scenario in particular, the probability of attack does not exceed 2.5% when $timeout = 1$ ms, and is less than 0.0025% when $timeout = 0.5$ ms. For the high-data setup the probability of attack is even smaller, since the adversary has to relay more bits.

Fig. 10 and 11 demonstrate the impact of using increased data rate for NFC communication, i.e., $dr\_nfc = 424$ $Kbps$, on the probability of the relay attack. In this case, we observe that the adversary has much higher probability to succeed for both data volume setups. For the low-data setup the probability of successful attack reaches almost to 30% when $timeout = 1$ ms, and it is above 2.5% when $timeout = 0.5$ ms. Even in the high-data setup, the probability

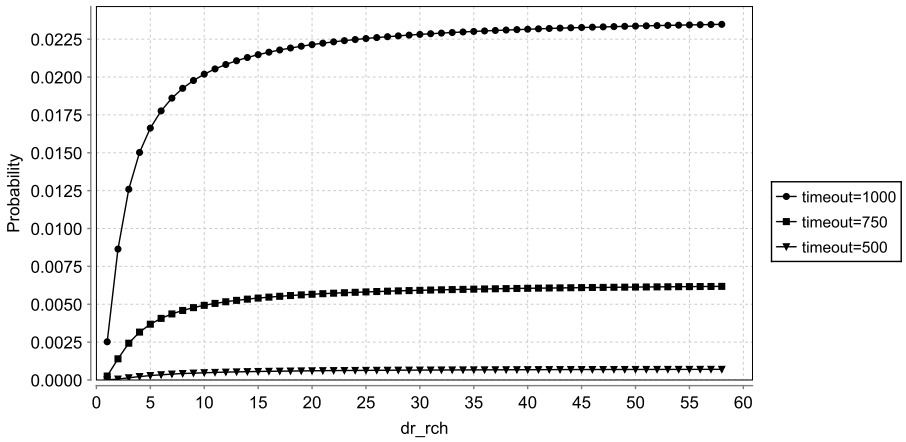

Figure 8: Probability of successful attack vs adversarial strength (dr_rch) for worst case scenario of $timeout$; low-data volume at 212 $Kbps$

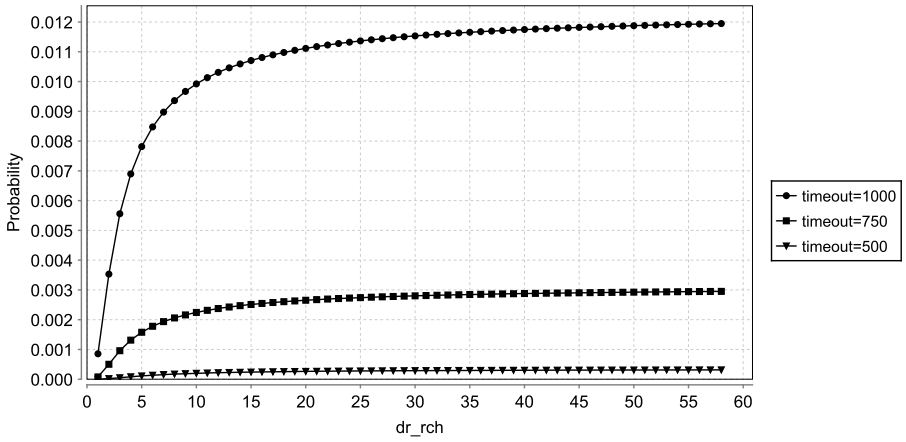

Figure 9: Probability of successful attack vs adversarial strength (dr_rch) for worst case scenario of $timeout$; high-data volume at 212 $Kbps$

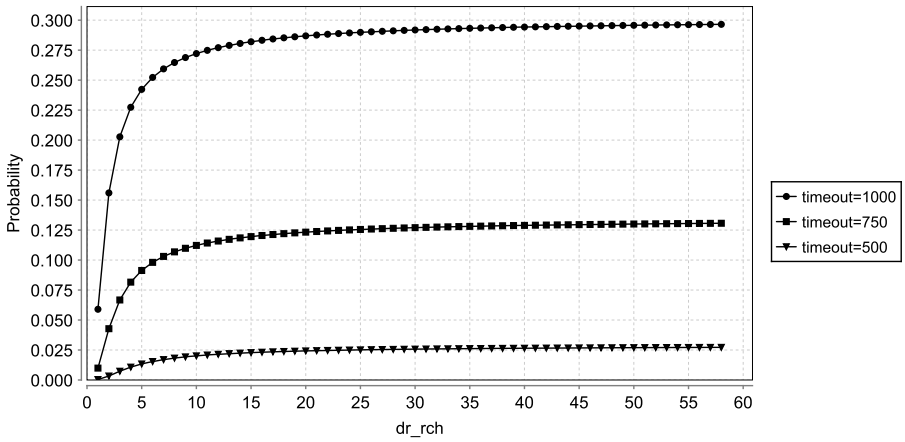

Figure 10: Probability of successful attack vs adversarial strength (dr_rch) for worst case scenario of timeout; low-data volume at 424 $Kbps$

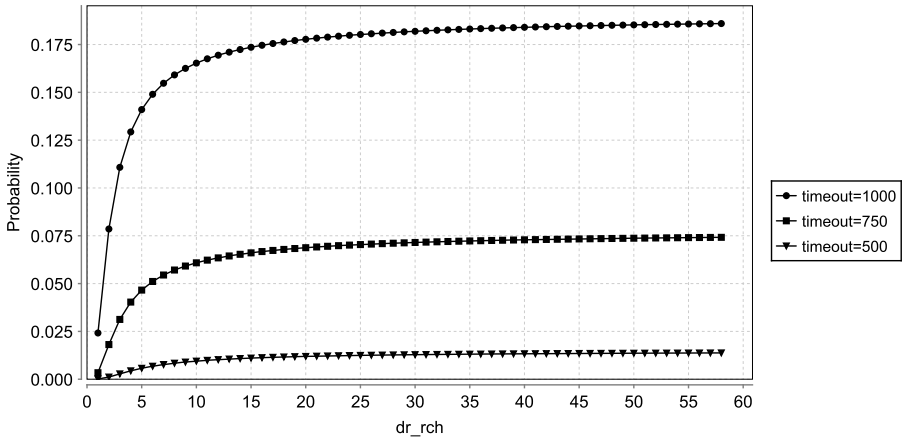

Figure 11: Probability of successful attack vs adversarial strength (dr_rch) for worst case scenario of timeout; high-data volume at 424 $Kbps$

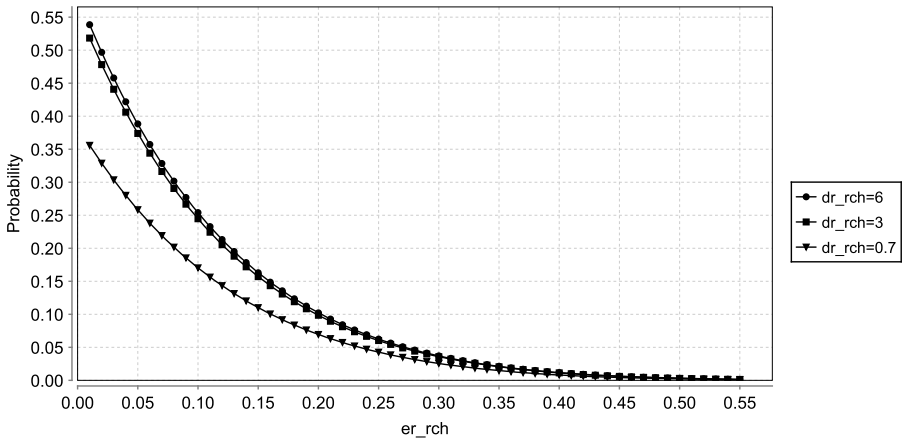

Figure 12: Probability of successful attack vs quality of adversarial channel (er_rch) for different relay channel data rates (dr_rch); low-data volume at 212 $Kbps$ and $timeout = 3.6\ ms$

remains considerably higher when comparing to NFC 212 $Kbps$. In summary, we observe that when $timeout$ is on the low-end values, i.e., $timeout < 0.75\ \mu s$, the probability of the attack is decreased in all cases. Especially when using $timeout = 0.5\ ms$ the probability of a successful attack drops to less that 2.7% in all cases. As a result, our analysis provides the evidence that the $timeout$ is an important parameter to thwart a relay attack. It is also worthwhile to be mentioned that a fast relay channel increases the chances of an attack for NFC 424 $Kbps$.

To complete our analysis, we test the probability of a successful relay attack for different qualities of adversarial channel, which is expressed by relay channel packet error rate, i.e. the er_rch parameter. Packet errors may occur when the adversary relays packets over longer distances where interference is expected. We test both our experimental setups for different values of er_rch, $dr\_nfc = 212$ and 424 $Kbps$ and for $timeout = 3.6\ ms$. This way we show the impact of er_rch on the probability of a successful attack that under no packet errors would be high; for example greater than 35% for all tested dr_rch in Fig. 4. Fig. 12 and 13 show the probability of a relay attack for the low and high-data setups when $dr\_nfc = 212\ Kbps$. When packets are dropped due to higher packet error rates, it is increasingly hard for the attacker to succeed in the attack. In both Fig. 12 and 13, we observe that for $er\_rch = 0.4$ the probability of the attack is less than 5% for all tested adversarial strengths. In Fig. 14 and 15, we test the same scenario when $dr\_nfc = 424\ Kbps$. As shown above a higher dr_nfc increases the adversary's chances to succeed an attack. However, packet errors can also prevent the adversary from launching a successful attack as in case of $dr\_nfc = 212\ Kbps$.

These last results are particularly significant, since they demonstrate that packet errors on relay channel could help to prevent an attack. Through this

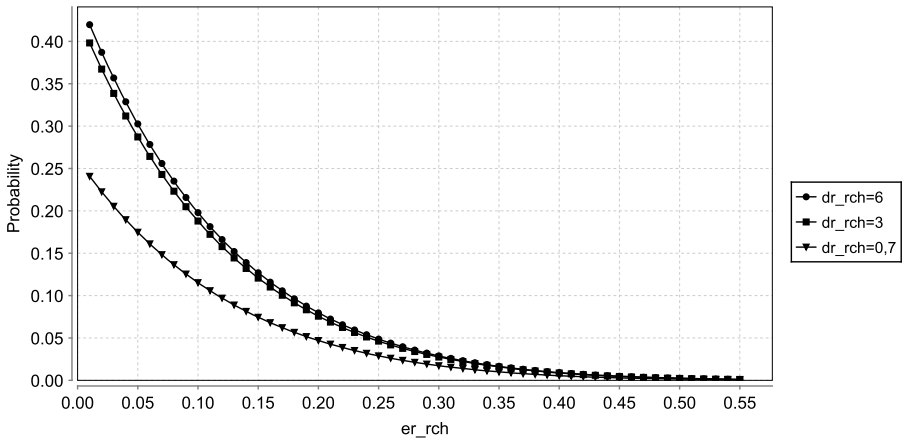

Figure 13: Probability of successful attack vs quality of adversarial channel (er_rch) for different relay channel data rates (dr_rch); high-data volume at 212 $Kbps$ and $timeout = 3.6\ ms$

observation, it is not assume that the adversary uses an unreliable channel; besides all previous results derived for $er\_rch = 10^{-8}$ [38]. We just highlight the twofold information provided: firstly, the adversary can lunch attacks with considerable probability even when $er\_rch > 10^{-1}$, which is placed at the high-end typical values for wireless links [38]; the second outcome is that results of Fig. 12 - 15 could provide application-oriented thresholds on how friendly jamming techniques can tune packet error in relay channel in order to deteriorate its quality and thus revealing an attack [36, 37].

7. Market-wise NFC and security challenges

Near Field Communication solutions have found their way in today's smart devices, opening up a broad range of applications [17]. Though NFC obvious characteristic that communication handshake occurs within a centimeter distance compared to the present wireless communications technologies (WiFi or Bluetooth), its security protection increases significantly as in practice an attacker has to be close enough to the reader in order to eavesdrop a message exchange. On the other hand, the aforementioned fact constitutes NFC devices susceptible to other advanced attack methods (e.g. [18]) related to reader emulators, proxy enablers, directed jamming or even low power electromagnetic interference. For example, consider a physical access control scheme where a Bring Your Own Device (BYOD) system is applied i.e. the user carries its credential-access control card to his smart phone. The exchanged NFC messages containing the identity of the user actually will initiate the verification process against the predetermined database of users, privileged-by-authority to access the protected resource. Raising the fact that NFC can transmit the appropriate user message to trigger a door unlock process, the impact of acquiring

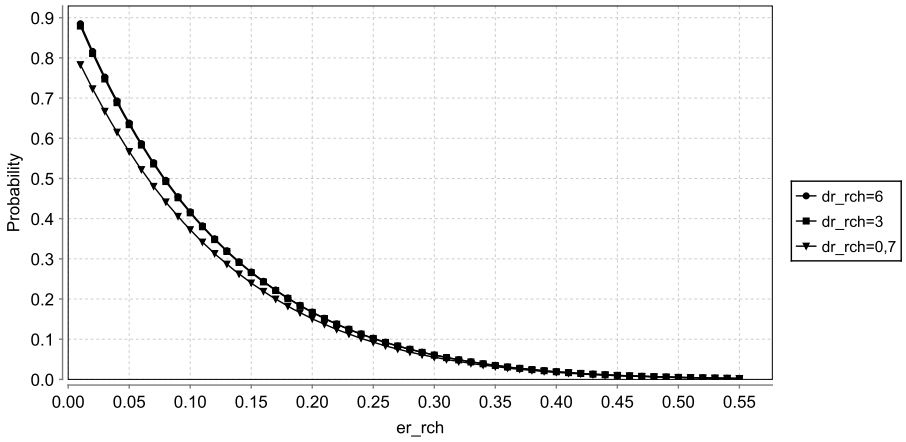

Figure 14: Probability of successful attack vs quality of adversarial channel (er_rch) for different relay channel data rates (dr_rch); low-data volume at 424 $Kbps$ and $timeout = 3.6\ ms$

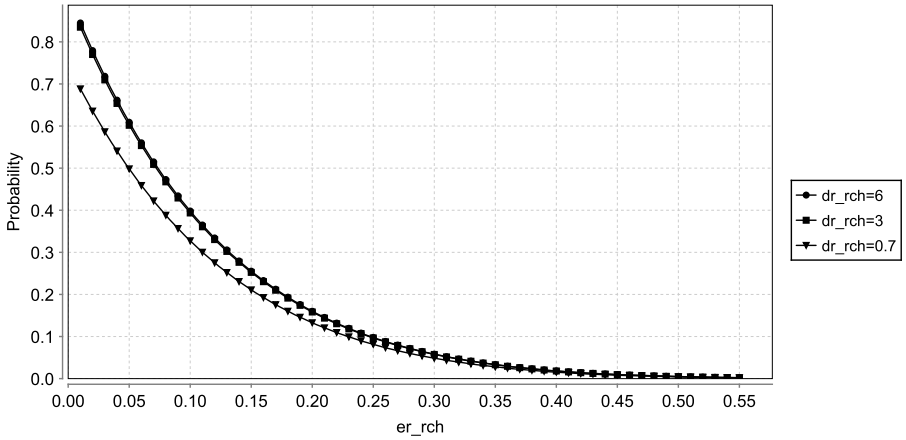

Figure 15: Probability of successful attack vs quality of adversarial channel (er_rch) for different relay channel data rates (dr_rch); high-data volume at 424 $Kbps$ and $timeout = 3.6\ ms$

such a message exchange by an attacker increases prohibitively the risk of having unauthorized access to a resource [39], thus motivating the attacker to engage maliciously with the NFC device.

Market-wise, NFC has shown a dramatic increase especially in the so called NFC-enabled smartphone devices for access control, electronic payments, electronic ticketing, as well as data sharing and device pairing. Till 2009, this technology was undergoing pilot testing phase from many companies around the world. Reports today estimate that in 2011, only 5% of the total mobile phones were NFC-enabled while it is predicted that by 2016 this number will rise to 46% [16]. Such a fact will force NFC to be adopted in new devices and consequently increase its application domains further (e.g. smart advertisement and infontainment)[1]. Such evolution will motivate protocol designers to invest in through NFC protocol analysis towards its security guarantees. The latter is supported by our analysis results, presented in Section 6, which demonstrate that protocol specifications, i.e., timeout during the data exchange protocol and size of packets exchanged, as well as, networking parameters that indicate adversarial strength and its channel quality, can affect the probability of a successful attack. But the question would still remain: will the NFC solution provide secure communications between IoT devices, or will the NFC solution enable additional backdoor opportunities to malicious users while offering its services? And more important, can NFC be considered as a secure solution for electronic payments?

Famous contactless electronic payments already are enabled through NFC communication [41]. Both Google Wallet [7] and ApplePay [40, 42] adapt NFC communication protocol principles. Reviewing both of the technologies, we argue that they try in common to solve the same set of problems related to mobile payment and storage of the cardholder's data; but they operate differently. Secure element (SE) is a solution adopted by both Google Wallet and ApplePay for storing and executing crypto operations. Having SE executing a payment transaction virtually as a contactless card, offers a strong authentication for the user as payment authorization can be performed in a secure way. Heterogeneity though [43], due to different wallet applications forced the aforementioned architecture to change. In addition to that, as not all of the NFC applications require security (e.g. device pairing), NFC parameters have to be reconfigured real-time when security level varies.

Protocol designers need at this point, to take under consideration which NFC parameters to tune based upon the security attacks such as relays, end-user applications, device characteristics and obsolescence. As the NFC controller

[1]iPhone recent versions include NFC technology enabling contactless payment applications such as ApplePay. Apple application patent in 2013 [40] discloses a shared near field communication solution integrated with sensor structures for authentication services. Such evolution will inevitably increase the NFC market as it indicates that there is a business value potential for the technology to be exploited further in future iPhones.

itself does not deal with the data or processing associated with the payment transaction, we argue that it is important to rapidly evaluate and verify their prototypes for attacks. Through our analysis, we are able to provide the means for the protocol designer to delegate carefully NFC parameters (e.g. timeout, payload size, reader clocks) according to the critical level of the services and applications used. Such a reconfiguration has to be in line with NFC devices capabilities and executed when the application security level is critical.

As discussed in Section 3.2, proximity does not mean security. Attackers can use high power antennas against the smart device or relay the NFC signal in order to gain unauthorized access [44]. Considering that NFC-enabled devices can be used for electronic payments, the fact that the NFC chip will be the means for transferring sensitive information (e.g. credit card details) strengthen the need for a collaborative NFC solution within the host device. NXP P5Cx[2] series [45] contain hardware based cryptographic solutions for collaborating with the NFC modules. To this end, Google Wallet in a joint agreement with MasterCard implements secure payments in a specialized hardware, the so-called Secure Element, SE. The SE chip mission is to decouple the operating system and smart devices applications with the external reader, allowing direct NFC secure interaction only between the embedded SE and the reader. SE applications are locked by the phone manufacturers and only applets previously signed by appropriate authorities can be installed to it [46]. Furthermore, access to the SE's environment is highly secured as the element provides delimited memory for each SE application and other functions that can encrypt, decrypt or sign the data packets to be transmitted. But limitations set by the SE chip e.g. memory capacity, force companies to seek SE implementation in portable memory means (SIM/UICC or SD card) enabling additional external threats due to device-host obsolescence and heterogeneity.

8. Conclusions and Future Work

In this paper we presented a framework that exploits probabilistic model checking to evaluate the resiliency of NFC protocol against relay attacks. Our automated analysis can verify the probability of a successful relay attack against NFC handshakes, based on a variety of characteristics, i.e., NFC specifications, mobile environment and security-aware parameters, that affect the attack. In practice we present an approach to test security of NFC and we use it to evaluate cases of indicative NFC applications. By capturing system-level NFC characteristics in a formal model within the PRISM model checker, and modeling intrusion tactics, such as a relay attack activity, we have been able to extract important security analysis results that can provide countermeasures against an attack. Our results show that a narrow timeout during the data exchange protocol, high-data volume regarding the size of packets exchanged and a high packet

[2]Adopted in the Samsung Galaxy Nexus S Phone

error rate on the adversarial channel can significantly reduce the probability of a relay attack.

On top of our probabilistic model checking approach, the proposed framework provides a fast, automated and highly accurate methodology for protocol and hardware designers to evaluate their prototypes at the design stage, revealing to them whether or not certain security requirements will be met by the final product. Additionally, our work is motivational towards innovative countermeasures against attacks. The latter is a vital evidence that the proposed approach can be applied in safety-critical software or hardware artifacts that have to comply to certain industrial standards where cyber-security attacks govern environment. For our future work, we plan to extend our analysis to cover a larger family of short range communication protocols and explore new methods to deploy efficient countermeasures. Furthermore we aim at initiating a new analysis approach based upon NFC-based real-hardware characteristics that will allow us to determine the trade-off between security provided and energy consumption on limited power devices that seek to complete secure ad-hoc connections.

References

[1] ECMA-340, Near Field Communication Interface and Protocol (nfcip-1), second edition (jun 2013).

[2] I. 18092:2013, Telecommunications and Information Exchange between Systems – Near Field Communication – Interface and Protocol (NFCIP-1) (2013).

[3] S. Almuairfi, P. Veeraraghavan, N. Chilamkurti, D. Park, Anonymous proximity mobile payment (APMP), Peer-to-Peer Networking and Applications 7 (4) (2014) 620–627.

[4] V. Coskun, K. Ok, B. Ozdenizci, Developing NFC Applications, John Wiley & Sons, Ltd, 2011.

[5] G. Madlmayr, J. Langer, C. Kantner, J. Scharinger, Nfc devices: Security and privacy, in: Proc. 3rd International Conference on Availability, Reliability and Security (ARES'08), 2008, pp. 642–647.

[6] S. Kang, J. Kim, M. Hong, Button-based method for the prevention of near field communication relay attacks, International Journal of Communication Systems.

[7] Google Wallet, https://wallet.google.com (oct 2013).
URL `https://wallet.google.com`

[8] Isis Mobile Wallet, https://www.paywithisis.com (oct 2013).
URL `https://www.paywithisis.com`

[9] G. P. Hancke, M. G. Kuhn, An rfid distance bounding protocol, in: Proceedings of the First International Conference on Security and Privacy for Emerging Areas in Communications Networks, SECURECOMM '05, IEEE Computer Society, Washington, DC, USA, 2005, pp. 67–73.

[10] S. Drimer, S. J. Murdoch, Keep your enemies close: distance bounding against smartcard relay attacks, in: Proceedings of 16th USENIX Security Symposium on USENIX Security Symposium, SS'07, 2007, pp. 7:1–7:16.

[11] L. Francis, G. Hancke, K. Mayes, K. Markantonakis, Practical relay attack on contactless transactions by using nfc mobile phones, Cryptology ePrint Archive, Report 2011/618, `http://eprint.iacr.org/2011/618` (2011).

[12] S. Basagiannis, P. Katsaros, A. Pombortsis, Synthesis of attack actions using model checking for the verification of security protocols, Security and Communication Networks 4 (2) (2011) 147–161.

[13] S. Basagiannis, P. Katsaros, A. Pombortsis, N. Alexiou, A probabilistic attacker model for quantitative verification of dos security threat, in: Computer Software and Applications, 2008. COMPSAC'08. 32nd Annual IEEE International, IEEE, 2008, pp. 12–19.

[14] I. Paparrizos, S. Basagiannis, S. Petridou, Quantitative analysis for authentication of low-cost rfid tags, in: Local Computer Networks (LCN), 2011 IEEE 36th Conference on, IEEE, 2011, pp. 295–298.

[15] M. Kwiatkowska, G. Norman, D. Parker, Prism 4.0: Verification of probabilistic real-time systems, in: G. Gopalakrishnan, S. Qadeer (Eds.), Proc. 23rd International Conference on Computer Aided Verification (CAV'11), Vol. 6806 of LNCS, Springer, 2011, pp. 585–591.

[16] Marketsandmarkets.com, Near field communication (nfc) market: Global forecast & analysis (2011 - 2016) - products (nfc chip, micro sd card, integrated sim, reader & middleware), applications (mobile payment, ticketing, booking, data sharing, access control, non-payment, infotainment, advertisement) (jun 2013).

[17] J. Rebello, Press release: Us wireless carriers partner with big credit card companies, boosting cell phone nfc market (may 2011).

[18] Z. Kfir, A. Wool, Picking virtual pockets using relay attacks on contactless smartcard, in: Security and Privacy for Emerging Areas in Communications Networks, 2005. SecureComm 2005. First International Conference on, 2005, pp. 47–58.

[19] G. P. Hancke, A practical relay attack on iso 14443 proximity cards, Technical report, University of Cambridge Computer Laboratory 59 (2005) 382–385.

[20] W. Issovits, M. Hutter, Weaknesses of the iso/iec 14443 protocol regarding relay attacks, in: RFID-TA, 2011, pp. 335–342.

[21] L. Francis, G. P. Hancke, K. Mayes, K. Markantonakis, Practical NFC peer-to-peer relay attack using mobile phones, in: Radio Frequency Identification: Security and Privacy Issues - 6th International Workshop, RFIDSec 2010, Istanbul, Turkey, June 8-9, 2010, Revised Selected Papers, 2010, pp. 35–49.

[22] M. Roland, Software card emulation in nfc-enabled mobile phones: Great advantage or security nightmare?, in: 4th International Workshop on Security and Privacy in Spontaneous Interaction and Mobile Phone Use, 2012.

[23] S. Brands, D. Chaum, Distance-bounding protocols (extended abstract), in: EUROCRYPT'93, Lecture Notes in Computer Science 765, Springer-Verlag, 1993, pp. 344–359.

[24] C. J. F. Cremers, K. B. Rasmussen, B. Schmidt, S. Capkun, Distance hijacking attacks on distance bounding protocols, in: IEEE Symposium on Security and Privacy, IEEE Computer Society, 2012, pp. 113–127.

[25] A. Mitrokotsa, C. Onete, S. Vaudenay, Mafia fraud attack against the rc distance-bounding protocol, in: Proceedings of the 2012 IEEE RFID Technology and Applications (IEEE RFID T-A), IEEE Press, Nice, France, 2012, pp. 74–79.

[26] G. P. Hancke, M. G. Kuhn, Attacks on time-of-flight distance bounding channels, in: Proceedings of the First ACM Conference on Wireless Network Security, WiSec '08, ACM, New York, NY, USA, 2008, pp. 194–202.

[27] G. P. Hancke, Design of a secure distance-bounding channel for rfid, Elsevier, Journal of Network and Computer Applications 34 (3) (2011) 877–887.

[28] N. Alexiou, S. Basagiannis, S. Petridou, Security analysis of nfc relay attacks using probabilistic model checking, in: Proceedings of 10th International Wireless Communications and Mobile Computing Conference, IWCMC'14, IEEE, 2014.

[29] S. Basagiannis, S. Petridou, N. Alexiou, G. Papadimitriou, P. Katsaros, Quantitative analysis of a certified e-mail protocol in mobile environments: A probabilistic model checking approach, Computers & Security 30 (4) (2011) 257–272.

[30] S. Petridou, S. Basagiannis, M. Roumeliotis, Survivability analysis using probabilistic model checking: A study on wireless sensor networks, IEEE Systems 7 (1) (2013) 4–12.

[31] I. F. 14443, Identification Cards — Contactless Integrated Circuit(s) Cards — Proximity Cards (oct 2007).

[32] B. Wu, J. Chen, J. Wu, M. Cardei, A survey of attacks and countermeasures in mobile ad hoc networks, in: Wireless Network Security, Springer, 2007, pp. 103–135.

[33] J. H. Conway, On Numbers and Games, Academic Press, 1976.

[34] M. Kwiatkowska, G. Norman, D. Parker, Stochastic model checking, in: M. Bernardo, J. Hillston (Eds.), Formal Methods for the Design of Computer, Communication and Software Systems: Performance Evaluation (SFM'07), Vol. 4486 of LNCS (Tutorial Volume), Springer, 2007, pp. 220–270.

[35] W. Diffie, M. Hellman, New directions in cryptography, Information Theory, IEEE Transactions on 22 (6) (1976) 644–654.

[36] J. Vilela, P. Pinto, J. Barros, Jammer selection policies for secure wireless networks, in: Communications Workshops (ICC), 2011 IEEE International Conference on, 2011, pp. 1–6.

[37] S. Oh, T. Doo, T. Ko, J. Kwak, M. Hong, Countermeasure of nfc relay attack with jamming, in: Emerging Technologies for a Smarter World (CEWIT), 2015 12th International Conference Expo on, 2015, pp. 1–4.

[38] D. Kliazovich, M. Devetsikiotis, F. Granelli, Formal methods in cross layer modeling and optimization of wireless networks: State of the art and future directions, in: S. Kotsopoulos, K. Ioannou (Eds.), Heterogeneous Next Generation Networking: Innovations and Platforms, IDEA Group Inc., 2008, pp. 1–24.

[39] A. Scarfo, New security perspectives around byod, in: Broadband, Wireless Computing, Communication and Applications (BWCCA), 2012 Seventh International Conference on, 2012, pp. 446–451.

[40] A. R. N. U. .-. US20130231046 A1, Electronic device with shared near field communications and sensor structures (feb 2013).

[41] P. Staib, J. Helm, T. Renard, System and method of facilitating contactless payment transactions across different payment systems using a common mobile device acting as a stored value device, uS Patent App. 10/940,939 (2004).

[42] S. C. Alliance, The mobile payments and nfc landscape: A us perspective, a smart card alliance payments council white paper (2011).

[43] M. Pasquet, S. Gerbaix, The complexity of security studies in (nfc) payment system, in: Australian Information Security Managment Conference, 2010, pp. 95–101.

[44] A. Francillon, B. Danev, S. Capkun, Proceedings of the network and distributed system security symposium, NDSS 2011, San Diego, California, USA, 6th February - 9th February 2011, in: NDSS, 2011.

[45] NXP Semiconductors P5Cx009/P5Cx072 Secure triple, dual and contact PKI smart card controller, 0.1 edition (feb 2010).

[46] M. Roland, J. Langer, J. Scharinger, Practical attack scenarios on secure element-enabled mobile devices, in: Near Field Communication (NFC), 2012 4th International Workshop on, 2012, pp. 19–24.