IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 16, 2020, accepted October 6, 2020, date of publication October 19, 2020, date of current version October 29, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3031979

An ISO/IEC 7816-4 Application Layer Approach to Mitigate Relay Attacks on Near Field Communication

CHRISTINA THORPE [1,2], JOHN TOBIN [3], (Graduate Student Member, IEEE), AND LIAM MURPHY [3], (Member, IEEE)

[1] School of Informatics and Engineering, Technological University Dublin, Blanchardstown Campus, Dublin 15, D15 YV78 Ireland
[2] Department of Computing, Technological University Dublin, Blanchardstown Campus, Dublin 15, D15 YV78 Ireland
[3] School of Computer Science, University College Dublin, Dublin 4, D04 V1W8 Ireland

Corresponding author: Christina Thorpe (christina.thorpe@tudublin.ie)

This work was supported in part by the Science Foundation Ireland under Grant 13RC2094.

ABSTRACT Near Field Communication (NFC) has become prevalent in access control and contactless payment systems, however, there is evidence in the literature to suggest that the technology possesses numerous vulnerabilities. Contactless bank cards are becoming commonplace in society; while there are many benefits from the use of contactless payments, there are also security issues present that could be exploited by a malicious third party. The inherently short operating distance of NFC (typically about 4 cm) is often relied upon as a means of ensuring intentional interaction on the user's part and limiting attack vectors. However, NFC is particularly sensitive to relay attacks, which entirely negate the security usefulness of the short-range aspect of technology. The aim of this article is to demonstrate how standard hardware can be used to exploit the technology to carry out a relay attack. Considering the risk that relay attacks pose, a countermeasure is proposed to mitigate this threat. Our countermeasure yields a 100% detection rate in experiments undertaken – in which over 10,000 contactless transactions were carried out on a range of different contactless cards and devices. In these experiments, there was a false positive rate of 0.38% – 0.86%. As little as 1 in every 250 transactions were falsely classified as being the subject of a relay attack and so the user experience was not significantly impacted. With our countermeasure implemented, transaction time was lengthened by only 0.22 seconds.

INDEX TERMS Near field communication, relay attack, security.

I. INTRODUCTION

NFC encompasses a set of close proximity wireless communication technologies that facilitate simple communication and data exchanges between coupled devices. The technology is found in a wide range of devices including smartphones, credit cards and other smartcards, interactive advertisement posters, and gaming figurines [1]. Among its many applications are contactless payment, access control, and automated fare collection. Launched in 2002, it has since become commonplace in modern society and is continually finding its way into an ever-increasing number of applications and devices. NFC's market penetration has largely been catalysed by the smartphone industry embedding the technology in their devices in recent years. Both Google and Apple, whose Android (86.8%) and iOS (13.2%) mobile operating systems collectively accounted for 100% of the smartphone OS market share in Q3 2018 [2], launched NFC-based contactless payment applications [3], [4] for their smartphone platforms. With industry adoption ever-increasing, it is prudent that the technology is secure.

Many of NFC's security-sensitive applications rely on the inherently short operating distance of the technology as a means of ensuring intentional interaction with other NFC devices and preventing eavesdropping and other attacks that wireless communication technologies typically are vulnerable to. The literature argues that reliance on this premise is unfounded. NFC technology has been shown to be susceptible to several attack vectors including relay attacks [5], [6], replay attacks [7], eavesdropping [8], and side-channel attacks [9]. Many instances of relay attacks on NFC have been published which is particularly concerning [10], [11]. A relay

The associate editor coordinating the review of this manuscript and approving it for publication was Mohamed Elhoseny.

attack entirely diminishes the security usefulness of the short-range aspect of NFC. In an attack of this kind, communication between reader and tag is intercepted by an attacker and the transmissions from one are passed to the other via the attacker. With an attacker using two devices to relay transmissions (one between the reader and attacker, and another between the tag and attacker) the range of the communication is only limited by the attacker's ability to form a communication channel between the two relay devices. Importantly this communication channel need not be over NFC, and could be over WiFi or wireless mobile telecommunication networks (3G/4G), for example. This effectively removes any security benefit from NFC devices normally having to be in close proximity of one another.

The main contributions in this article are as follows:

1) Implementation of a known relay attack on NFC to show how the technology can be exploited by a malicious third party.
2) Proposal of a novel countermeasure to prevent relay attacks at the application layer using the ISO/IEC 7816-4 protocol.
3) Definition of a formal model of the countermeasure and implementation of a realistic prototype.
4) Design and execution of rigorous empirical tests to validate the new countermeasure.

The remainder of this article is organised as follows: The technical background, Section II, provides the reader with a knowledge of the fundamental operating principles behind NFC technology and the technical standards applicable to it. Section III begins by explaining the concept of a relay attack and goes on to detail our implementation of one on NFC. In Section IV, we propose our countermeasure to mitigate the threat of relay attacks on NFC technology. An evaluation of the performance and effectiveness of our countermeasure, and the setting in which it was tested, can be found in Section V. A review of the literature in the field of relay attack countermeasures is provided in Section VI. Finally, Section VII concludes the findings of this paper and outlines potential lines of future research.

II. TECHNICAL BACKGROUND

NFC is a wireless communication technology closely related to the High Frequency (HF) subset of Radio Frequency Identification (RFID). Sharing many of the same characteristics, NFC also operates on the 13.56 MHz carrier frequency and accordingly has a short operating distance, generally less than 10 cm [12]. The driving force behind this new technology was the need for bi-directional peer-to-peer communication between devices, which HF-RFID lacked. NFC was designed to remain backwards compatible and, in modern usage, the term typically encompasses HF-RFID technology.

Contactless NFC devices can be broadly classified under one of two headings based on whether they have their own power source or not. Those that do are known as ''Active'' devices and have more functionality available to them; their counterpart, devices that do not have any internal power source and rely on an Active device to externally power them, are termed ''Passive'' devices. This gives rise to the two modes of communication possible in NFC, dictated by the devices used in a transaction. Active communication mode features two Active devices which use their own RF fields to communicate, each modulating their respective RF fields. Passive communication mode is used when communication is desired between an Active and a Passive device. The Active device always initiates the transaction by generating an RF field. The Passive device responds by modulating the Active device's RF field, a technique known as load modulation. Communication between two Passive devices is not possible as by definition neither device has an independent power source and could not generate an RF field. Within a transaction, the device which starts the communication is referred to as the ''Initiator'' and is always an Active device. The respondent, coined the ''Target'', may be either Active or Passive.

NFC is disparate in its method of operation in comparison to other widespread wireless communication technologies, such as the 802.11 (wireless LAN) and, more closely related, 802.15 (wireless PAN) families [13]. Typically these technologies transmit information over the air through the modulation of periodic radio waves that they generate. As not all NFC devices have an internal battery or fixed external power supply, independent radio wave generation cannot be assumed. The technology places an onus on Active devices to wirelessly power their Passive counterparts. This is achieved through inductive coupling whereby a change in the current flowing through the loop antenna of the Active device induces a voltage across the Passive device's antenna. The Biot-Savart law [14] provides that a flow of charges giving rise to a current across a loop correspondingly provokes a magnetic field. Changes in this current accordingly causes a change in the magnetic flux of the loop's magnetic field which, under Faraday's laws of electromagnetic induction [15], induces an electromotive force on a coupled loop. As this is a non-radiative method of wireless power transfer relying on changing magnetic fields, it's operation is limited to the near-field region of a generated RF field, which is where the technology derives its name from. The antennae used by NFC devices are too small to allow a standing wave to propagate on the carrier frequency which precludes the use of the far-field region of the RF field. Since the near-field is the only concern, NFC antennae are designed as loop-type inductors as opposed to conventional antennae.

Near Field Communication Interface and Protocol (NFCIP-1) is the governing standard for NFC technologies and is defined by the ISO/IEC 18092 and ECMA-340 technical standards. Bit rates of 106, 212, and 424 kbit/s are supported by the standard for both communication modes, though bit rates of up to 6,780 kbit/s are possible in active communication mode however modulation and bit coding are unspecified beyond 424 kbit/s. Manchester coding with a 10% Amplitude-Shift Keying (ASK) modulation scheme is used in all cases with the exception of Active devices transmitting at 106 kbit/s where a modified Miller coding with a 100% ASK modulation scheme is employed. NFC also encompasses Proximity Card technology (comprising of Proximity Coupling Device (PCD) and Proximity Integrated Circuit Card (PICC) devices), defined by the ISO/IEC 14443 family of standards, and Vicinity Card technology, defined by ISO/IEC 15693. The NFC Forum, a collaborative industry association created to advance, develop and bring about compliance with NFC standards among its members, has also released an additional 20 standards for the technology [16]. Application layer protocols can occasionally be vendor specific but more commonly the ISO/IEC 7816-4 application protocol is adopted. The main abstraction of this protocol is the Application Protocol Data Unit (APDU) communication unit, grouped into command and response categories.

III. RELAY ATTACK

A relay attack is analogous to the 'Chess Grandmaster' problem [17]. The problem presents a novice player, N, with little to no knowledge of the rules of the game, and two chess grandmasters, $G1$ and $G2$. Player N is tasked with defeating a chess grandmaster, though without ability to play the game let alone having any game strategy, the odds of him succeeding seem infinitesimally small. This does not deter N however and so he devises a plan. N challenges two chess grandmasters to play against him; they both accept the invitation. Both games have been set up so that they are played simultaneously and at the same venue with $G1$ to the left of N and $G2$ to his right. In the game against $G1$, N plays black, while against $G2$, N plays white. The game against $G1$ starts first and as white makes the first move, $G1$ plays his opening move. N then turns around and starts the game against $G2$, mimicking the move made by $G1$ from the first game of chess. $G2$ then makes a move in response to N's play. N then returns to the first game of chess and copies the move made by $G2$. He then waits for $G1$'s response before returning to the second game and mimicking $G1$'s move against $G2$ once again. This process continues and in doing so it essentially becomes a case of $G1$ playing against $G2$; however, as N has split the gameplay into two games in which he is the opponent in both, N will succeed in his task of defeating a chess grandmaster. In all cases, except for a stalemate, N is guaranteed to win against either $G1$ or $G2$ while conceding to the other. This scenario in fact depicts a classical example of a relay attack; N simply relays the moves of one chess grandmaster against the other.

Extrapolating this concept to modern communication technology, the idea behind a relay attack is that communication between two parties is intercepted by an attacker who breaks the direct communication path and relays messages to each party via the attacker's relay device(s). Quite often there is no other aspect to this attack – no message alteration, packet inspection, etc. Although simplistic, its usefulness to an attacker and, consequently, danger to an unwilling subject can't be understated. As a wireless communication technology with security applications such as contactless payment and access control, a relay attack poses a real danger to the security of the technology. NFC is particularly sensitive to relay attacks. The NFC forum states that as "NFC transmissions are short range", they are "inherently secure" [18] but there are instances, such as on a busy bus or train, where an attacker would be in range to engage in an NFC transaction with a victim's credit card. Both reader and card assume that when they are in close proximity of one another, a connection should be established over NFC. In the case of access control systems, this assumption goes a step further and assumes that access should be granted (if that tag normally has right of entry at that location). This is by design as NFC is a user-centric technology and it is believed that the "very short distance" of the technology implies that you must make an intentional action to interact with an application over NFC [18]. This does undoubtedly improve the user experience but at the cost of security, and arguably is more of a vulnerability than design feature. A successful implementation of a relay attack allows the attacker to use the victim's card to purchase something completely unbeknown to the victim.

A. RELAY ATTACK IMPLEMENTATION

The relay attack used in our study was developed on the Android platform using devices readily available. Lee's [19] app 'NFCProxy' served as the foundation for the software developed to facilitate this attack.

Figure 1 depicts an active relay attack. For the purposes of this example, a contactless payment system is the subject of the relay attack. In this scenario, the attacker is in possession of two smartphones b and c; the first b running NFCProxy in Relay mode (referred to as 'NFCRelay') and the second c running it in Proxy mode. In our execution of this attack we used an Asus Nexus 7 device for b and a Sony Xperia T smartphone for c. NFCRelay hosts a server on device b; the proxy device c opens a socket and connects to b when a Proximity Coupling Device (PCD) is detected over NFC. Both devices must be connected to the same wireless network.

The attack is executed as follows:

- Prior to approaching a PCD (e.g. a contactless payment terminal) with c, the relay device b is brought in range of an NFC-enabled credit/debit card a.
- NFCRelay will report detection of this card if an NFC connection is successfully established.
- At this point the attacker can proceed with paying for a purchase and approach the contactless payment terminal with the proxy device c.
- Data is exchanged between the credit card a and the PCD d via relay devices b and c over a WiFi connection.
- The transaction completes and the relay attack was successful.

An entire transaction performed through this relay attack setup takes on average 1 - 4 seconds to complete.

This relay attack operates on the application layer, forwarding APDUs rather than frames. An advantage of this attack style over ISO/IEC 14443 protocol relay attacks is

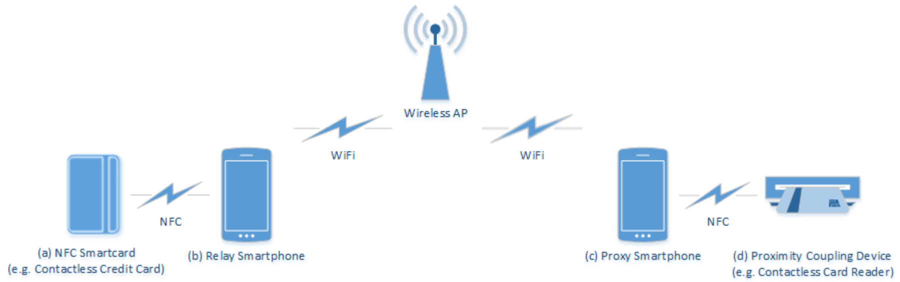

Wireless AP
WiFi
WiFi
NFC
NFC
(a) NFC Smartcard
(e.g. Contactless Credit Card)
(b) Relay Smartphone
(c) Proxy Smartphone
(d) Proximity Coupling Device
(e.g. Contactless Card Reader)

FIGURE 1. Illustration of relay attack.

that the proxy and relay devices could in theory issue wait time extension requests in their own right thus bypassing more stringent time constraints that can be imposed on lower layers. It also avoids Unique Identifier (UID) mismatches as the APDUs are encapsulated in a new ISO/IEC 14443 frame with the UID of the proxy or relay device that established the connection.

In theory this attack should be transferable to any Android device with an NFC chipset, however certain co-requisites exist that presently aren't available for all Android devices. The proxy device must run a version of CyanogenMod Android OS. In addition, the Android framework used by this version of CyanogenMod must contain the IsoPcdA class, to allow the proxy device to emulate an NFC-A smartcard. This class never featured in official Android releases and was subsequently removed from CyanogenMod to add support for Google Wallet. As such, a version of CyanogenMod 9 built between January 20th and March 22nd, 2012, is required for the proxy device. Unfortunately most devices nowadays, included those available to this project, never had such a build made for them as they weren't manufactured at that time. Similarly, many of the devices made in early 2012 didn't have NFC incorporated into them.

However, Android on its own initiative introduced Host-based Card Emulation (HCE) in late 2013 with Android 4.4 KitKat (API level 19) to allow for Android devices to emulate credit/debit cards and be used in their place in contactless payment system. HCE fulfils what we were trying to achieve with the IsoPcdA class – emulation of a smartcard. As such, we set out to modify NFCProxy to use Android's HCE service instead. The Sony Xperia T device was chosen for this purpose as the NFC driver for the chipset used in the Nexus 7 (2012 version) does not support Host Card Emulation, which is needed for the proxy device.

The NFC framework was largely overhauled in KitKat and many changes were introduced. The most significant change was the introduction of the HCE service; a background service that initially receives all APDUs before dispatching them to any service registered to receive ADPUs from the reader sending them. While this HCE service is useful for developers of NFC payment apps, it unexpectedly hindered development of our relay attack.

After exhausting other options, we decided that modifying the framework of an existing build in situ would be the most efficient approach. This was achieved by rooting a Sony Xperia T device running CyanogenMod 11 (Android version 4.3.1; Jelly Bean). Once rooted, the framework.jar and framework.odex files were pulled from /system/framework/. These files were deodexed using the Java SE Development Kit (JDK) and the dex2jar application. The classes.dex file was extracted from the resultant jar files. This file was deodexed as before and decompiled using Java Decompiler. The required changes were made to the IsoDep, IsoPcdA, IsoPcdB and TagTechnology classes. The classes file was re-assembled and odexed using JDK, and inserted back into the framework files. The framework files were then odexed as necessary and pushed to the device while it was in recovery mode, overwriting the old framework files. This process succeeded, allowing us to use NFCProxy and perform the relay attack.

B. RELAY ATTACK RESULTS

The additional delay on the system imposed by the relay attack was initially a concern. The overhead from processing on the relay devices and from the additional communication links could have caused timeouts, which would cause the relay attack to fail. However, this was not the case and the relay attack succeeded despite the extra delays. Hancke's [5] relay attack implementation attracted a 15 - 20us delay whereas our application of this attack amassed delays ranging from 14 ms to 68 ms, depending largely on WiFi network congestion. This is a considerable delay to introduce to a system; unexpectedly none of our attacks failed due to timeout and loss of synchronisation. This adds to the vulnerability of the technology.

We built a testbed consisting of a total of 7 debit and credit cards from three Irish financial institutions, in addition to a developer pack of different NFC tags and form factors purchased from RapidNFC. The payment terminals tested included contactless-enabled vending machines from bds Vending Solutions Ltd., an Ingenico iCT250 Point of Sale (POS) terminal, a ViVOTech ViVOpay 4000 contactless payment device, and a PN532 NFC controller breakout board simulating POS transactions. The relay attack succeeded in all instances. No protection from relay attacks was afforded by any of these devices.

IV. COUNTERMEASURE

We initially sought to frame our countermeasure on a two-pass authentication mechanism between the reader and the tag, in a manner similar to that defined in ISO/IEC 9798-2 [20]. Amongst the encrypted data exchanged would be a temporal parameter, a timestamp, used to benchmark the time taken for authentication of both parties. The time taken would then be used to establish whether the communication link was subject to a relay attack by detecting a higher RTT, however this metric was limited. This method presented problems. Sufficient clock synchronicity is unlikely achievable on low-cost passive NFC devices. Use of an asymmetric algorithm would likely prove easier to implement in devices currently in circulation but the resource greedy nature of asymmetric algorithms would mean processing time on tags would vary greatly, and thus the ability to detect a relay attack is reduced and implementation becomes specific to the device and form factor. It is apparent that an authentication-type countermeasure would not be practical in this instance.

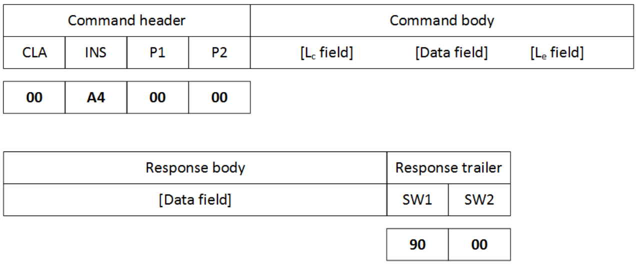
We propose a time measurement based countermeasure. Our countermeasure seeks to prevent relay attacks by detecting the delays that they impose on the system. This permits an upper bound to be established on the RTT of fixed size commands and responses, differentiating genuine transactions from relayed transactions. Hancke [5] discusses the difficulty faced by time checking countermeasures, caused by processing time variation and indeterminate data transfer delays, however we argue that such variance can be bounded and sufficiently accounted for to defend against common relay attack setups. Indeed, relay attacks on NFC have evolved from requiring technical expertise and custom-built hardware to needing little more than a foundation level knowledge of a mobile operating system and having access to common, low-cost NFC-equipped smartphones. Consequently, the user base capable of performing relay attacks on NFC devices has grown from a handful of academics to a substantial cohort of the general population, and correspondingly so too has the potential for malicious use. It is this smartphone relay attack implementation scenario that we believe presents the greatest danger to the technology, ergo our countermeasure focuses on mitigating this threat scenario.

This countermeasure differentiates itself from other proposed time-based countermeasures by its implementation at the application layer. Previous solutions have focused on the transmission protocol – defined by ISO/IEC 14443-4 [21]. Instead this approach operates on the ISO/IEC 7816-4 [22] application layer protocol which is used by the majority of NFC Forum Type 4 tags. While this makes our countermeasure unsuitable for simple NDEF record-holding tags (e.g. NTAG203, Mifare Ultralight), these tags are typically not used in security sensitive applications where relay attacks pose a significant threat. Rather, it is Type 4 tag technology that is used for contactless payment, access control and other security-demanding applications. By integrating our countermeasure further up the stack, its scope is naturally reduced; however, this is offset by greater portability and ease in deployment. Another significant reason for designing the countermeasure around the ISO/IEC 7816-4 application protocol is that the Android NFC API (in addition to the Windows Phone Smartcard API and Apple Pay API) allows for communication and message encapsulation solely via this protocol. ISO/IEC 14443 protocol procedures are handled exclusively by the device's embedded NFC controller with a limited controller interface available to the Android operating system kernel. Furthermore, EMV technical standards for contactless payments adopt the ISO/IEC 7816-4 protocol for data exchange.

By measuring the RTT taken by a sufficient sample of successive APDUs, many common relay attack implementations can be detected and prevented. The APDU used must have specific attributes, which we will discuss in more detail. The RTT measurement phase takes place after an NFC connection has been established and the initialisation sequence is complete, and before transaction-related data is exchanged. While the RTT taken differs with tag type and form factor, our test results show that the delay caused by even a dedicated wireless communication channel with no other traffic is substantial enough to build a generalised model with a high detection rate, which will be discussed further in Section V.

The ISO/IEC 7816-4 protocol specifies the APDU structure and defines 39 APDU commands. Of these, the NFC Forum specifies that only 3 of these must be supported by NFC Forum Type 4 devices, leaving support for the remainder as optional and at device manufacturers' discretion [23]. The three mandatory commands consist of SELECT (for the selection of files or applications), READ BINARY (for reading data from a file), and UPDATE BINARY (for updating of data in or to a file). For our countermeasure we required a small, fixed-size command APDU (C-APDU) that was universally supported and that added little to no processing on the recipient tag. Furthermore, the response APDU (R-APDU) must also be fixed in size and of small size to curtail discrepancies due to data transfer delays. Accordingly, we opted for our command APDU to be a SELECT command for the tag to select the Master File record on that tag. This is similar to selecting the root directory of a file system and is not processing intensive. Furthermore, after the initial selection any further command APDUs to select the same file record incur a commensurate processing time allowing us to safely attribute any variances to the communication link.

The structure of our command and response pair can be seen in Figure 2. Notably, neither use their optional data field for SELECT commands. The value of the response trailer for the R-APDU may differ among devices, with some returning "6A 86" instead. While this indicates incorrect parameters in the C-APDU, it in fact has no bearing on our countermeasure and can marginally reduce processing time which is a beneficial consequence. The C-APDU and R-APDU are 4 bytes and 2 bytes, respectively, at the application layer.



Command header				Command body		
CLA	INS	P1	P2	[L_c field]	[Data field]	[L_e field]
00	**A4**	**00**	**00**			

Response body	Response trailer	
[Data field]	SW1	SW2
	90	**00**

FIGURE 2. Command and response pair structure.

Depending on parameters selected at initialisation, the frame size is between 3 and 5 bytes larger for both when overhead from lower layers is accounted for. At the default bit rate of 106 kbit/s, this results in a theoretical transmission time of 528 – 679 μs for the command, and 377 – 528 μs for its response; an acceptably small time for delays incurred by a wireless network used as a relay channel, for example, to be detectable.

Algorithm 1 Relay Attack Countermeasure Strategy

```
1:  $int\ i \leftarrow 0,\ sum \leftarrow 0,\ sumdev \leftarrow 0$
2:  $int\ rApdu,\ cApdu \leftarrow$ “\x00\xA4\x00\x00”
3:  $int[N]\ rtt$
4:  $float\ mean,\ stddev$
5:
6:  while $i < N$ do
7:      Start $timer_i$
8:      $rApdu \leftarrow send(cApdu)$
9:      End $timer_i$
10:     $rtt[i] \leftarrow timer_i$
11:     $i{+}{+}$
12:
13: for $i \leftarrow 1, i{+}{+}$, while $i < N$ do
14:     $sum\ rtt[i]$
15:
16:     if $i = N - 1$ then
17:         $mean \leftarrow sum/(N-1)$
18:
19: for $i \leftarrow 1, i{+}{+}$, while $i < N$ do
20:     $sumdev\ (rtt[i] - mean)^2$
21:
22:     if $i = N - 1$ then
23:         $stddev \leftarrow \sqrt{sumdev/(N-1)}$
24:
25: if $mean > T_{MAX}$ then
26:     End transaction
27: else if $stddev > STDDEV_{MAX}$ then
28:     End transaction
29: else
30:     Transaction continues
```

Algorithm 1 shows the protocol for our countermeasure. The initialisation sequence of the NFC devices has been omitted as this is handled by the NFC controller and is immaterial to this algorithm. Values for T_{MAX}, the maximum permissible mean RTT, and $STDDEV_{MAX}$ are imperative to the functionality of the protocol and are discussed in Section V. These variables represent the upper bounds established to distinguish between genuine systems and those subject to a relay attack. This protocol should commence immediately after activation of the transmission protocol has completed. The countermeasure algorithm begins by gathering timing samples of the round trip time taken for the PCD to send the chosen C-APDU and receive the R-APDU (lines 6 – 12). The mean (lines 14 – 20) and standard deviation (lines 22 – 28) of the RTT sample are then computed. Note that the first RTT value obtained is omitted in these calculations as our tests have consistently found that the first timing tends to be an outlier and unrepresentative of the remainder of the sample taken. If either the mean RTT or the standard deviation between the RTT values obtained exceeds these threshold values, the connection is dropped before application data is exchanged (lines 30 – 33). Else, the transaction may proceed and is handed off to the application supported by the devices as no relay attack has been detected. The risk presented by relay attacks is thus mitigated when this countermeasure is implemented as described.

V. VALIDATION

With NFC now being integrated into most modern smartphones, the equipment required to perform a relay attack on the technology is readily accessible. It is this particular threat scenario that this countermeasure attempts to protect against. As such, this countermeasure was built around the Android OS framework and designed to defend against the relay attack implementation described in Section III. Results presented in this section were obtained from tests carried out on such a relay attack setup. An Adafruit PN532 NFC controller board was used as the Active device and Passive devices tested included a DESFIRE 4k access card, MIFARE credit cards from two financial institutions, a MIFARE automated fare collection card, and an Irish passport.

Taking the C-APDU/R-APDU pair as discussed in Section IV, we examined the round-trip time incurred by sending the C-APDU from the PCD to the contactless card, processing the command on the card, and replying with the R-APDU from the contactless card to the PCD over NFC. Experiments to investigate this were undertaken for both uninterrupted and relay attack setups. We discovered from 10,000 tests on each card that there is a considerable difference in the RTT of a genuine, uninterrupted transaction over NFC and one relayed over a dedicated WiFi network with no other traffic. In addition, we found that the RTT varies only slightly in genuine transactions, with a standard deviation of 112 – 267 μs depending on the card tested, while relayed transactions deviate wildly from the mean, with a standard deviation in the range of 10,473 – 22,314 μs in our tests, as can be seen in Figure 3. From these results a two-part algorithm based on mean RTT values and their standard deviation was devised.

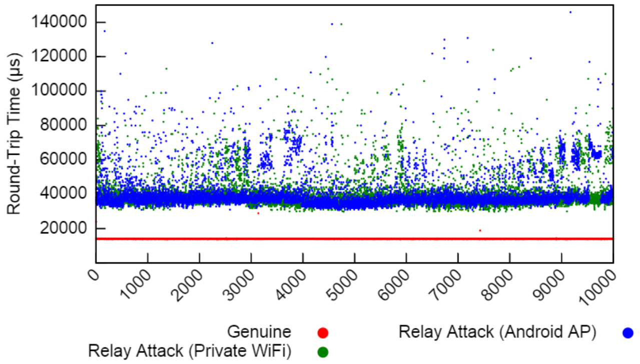


FIGURE 3. RTT for countermeasure command-response protocol in both uninterrupted and relay attack setups for 10,000 samples.

Finding suitable upper limits, in order to distinguish between genuine and relayed transactions, for both the mean and standard deviation of a sample of RTT values remains a challenge. This is predominantly due to the variety of different form factors, chipsets and antennae used by NFC devices which can all effect the typical RTT of samples during the measurement phase of our countermeasure algorithm. An ideal solution may take the form of a database of standard RTT values for each NFC device, or a hard-coded value in the memory of passive NFC devices representing an acceptable upper limit on the RTT. As the difference in the RTT for genuine and relayed transactions was found to be in the order of tens of milliseconds, a more realistic solution presented itself in the form of establishing a generalised upper bound, as implemented in our algorithm. Values for T_{MAX} and $STDDEV_{MAX}$ were fixed at 20 ms and 500 μs, respectively. 20 ms was selected for T_{MAX} as it is greater than the mean RTT plus three standard deviations (14.58 ms) that we encountered for a genuine transaction, yet lower than the minimum RTT (23.97 ms) for a relayed transaction, and allows for some padding. $STDDEV_{MAX}$ was set at 500 μs for similar reasons and lies between the standard deviation of genuine transactions (mean: 175.56 μs) and relayed transactions (mean: 14,281.33 μs).

With these upper bounds in place and the countermeasure implemented as described in Algorithm 1, we initiated 10,000 transactions for each of our test devices in uninterrupted and relayed systems. Our countermeasure yielded a 100% detection rate in these tests and had a false positive rate of 0.38% – 0.86%. These results suggest that a time-based countermeasure to relay attacks is viable, at least in the circumstances we examined. Less than 1 in every 100 transactions were falsely classified as being the subject of a relay attack and so the user experience was not significantly impacted. Figure 4 shows the vast majority of transactions fall well below T_{MAX} and how the typical RTT can vary between different passive NFC devices.

An important consideration in implementing this countermeasure was determining what constitutes a sufficient sample for the amount of RTT values obtained. Ease of use and expedient transactions are core values of NFC. With a sample size too large, the temporal overhead incurred by sending and receiving APDUs during the measurement phase will degrade the user experience. If the sample size is too small, the countermeasure may fail to detect a relay attack. Over 10,000 iterations we tested our countermeasure algorithm with sample sizes of 10, 15, 20, 25, 30, 40, and 50. RTT per command and response pair was in the range of 11 – 14 ms, depending on the Passive device. A sample size as low as 10 was found to be satisfactory in most cases, however when accounting for human interaction a sample size of 20 allowed for a tolerance of outliers caused by user movement to or from the PCD. This lengthened transactions by 0.22 seconds.

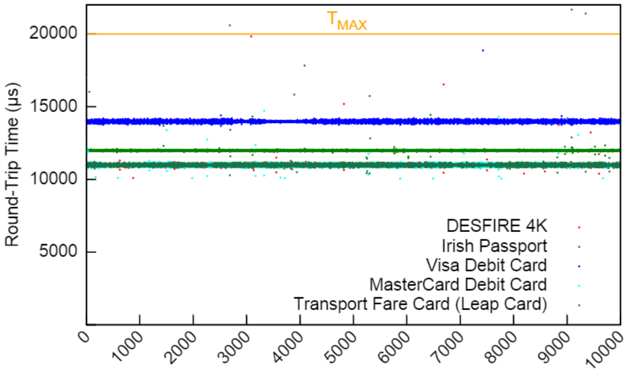


FIGURE 4. RTT for countermeasure command-response protocol with genuine transactions for 10,000 samples and 5 NFC devices. T_{MAX}, the countermeasure's upper bound for the mean RTT, is also shown.

VI. RELATED WORK

This section will provide a review of the literature and discuss other possible countermeasures to our relay attack implementation. Countermeasures can be broadly divided into two categories based on their principle mechanism of action – that is to say either physical or logical.

A. PHYSICAL APPROACHES

The biggest challenge an attacker faces when seeking to perform a relay attack is gaining access to the victim's credit/debit card (similarly this applies to other NFC-based smartcards and tags). An attacker doesn't necessarily need to be within the typical reading distance of the card; with the right equipment, and in particular the right antenna [24], the attack can reach a card from nearly half a meter away, if not further. Most generic wallets and purses offer little-to-no protection from relay attacks and do not block signal propagation in a significant way. However some materials such as aluminium do block NFC communications and, accordingly, could prevent an attacker from reading a contactless card. A simple, yet effective, method would be to shield contactless cards when not being used. While it is unlikely anyone would want to wrap their credit cards in aluminium foil, a wallet or card holder with an embedded metallic mesh would shield cards from being read by an attacker. Such wallets already exist for RFID [25] and these RFID-blocking wallets also work for NFC. They operate on the principle that a Faraday cage is created around the card which causes any external electric field to dissipate across the embedded mesh, and thus preventing it from reaching the card. Lining the card slots of ones wallet with aluminium foil works equally well however; in fact only a thickness of 27 microns of aluminium is needed to function as a barrier [26].

As the use of shielding is user-dependent, a physical authentication mechanism introduced at the manufacturing stage is favoured. The most basic implementation of this may be to disable the NFC module in contactless cards by default and to introduce a button, which when pressed and held would enable the NFC module. More advanced solutions take the form of requiring the user to enter a PIN code on such a card to authenticate themselves and enable card functionality. In 2012, MasterCard© launched a new line of credit cards, named 'Display Cards' [27], which feature an embedded LCD display and touch-sensitive capacitive buttons ranging from 0 to 9 in addition to 'Ok', 'Cancel', and 'On/Off' buttons. While early models did not feature a contactless interface, newer models have been released with an embedded NFC controller. MasterCard's Display Card technology is primarily aimed towards authentication for online banking services but it would serve equally well as a two-factor authentication mechanism for NFC transactions.

The fingerprinting of NFC devices based on modulation patterns and spectral characteristics could be used to prevent relay attacks. Consider a case where NFC tags are assessed at the manufacturing stage and features specific to the individual tag's waveform are extracted. These identifiers are then used to build a fingerprint unique to the tag which is digitally signed and stored in a locked region of the tag's memory. When the tag is presented to a reader, the reader queries the tag for its fingerprint and then performs its own analysis of the tag's waveform using the same process as the manufacturer, thus generating its own fingerprint for the device. If both fingerprints match, the transaction proceeds. In the event of a relay attack, the fingerprints would fail to match as the reader would have generated a fingerprint for the proxy device instead. The feasibility of such a countermeasure is premised on individual tags varying enough from the manufacturing process to allow for the generation of unique fingerprints, and on readers being able to accurately derive these fingerprints. The work of Danev *et al.* [28] on the fingerprinting of HF-RFID devices shows that this is in fact achievable. Using their identification technique over 50 smart cards of the same type and manufacturer, and operating on the 13.56 MHz frequency, they attained an Equal Error Rate of 2.43% with a single run and 4.38% over two runs. The size of their generated fingerprint is approximately 120 bytes and could fit on the vast majority of NFC tags. We have some considerations regarding the use of such a technique in practice, namely the unlikeliness of tags to be kept fixed with regard to the acquisition antenna of the reader when being measured and how this might affect the accuracy of fingerprint generation. We also regard 2 seconds for feature extraction as too significant of an impact on the user experience for this technique to be deployed as is, however we share the authors' opinion that performance could be significantly improved with a more efficient hardware implementation. We believe fingerprinting countermeasures for NFC warrants further investigation and we intend to examine these in future work.

Although effective, a physical counter measure isn't always practical. The range of RFID blocking wallets and purses available is limited and big fashion labels don't incorporate this feature into their products. Likewise, lining card holders with aluminium foil is displeasing to the eye and lessens the user experience, which is one of the big attractions of NFC technology. We also can't discount the fact that users must remove their card at some point to use it; at which point the card will be exposed and vulnerable to relay attacks. Similarly, any on/off button to enable the NFC module would still leave a window of time during a transaction whereby the card is susceptible to attack, albeit brief. With our relay attack implementation taking less than 4 seconds to complete a transaction, a card need not be exposed for long to be exploited by a malicious attacker. The promise shown by MasterCard's Display Card is similarly negated by its lack of adoption by financial institutions, with interested therein waning. A significant problem faced by any physical integration of a means for authentication is that it increases transaction time and thus abates the user experience. For these reasons a logical counter measure is more appropriate.

B. LOGICAL APPROACHES

A review of the literature discovered limited success in preventing relay attacks on NFC and related technologies. The counter measures proposed focus on the changes introduced to a system by a relay attack — distance and time — and seek to measure them in order to differentiate a normal transaction from one under a relay attack.

Hancke and Kuhn [29] devised a distance-bounding protocol for RFID systems aimed at simple, low-power hardware, akin to that of Passive NFC tags. The cryptographic protocol works on the premise of the Round-Trip Time (RTT) of signals being accurately calculated, but requires a high bandwidth channel. They acknowledge that their distance-bounding protocol would be unsuitable for technologies operating on the 13.56 MHz carrier frequency due its low data bandwidth in the magnitude of 300 KHz. For the ISO/IEC 14443-A standard, and thus most NFC communications, their system could only resolve distance to one kilometer rendering it only partially effective against our relay attack. Brands and Chaum [30], pioneers in the field of distance bounding, describe the original distance-bounding protocol with a lower false acceptance rate, but it notably lacks the noise resilience of that of Hancke and Kuhn, and a practical implementation was never proposed by the authors, much less for NFC. More recent protocols, such as Fischlin and Onete's [31], have brought us closer to a usable distance bounding solution however there is still much work to be done in this field and many challenges to overcome before distance-bounding protocols become a practical countermeasure to relay attacks.

Boureanu and Vaudenay [32] highlight possible integration solutions for distance-bounding protocols in typical RFID and NFC applications. They hold that for contactless payment systems, for example, a solution lies in public-key distance bounding – of which there are only two such protocols in the literature. Of these two protocols, neither afford protection against terrorist fraud attacks, which are a form of relay attack that attempts to misrepresent the distance between the legitimate parties in the transaction. Further work in this field towards public-key distance-bounding protocols has the potential to produce an effective countermeasure to relay attacks in NFC. It should also be mentioned that there is no provision in the standards allowing for a distance-bounding protocol and thus any solution of this nature would be incompatible with current standards [33], [34].

Alhothaily *et al.* [35] propose a multi-factor verification approach to prevent various attacks by using multi-possession factor authentication with distance bounding in the verification process. This requires the user to use a bank card and a personal device to verify a transaction. This approach requires additional configuration, extra hardware, and results in a longer transaction time.

Hu *et al.* in [36] explored methods for the detection of wormhole attacks on wireless ad-hoc networks. A wormhole attack is synonymous with a relay attack in conceptual terms but can be more broadly varied in actual execution; its use is more predominant in the literature for wireless sensor networks. Their solution to this threat was to add information to packets being transmitted such that this information could be used to impose an upper bound on the transmission distance allowed for the packet. This was coined a ''packet leash'' and two types were described – geographical leashes and temporal leashes. Construction of geographical leashes require nodes to know their own location, thus making the construct inapplicable to most NFC technology implementations. Certain NFC-enabled mobile phones with location services may pose as candidates for use of this detection mechanism when being used in peer-to-peer mode. Temporal leashes require nodes to have tightly synchronised clocks with the maximum permissible deviation between each node's clock being in the scale of hundreds of nanoseconds to a few microseconds. Achieving this level of clock synchronicity falls outside of the capabilities of common NFC devices. It is possible that this clock synchronisation accuracy may be achievable on NFC devices with an integrated Global Positioning System (GPS) [37]. In practice, it is our opinion that a counter measure to NFC relay attacks based on packet leashes is not feasible due to the limitations of NFC devices in circulation.

Weiß [34] proposes a solution based on time measurement. They note that relay attacks impose additional delays on RTT during the challenge-response protocol and suggest that vendors determine an average RTT value for their application during this protocol. From this, vendors can then set a maximum threshold at which they decline to proceed with communication if exceeded. Although this is effective against some relay attacks, and successfully prevented Weiß's relay attack implementation, it is overly specific. Its implementation on the ISO/IEC 14443 protocol stack presents possible standards compliance issues. This countermeasure also fails to account for design variances among different NFC tags and smartcards, and how these variances can affect response times. This issue is compounded by the use of AES encryption in the challenge-response protocol which introduces processing time variation as a new unconsidered variable. Furthermore, Weiß acknowledges that each terminal system would have to be evaluated and adjusted separately which would incur a disproportionate effort to implement on a large-scale. Therefore, we do not consider this counter measure a viable solution in its current form.

In comparison, our countermeasure is compliant with the standards by being implemented at the application layer. Processing time was accounted for in all of our tests and we have accounted for variances that may occur in a practical setting. The effectiveness of our countermeasure is comparable with Weiß and we believe that our proposal could be deployed even without individual terminal evaluation.

VII. CONCLUSION

A low operating range may be seen as advantageous to a wireless communication technology used for security sensitive applications but it should not be relied on as a security mechanism in its own right. With NFC this is too often the case despite many documented relay attacks on the technology. The problem is compounded by technical standards that fail to provide adequate protection to the communication channel and require revision to add support for distance-bounding protocols. We successfully implemented a relay attack over a WiFi network using inexpensive, readily available Android smartphones. As vulnerabilities in contactless payment systems would have serious implications, we subjected these systems to our relay attack. The relay attack succeeded, further highlighting that NFC is not an appropriate wireless communication technology for security critical systems.

In an effort to mitigate the threat posed by this kind of attack, we proposed an application-layer countermeasure that aims to detect overhead added by relaying communications over an additional communication channel. This countermeasure succeeded with a detection rate of 100% for the 50,000 instances it was tested over. While not all relay attack implementations will cause sufficient overhead to be detected, it is apparent that relay attacks on NFC over WiFi are detectable. In future work on this subject, we intend to investigate the applicability of our countermeasure to other common relay communication channels, such as Bluetooth. A trade-off to our approach is the additional time required to process a transaction, however, in our experiments the average delay was 0.22 seconds, which would have a negligible impact on the user experience of contactless payment or access control.

REFERENCES

[1] H. Gilbert. (2014). *Amiibos Explained–The How and Why of Nintendo's New Toys*. Accessed: Apr. 4, 2015. [Online]. Available: http://www.gamesradar.com/amiibos-explained-how-and-why-nintendos-new-toys/

[2] International Data Corporation (IDC). (2015). *Worldwide Quarterly Mobile Phone Tracker*. Framingham, Massachusetts. [Accessed 27 Sep. 2015]. Accessed: Sep. 27, 2015. [Online]. Available: http://www.idc.com/prodserv/smartphone-os-market-share.jsp

[3] Apple Inc. (2015). *Apple Pay*. [Online]. Available: http://www.apple.com/apple-pay/

[4] Google Inc. (2011). *Google Wallet*. [Online]. Available: https://wallet.google.com/

[5] G. P. Hancke, "A practical relay attack on ISO 14443 proximity cards," Univ. Cambridge Comput. Lab., Cambridge, U.K., Tech. Rep., 2005, pp. 382–385, vol. 59.

[6] W. Issovits and M. Hutter, "Weaknesses of the ISO/IEC 14443 protocol regarding relay attacks," in *Proc. IEEE Int. Conf. RFID-Technol. Appl.*, Barcelona, Spain, Sep. 2011, pp. 335–342.

[7] Z. Kfir and A. Wool, "Picking virtual pockets using relay attacks on contactless smartcard," in *Proc. 1st Int. Conf. Secur. Privacy Emerg. Areas Commun. Netw. (SECURECOMM)*, Sep. 2005, pp. 47–58.

[8] G. P. Hancke, "Eavesdropping attacks on high-frequency RFID tokens," in *Proc. 4th Workshop RFID Secur. (RFIDSec)*, Budapest, Hungary, 2008, pp. 100–113.

[9] T. Kasper, D. Oswald, and C. Paar, "New methods for cost-effective side-channel attacks on cryptographic RFIDs," in *Proc. 5th Workshop RFID Secur. (RFIDSec)*, Leuven, Belgium, 2009, pp. 1–15.

[10] L. Francis, G. Hancke, K. Mayes, and K. Markantonakis, "Practical relay attack on contactless transactions by using NFC mobile phones," in *Proc. Workshop RFID Secur. (RFIDSec Asia)*, Tapei, Taiwan, 2012, pp. 1–16.

[11] R. Verdult and F. Kooman, "Practical attacks on NFC enabled cell phones," in *Proc. 3rd Int. Workshop Near Field Commun.*, Feb. 2011, pp. 77–82.

[12] E. Haselsteiner and K. Breitfuß, "Security in near field communication (NFC)," in *Proc. Workshop RFID Secur. RFIDSec*, 2006, pp. 12–14.

[13] J. M. Tjensvold. (2007). *Comparison of the IEEE 802.11, 802.15.1, 802.15.4 and 802.15.6 Wireless Standards*. Accessed: Oct. 19, 2020. [Online]. Available: https://janmagnet.files.wordpress.com/2008/07/comparison-ieee-802-standards.pdf

[14] F. Miller, A. Vandome, and J. McBrewster, *Biot-Savart Law*. New York, NY, USA: Alphascript Publishing, 2009.

[15] C. R. Nave. (1998). *Faraday's Law*. Accessed: Apr. 3, 2015. [Online]. Available: http://hyperphysics.phy-astr.gsu.edu/hbase/electric/farlaw.html

[16] NFC Forum. *NFC Forum Technical Standards*. Accessed: Apr. 4, 2015. [Online]. Available: http://nfc-forum.org/our-work/specifications-and-application-documents/ specifications/nfc-forum-technical-specifications/

[17] J. H. Conway, *On Numbers and Games*, 2nd ed. Natick, MA, USA: A K Peter, 2000.

[18] NFC Forum. *NFC in Action*. Accessed: Apr. 4, 2015. [Online]. Available: http://nfc-forum.org/what-is-nfc/nfc-in-action/

[19] E. Lee, "NFC hacking: The easy way," in *Proc. DEF CON*, Las Vegas, NV, USA, 2012, pp. 1–29.

[20] International Organization for Standardization/International Electrotechnical Commission, *Information Technology–Security Techniques–Entity Authentication—Part 2: Mechanisms Using Symmetric Encipherment Algorithms*, 2nd ed., Cham, Switzerland, Standard JTC 1, ISO/IEC 9798-2, 1999.

[21] *International Organization for Standardization/International Electrotechnical Commission Identification Cards–Contactless Integrated Circuit(S) Cards–Proximity Cards—Part 4: Transmission Protocol*, 2nd ed., Cham, Switzerland, Standard JTC 1/SC 17, ISO/IEC 14443-4, 2008.

[22] *Identification Cards–Integrated Circuit Cards—Part 4: Organization, Security and Commands for Interchange*, 3rd ed., Cham, Switzerland, Standard ISO/IEC 7816-4, 2013.

[23] N. Forum, "Type 4 tag operation [T4TOP]," Wakefield, MA, USA, Tech. Rep. NFC Forum, T4TOP 2.0, NFCForum-TS-Type-4-Tag_2.0 2011-06-28, 2014.

[24] P. Sorrells, "Optimizing read range in RFID systems," *EDN-Boston Denver*, vol. 45, no. 25, pp. 173–184, 2000.

[25] DIFRwear. *RFID Blocking Wallet*. Accessed: Apr. 5, 2015. [Online]. Available: http://www.difrwear.com/

[26] S. Garfinkel and H. Holtzman, "Understanding RFID technology," in *RFID: Applications, Security, and Privacy*, S. Garfinkel and B. Rosenburg, Eds. Indianapolis, IN, USA: Addison-Wesley, 2005, p. 26.

[27] M. Worldwide. (Nov. 2012). *MasterCard Introduces Next Generation 'Display Card' Technology, a first for Singapore*. Press Release, Singapore. Accessed: Aug. 24, 2015. [Online]. Available: http://tinyurl.com/mastercardNFC

[28] B. Danev, T. S. Heydt-Benjamin, and S. Capkun, "Physical-layer identification of RFID devices," in *Proc. USENIX Secur. Symp.*, 2009, pp. 199–214.

[29] G. P. Hancke and M. G. Kuhn, "An RFID distance bounding protocol," in *Proc. 1st Int. Conf. Secur. Privacy Emerg. Areas Commun. Netw. (SECURECOMM)*, Athens, Greece, Sep. 2005, pp. 67–73.

[30] S. Brands and D. Chaum, "Distance-bounding protocols," in *Proc. Workshop Theory Appl. Cryptograph. Techn.* Springer, 1993, pp. 344–359.

[31] M. Fischlin and C. Onete, "Terrorism in distance bounding: modeling terrorist-fraud resistance," in *Proc. Appl. Cryptogr. Netw. Secur.* Banff, AB, Canada: Springer, Jun. 2013, pp. 414–431.

[32] I. Boureanu and S. Vaudenay, "Challenges in distance bounding," *IEEE Secur. Privacy*, vol. 13, no. 1, pp. 41–48, Jan. 2015.

[33] *International Organization for Standardization/International Electrotechnical Commission Information Technology–Telecommunications and Information Exchange Between Systems–Near Field Communication–Interface and Protocol (NFCIP-1)*, 2nd ed., Cham, Switzerland, Standard JTC 1, ISO/IEC 18092, 2013.

[34] M. Weiß, "Performing relay attacks on ISO 14443 contactless smart cards using NFC mobile equipment," M.S. thesis, Der Technischen Universität München, Munich, Germany, 2010.

[35] A. Alhothaily, A. Alrawais, X. Cheng, and R. Bie, "A novel verification method for payment card systems," *Pers. Ubiquitous Comput.*, vol. 19, no. 7, pp. 1145–1156, Oct. 2015.

[36] Y.-C. Hu, A. Perrig, and D. B. Johnson, "Wormhole attacks in wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 2, pp. 370–380, Feb. 2006.

[37] B. Sterzbach, "GPS-based clock synchronization in a mobile, distributed real-time system," *Real-Time Syst.*, vol. 12, no. 1, pp. 63–75, 1997.

CHRISTINA THORPE received the B.Sc. degree (Hons.) in computer science from University College Dublin, in 2005, and the Ph.D. degree in computer science, in 2011. From 2011 to 2018, she was a Postdoctoral Research Fellow with the Performance Engineering Laboratory, UCD. She is currently a Lecturer in cyber security with Technological University Dublin.

JOHN TOBIN (Graduate Student Member, IEEE) received the B.Sc. degree (Hons.) in computer science from University College Dublin, in 2015. He completed an internship with the Performance Engineering Laboratory (PEL), UCD, in September 2015, where he joined as a Researcher.

LIAM MURPHY (Member, IEEE) received the B.E. degree in electrical engineering from University College Dublin, in 1985, and the M.Sc. and Ph.D. degrees in electrical engineering and computer sciences from the University of California at Berkeley, Berkeley, CA, USA, in 1988 and 1992, respectively. He is currently a Professor of computer science with University College Dublin, where he is also the Director of the Performance Engineering Laboratory.