SECURE NEAR FIELD COMMUNICATION

Dhokia Alpesh[1]
[1]Electronics and Communication Engineering Department,
Sobhasaria Engineering College,
Sikar, Rajasthan, India.
[1]alps422@gmail.com

Abstract: Near Field Communication's card emulation mode is a way to put virtual smart cards into mobile phones. A recently launched application is Google Wallet. Google Wallet turns a phone into a credit card, a prepaid card and a tool to collect gift certificates and discounts. Card emulation mode uses dedicated smart card chips, which are considered to fulfill high security standards. Therefore, card emulation mode is also considered to be safe and secure. However, an NFC-enabled mobile phone introduces a significantly different threat vector. Especially a mobile phone's permanent connectivity to a global network and the possibility to install arbitrary applications onto smart phones open up for several new attack scenarios. This paper gives an overview of the new risks imposed by mobile connectivity and untrusted mobile phone applications. The various APIs for secure element access on different mobile phone platforms and their access control mechanisms are analyzed. The security aspects of mobile phones are explained. Finally, two practical attack scenarios, a method to perform a denial of service (DoS) attack against a secure element and a method to remotely use the applications on a victims secure element without the victim's knowledge, are highlighted.

Keywords: **NFC, CAI**.

1. INTRODUCTION

NFC is a radio technology that supports transactions at distances of a few centimeters. NFC is designed to support existing RFID transactions including contactless payments and some ticketing systems, as well as being a generally programmable platform. During a transaction, one party can be completely inactive, drawing power inductively from the active party. Even the active party draws little power and can be left on all the time with minimal elect on the phone's overall power draw. Also, the nearness of NFC transactions creates the possibility of using proximity as context and triggering an appropriate action almost instantaneously.

We envision widespread adoption of NFC in future generations of smart phones. The primary driver for the adoption of NFC on cell phones is contactless payments and ticketing. NFC, in the form factor of a credit card, has been used widely in Japan and Hong Kong for many years: for public transportation, vending machines, and convenience stores. Standards have also been created for "smart posters" [11]; posters, signs, and magazine pages can possess cheap, embedded data tags that contain information such as details of museum exhibits, transportation schedules, discount coupons, movie clips, or links to e-commerce sites. A third important use of NFC is for making connections between electronic devices simply touching the devices together will configure them to connect over a longer-range protocol such as Blue- tooth or Wi-Fi.

1.1 NFC Applications on the Phone

Analyzing the many applications we came up with using NFC, we have identified three important classes of CAI:

Transaction attachments: There is a class of useful applications that can be classified as attachments to traditional NFC payment and ticketing transactions. The products we pay for or the events we attend provide the context, the de- vice we connect with via NFC can supply us with additional information relevant to the context. For example, the bill as- associated with my payment can be transferred via NFC to my phone so that my phone can later submit it to my employer for reimbursement.

Virtual tokens: NFC can be used to replace various applications involving physical tokens: from getting a claim check for valet parking to getting loyalty cards from restaurants for attracting repeat customers. By using NFC on the phone, we do not have to worry about misplacing physical tokens; further- more, these virtual tokens can be entered into our databases and tracked automatically. Here, the token grantor provides the context that defines the relevant interaction. We need to define a secure protocol that protects the interests of both the grantor and grantee of the tokens.

Junctions: Friends can also take advantage of NFC to have their phones interact in peer-to-peer multi-party applications. For example, people may want to play a peer-to-peer game, share their playlists, or exchange photos. It is simple and direct if we can just launch an application, touch our phone with our friends, and have their phone automatically run the same application (after user confirmation). To facilitate this class of applications, we propose the notion of a Junction URI, which provides the context necessary for a device to join a peer-to-peer application in progress. Because phones do not have static IP addresses, a Junction URI specifies a (secure) channel, consisting of the chat session on a rendezvous server and an ID for the session. From the Junction URI, a device can also find out where the application can be downloaded.

2. TRANSACTION ATTACHMENTS

NFC was designed to interoperate with existing deployments of near-field radio technologies, including contactless payments and access to public transit systems. Moving these transactions to the phone may help reduce the number of things a person carries, but there are other more significant benefits.

We can improve the usability of ticketing for a public transit system by using the phone as our pass. The connectivity on the phone allows us to purchase the pass from anywhere, without waiting in line at a kiosk. We can also see how many rides we have available or how much credit is left in our account from anywhere. All the while, we can still swipe into the transit system quickly and also verify our ticket to a conductor onboard. [7]

We can improve the security of credit card transactions by moving the contactless payment to an active, programmable device by supporting one-time use credit cards. One-time use credit cards are tremendously useful for reducing credit card fraud—instead of giving a merchant our credit card numbers, we can request our bank to give us credit card numbers that can be used only once. So far the cumbersome procedures required to get single-use cards have limited adoption. With NFC on a phone, users can run an application that stores one-time credit card numbers securely and easily, and the application can present these one-time numbers to merchants, on behalf of the user. Users don't have to know about the added layer of security, using their phone to make payments as they would a contactless card. The phone may negotiate several one-time use numbers in advance so that payments can be made with the phone offline.
With these applications moved to the phone, we can further enhance the mobile experience by leveraging the contextual information gleaned from them. We first describe several such applications, then discuss the security considerations for NFC transactions.

2.1 Applications

Receipts, reimbursements, and money management.

As an add-on to contactless payments, we imagine the transaction results in a receipt being sent to the user's phone. The receipt may be transmitted as part of an enhanced standard for contactless payments**,** or may occur as an additional transaction during the same NFC scan. The phone keeps a local database of transactions and receipt objects, and allows programmatic access to them (with appropriate security restrictions). This will enable, for example, an application for managing receipts. Another application can help file reimbursement claims. After a business trip, a user could select purchases from a list of gathered receipts over some span of time. With a few clicks, she can email this list to file a reimbursement claim. The receipt data is stored privately on the phone, and is only released at the user's discretion.

Sporting events: We can use our NFC-enabled smart phones as a ticket for entry into sporting events. After scanning in, the phone launches an application associated with the event. It is loaded knowing the user's seat, and can be used to order concessions for delivery. Payment can occur through the application as well for a smoother user experience. The application can also better connect the user to the event, providing video replays and letting them interact with events occurring on the venue's big screen, such as trivia, polls, or shout-outs.

Reviews: Our phones will be able to determine the products we buy, the restaurants we visit, and the movies we see. The data can be kept privately, and applications can request permission to view different classes of data. If the user has a movie application installed, it may request access to movie- related events from the user's activity stream. This allows the user to plug into any of her favorite sites.

Public transportation. An NFC device can be used to access a public transportation system, be it train, bus, or subway. Again, scanning into the system can invoke an application. This program can provide the user with a real-time schedule, customized to their current stop, and can alert the user when their destination nears.

3. SECURITY OF NFC TRANSACTIONS

Security threats in current uses of NFC are well understood from similar applications in areas like content distribution (DRM), web browsing, and networking. Here we discuss techniques and principles to provide security in NFC-based applications.

3.1 Preventing unauthorized ticket sharing

In the case of electronically presented service "tickets", such as in public transportation or sports events without assigned seating, we have to ensure that users can not share their benefits with other parties. Consider the case of Shawn who has a ticket to watch the San Jose Sharks. Shawn decides to share his ticket with a friend: he beams the contents of the ticket over to Sara's Smartphone, and now both of them can present a valid token at the entrance. The means of dealing with unauthorized sharing depend heavily on the level of protection desired. At one end of the spectrum, a centralized database can keep track of used tickets at the venue, and a ticket becomes invalid once presented. Either Shawn or Sara can get in, but not both of them. Optionally**,** the ticket can be made valid again in case the owner exits the venue, to allow reentry. Note that if transfer of ownership must be supported, then centralized tracking of ownership has to be in place from the time of ticket issue, all the way to the time of use. A less centralized solution involves tickets that are tied to a specific person: at the time of ticket generation, the issuing authority uses a private key to sign the ticket along with a photo of the authorized owner. The benefits of this approach extend to long-term tickets that can be used multiple times (such as commuter rail permits or ski passes). Finally, in situations where long-term permits are not visually checked (such as in high-traffic areas like the subway), data mining can be used to verify legitimate use and flag suspicious cases for examination or even revocation.

3.2 Man-in-the-Middle Attacks

With NFC, we must watch out for the possibility for an attacker, a third party with an active tag, to inject itself in the conversation and modify it to his advantage possibly even

without being noticed. While peer certificates can go a long way towards excluding third parties from an exchange, they will never be the complete answer: certificates can be obtained fraudulently, or perhaps with an apparent owner which appears to be legitimate, but is not (such as using a slightly misspelled version of the legitimate owner). Because of this, it is imperative that interactions be designed with multiple safeguards: verification based on cryptography, as well as user verification and common sense (e.g. when confirming a payment, there should not be two or more simultaneous payment requests from different payees, or when payment is confirmed but the service is still unavailable, assume fraudulent use the payment went to the wrong destinations so the user should investigate).

3.3 Preventing relay attacks

In a relay attack the authentication protocol is bridged, such that authentication no longer requires physical proximity [4, 5]. Users transacting unique low-cost objects (such as people presenting movie tickets at the entrance) are particularly vulnerable to relay attacks. On the one hand, the low value of the transaction makes an interaction-free approach more acceptable. On the other hand, if the object owner is willing to publicly share the object, then she becomes vulnerable to malicious relaying of the ticket and involuntarily granting entry to an attacker. While relay attacks can be prevented by distance bounding [13], the technology is still in its infancy: a simpler approach could be to give ticket owners a choice be- tween security (user confirmation required to use the ticket) and convenience (the ticket is presented automatically). This behavior could adjust based on context: the ticket manage- ment application can decide whether it is safe to present a token without asking the user based on the device location and past history of fraud at that location.

3.4 Preventing relay attacks

In a relay attack the authentication protocol is bridged, such that authentication no longer requires physical proximity [4, 5]. Users transacting unique low cost objects (such as people presenting movie tickets at the entrance) are particularly vulnerable to relay attacks. On the one hand, the low value of the transaction makes an interaction-free approach more acceptable. On the other hand, if the object owner is willing to publicly share the object, then she becomes vulnerable to malicious relaying of the ticket and involuntarily granting entry to an attacker. While relay attacks can be prevented by distance bounding [13], the technology is still in its infancy: a simpler approach could be to give ticket owners a choice between security (user confirmation required to use the ticket) and convenience (the ticket is presented automatically). This behavior could adjust based on context: the ticket management application can decide whether it is safe to present a token without asking the user based on the device location and past history of fraud at that location.

4. P2P APPLICATIONS WITH JUNCTION

NFC is useful to introduce two peers so they can interact online. By allowing peers to exchange a session-specific secret, we can enable all forms of peer-based interactions, without having to be monitored by third-party servers. We have created Junction [8], a platform to support such interactions. NFC allows two devices to interact simply by placing them together. However, the nearness requirements for NFC would not lend itself well for comfortable interaction in a multi- party application session. Instead, we use NFC as a tool for bootstrapping multi-party applications, with the application session running over another channel. Imagine Alice wants to play a multiplayer dice game called Blue with her friend Bob, which is played across two or more phones. Alice opens her phone and browses to her Blue application. The application instructs her to tap the phones to start a game. Bob takes out his phone, unlocks it, and they touch their devices together. Bob hears a beep from his phone, indicating the NFC transaction worked. His phone asks him if he'd like to join the game of Blue, which he does. Had Bob not had the Blue application installed, his phone would prompt him to download and install it. After the installation, another tap of the devices launches the game.

In our example, only one device needs to explicitly launch the application. The other phone is given enough contextual information to locate the appropriate program and join the existing session. To avoid unwanted applications from launching on a device, we require the phone to be unlocked prior to the NFC transaction, and also prompt the user when we detect a joinable session.

We have created Junction to foster the creation of such multi-party applications. Junction includes client-side libraries for application development, as well as infrastructure supporting their connectivity at runtime. Junction applications do not have a central server, instead using a switchboard service that simply routes messages between devices, but does no application-specific computation.

Junction development is session oriented. A session supports multiple participants, with an activity uniquely represented with a URI such as:

junction://sb.openjunction.org/un1qu3?key=s3cr3t

This URI expresses four things.

The scheme "junction" indicates to a host platform that the URI should be recognized as a Junction session. "sb.openjunction.org" is the switchboard that is hosting the particular activity session, with session identifier "un1qu3". The URI also contains a key ("s3cr3t") that is used to encrypt communication. Because this URI is never seen by the switchboard, messages can be encrypted between peers without the switchboard knowing their contents. The Junction protocol also includes a mechanism for looking up the activity's details. Most importantly, an activity can have supporting code on a number of different platforms, and the protocol allows a device to locate the codebase it requires.

Since NFC is not yet commonplace on smart phones, we have been using QR codes to exchange Junction URIs. The user experience is more cumbersome to join a session; a user must launch the QR code scanning application and take a photo, a process that can take ten seconds or more. With NFC, the exchange of session information is nearly instant.

Because of the simple URI representation of sessions, integrating Junction with NFC is simple. One device emits a tag, with its content being the session URI. The other device scans the phone and handles the URI, which is opened with the Junction application installed on their phone, which in turn launches the appropriate peer-to-peer application. Our Junction applications that have been using QR codes need little to no modification to support NFC.

5. DEALING WITH DEVICE LOSS

With the phone holding increasing amounts of private, as well as financial data, loss of the device is a top concern. Password authentication has been the de facto standard for securing user data, yet passwords do not work well on smart phones: small or touch-based keyboards and frequent unlock events make entering a password every time a nuisance. The alternative is to build a hardware authenticator, in the form of a ring, wristband, or key, which can unlock the user's phone without requiring any interaction. A stolen device will be impossible to unlock without the matching authenticator.

Even though hardware authenticators have existed for a long time, they have failed to become ubiquitous because of their cost as well as complexity of deployment. One current example is the RSA SecureID, a small key fob device which provides a stream of unique numbers that the user can type along with her password when authenticating to remote services. The server verifies that the supplied number ungues sable by an attacker—matches what is expected. The cost of one authenticator is more than $10 per year, with the added expense and hassle of installing a central authentication server. With wide adoption of NFC, and the low cost to manufacture passive or semi-passive tokens, we envision that for the first time hardware authentication can become pervasive.

In a basic scenario, hardware authenticators can serve as keys for unlocking smart phones. A contactless, passive (no battery) key fob can perform a challenge-response protocol with the Smartphone over NFC. The authentication happens on boot as well as screen unlocks events. After the authentication, the key fob provides to the smart phone a symmetric cryptographic key necessary to unlock data storage on the phone—this key is only stored in volatile memory on the phone, and erased on screen lock or shutdown. Phone loss or thefts are now reduced to an inconvenience: without having physical possession of the authenticator, an attacker can no longer access any private data on the phone. Furthermore, without the authenticator the phone can lock up and become unusable, discouraging theft in the first place.

The NFC authenticators we envision in the future will be able to assume multiple identities, performing challenge- response authentication with different types of peers electronic de vices, online services, and physical-world infrastructure. Some credentials can work automatically: devices that the user carries (such as the smart phone itself) could be un- locked by the mere presence of the NFC token at the time of use. In other cases, the push of a button may be required to prevent relay attacks. Such attacks are a possibility when users unlock assets that are usually away, such as vehicles or buildings. Hardware authenticators must have dedicated buttons to confirm the unlocking of such assets to prevent car theft or home burglary while the owner is away.

6. RELATED WORK

More research has been done to bring interactions between mobile phones and physical spaces. Broll et al. explore a platform for running generic web services on phones, invoked by NFC or other proximity cue. [1] The focus is on supporting web services on phones, and in our paper we explore other context-aware platforms for mobile interactions.

The MIT Mobile Experience Laboratory demonstrates several real world uses of NFC in phones, including making payments and pairing devices for peer-to-peer games. [9] Gh`ıron et al. explore a system for virtual ticketing on an NFC- enabled phone. [7] We expand on these ideas to associate a contextually-driven platform to the transactions, supporting a new variety of applications.

With NFC Social, Fressancourt et al. explore the use of NFC to update presence information in a social network. [6] NFC- Social leverages the ability of an NFC transaction to invoke an application on a phone, and apply it to social networking check-ins. We expand on the idea to support in situ checkins, taking location-aware cues from other NFC transactions such as payments.

Junction uses NFC to launch programs across multiple phones. Similarly, Bump Technologies has built an API to support communication between two devices after they've "bumped" together. [2] Their approach requires a matching algorithm running in the cloud, and all subsequent data flows through a central server managed by Bump. We believe the many applications built with Bump demonstrate the utility of NFC as a method for bootstrapping cross-device applications, and that widespread adoption of NFC would replace the need for this cloud-based interaction. Also, NFC supports a better user experience, since one device can scan another without first launching a special application.

7. CONCLUSION

In this paper, we have presented our vision of how smart phone applications will change when NFC becomes common- place. NFC will allow what we term contextual application invocations. Applications can be invoked as a side-elect (attachment) of another transaction that provides it meaningful context. Applications can also be launched to exchange to- kens, with our phones responding to the context of the token grantor. Finally, one phone may provide context to another to create a junction between them, allowing them to partake in a cross-device activity. We have implemented the Junction platform and written several applications for it, demonstrating the usefulness of programmable NFC on smart phones.

REFERENCES:

[1] G. Broll, S. Siorpaes, E. Rukzio, M. Paolucci, J. Hamard, M. Wagner, and A. Schmidt. Supporting mobile service usage through physical mobile interaction. In In Proceedings of PerCom 2007, White Plains, pages 262–271. IEEE Computer Society, 2007.

[2] Bump technologies. http://bu.mp.
[3] R. Dhamija. The battle against phishing: Dynamic security skins. In In SOUPS 2005: Proceedings of the 2005 symposium on Usable privacy and security, pages 77–88. ACM Press, 2005.
[4] A. Francillon, B. Danev, and S. Capkun. Relay attacks on passive keyless entry and start systems in modern cars. Cryptology ePrint Archive, Report 2010/332, 2010. http://eprint.iacr.org/.
[5] L. Francis, G. P. Hancke, K. Mayes, and K. Markantonakis. Practical NFC Peer-to-Peer Relay Attack using Mobile Phones. In Workshop on RFID Security – RFIDSec'10, Istanbul, Turkey, June 2010.
[6] A. Fressancourt, C. Herault, and E. Ptak. Nfcsocial: Social networking in mobility through ims and nfc. Near Field Communication, International Workshop on, 0:24–29, 2009.
[7] S. L. Ghiron, S. Sposato, C. M. Medaglia, and A. Moroni. Nfc ticketing: A prototype and usability test of an nfc-based virtual ticketing application. Near Field Communication, International Workshop on, 0:45–50, 2009.
[8] Junction. http://openjunction.org/.
[9] MIT Mobile Experience Lab. A day at mit with near-field Communication.http://techtv.mit.edu/videos/1369-a-day-at-mit-with-near-field-communication.
[10] C. Mulliner. Vulnerability analysis and attacks on nfc enabled mobile phones. Availability, Reliability and Security, International Conference on, 0:695–700, 2009.
[11] NFC Forum. Smart poster record type definition technical specification. 2006.
[12] NFC Forum. Generic control record type definition technical specification. 2007.
[13] K. B. Rasmussen and S. ˇCapkun. Realization of rf distance bounding. In Proceedings of the USENIX Security Symposium, 2010.
[14] H. F. D. Team. Happy feet: The social way to walk, jog, run, cycle, or even... ski. 2010. http://happyfeet.heroku.com/.