NFC Security Research

Will Laurance

December 2011

Abstract

This paper analyzes the potential security risks associated with Near Field Communication protocols and provides secure implementation techniques for communication between NFC devices. This paper also looks into the social aspects of NFC.

Contents

1 Testing Environment

1.1 Hardware

- ACR122T NFC tag reader/writer
- ISO14443A MIFARE Classic 1k Card(s)
- Dell Latitude 620 x86, Ubuntu 10.04.3 32-bit

1.2 Libraries

- libnfc-1.4.1
- nfcip-java-1.3.1
- nfc-tools
- nfcutils
- mfcuk 0.3.3
- java-6-sun-1.6.0.26

2 Terms

NFC Near Field Communication.

RF Radio Frequency.

Active Mode When a device is powering the RF field.

Passive Mode When a device is woken up by an active RF field.

MD Message Digest.

UID Unique Identification

Foursquare A location based social application

NUID None unique identification number.

3 Initial Observations

Setting up the testing environment proved difficult as I had many dependencies and libraries to build and learn how to use. Using ‘nfc-list‘ command from the libnfc library, I was able to read the NUID from a 1k MIFARE Classic card.

4 Potential Security Risks

Although NFC is considered a close range radio transmission it is still possible to eavesdrop on the NFC packets. In a busy store, market, or other financial transaction instances, the possibility of eavesdropping is much of concern. According to researchers Ernst Haselsteiner and Klemens Breitfuß 2mm at Philips Semiconductors, "When a device is sending data in active mode, eavesdropping can be done up to a distance of about 10 meters, whereas when the sending device is in passive mode, the distance is significantly reduced to about 1 meter." [**?**, chap. 3.1]. A denial of service attack is possible because the attacker would only need to transmit data on the NFC frequency during an attempted communication. Haselsteiner and Breitfuß claim that data modification is possible in both Miller and Manchester coding schemes for transmitting data, but Miller is more secure because only half the bits can be altered when 100% ASK modulation is used. In Miller, an attacker can only change a 1 to 0, not the opposite. However a 1 can change to a 0 and a 0 to a 1 with Manchester encoding. Thus Miller encoding is the industry standard for modulation because it is impossible to change the entire message.

4.1 Data Validation and Integrity

Another security risk NFC devices encounter is data validation and integrity. If there is an NFC tag that is supposed to be scanned to read a URL, token, or other information, how do we know that it was the intended information? Someone could have rewrote the tag with their own data. There must be a mechanism to allow NFC devices to know if they have received the intended data from the read tag. The location of said tags need to be tamper-proof in order for security to work. Otherwise someone could physically switch the tag with a malicious one. Reading and writing operations need to ensure that the data is exchanged correctly. According to NXP Semiconductors, the MIFARE Classic 1K uses the following mechanisms to ensure data integrity:

- 16 bits CRC per block
- Parity bits for each byte
- Bit count checking
- Bit coding to distinguish between "1", "0" and "no information".
- Channel monitoring (protocol sequence and bit stream analysis)

[**?**, page 7]

The byte level protocol for the MIFARE type tags handles the lower level guarantee of data being written and read correctly. In order to ensure the reader is authorized to write to the card, a 48 bit secret key is loaded on to the card and known to the reader at the initialization period. As we know brute forcing 2^{48} key space is certainly possible. Thus an application level protocol should be implemented to ensure data integrity.

4.2 Man-in-the-Middle-Attack

Unlike WIFI connections and other over air protocols, NFC is practically immune to a man-in-the-middle attack. The NFC protocol is immune because of the nature of RF fields. Haselsteiner and Breitfuß explain:

> Alice and Bob must not be aware of the fact that they are not talking to each other, but that they are both sending and receiving data from Eve. Such a setup is the classical threat in unauthenticated key agreement protocols like Diffie-Hellmann protocol. Alice and Bob want to agree on a secret key, which they then use for a secure channel. However, as Eve is in the middle, it is possible for Eve to establish a key with Alice and another key with Bob. When Alice and Bob later use their key to secure data, Eve is able to eavesdrop on the communication and also to manipulate data being transferred. How would that work when the link between Alice and Bob is an NFC link? Assuming that Alice uses active mode and Bob would be in passive mode, we have the following situation. Alice generates the RF field and sends data to Bob. In case Eve is close enough, she can eavesdrop the data sent by Alice. Additionally she must actively disturb the transmission of Alice to make sure that Bob doesnt receive the data. This is possible for Eve, but this can also be detected by Alice. In case Alice detects the disturbance, Alice can stop the key agreement protocol. Lets assume Alice does not check for active disturbance and so the protocol can continue. In the next step Eve needs to send data to Bob. Thats already a problem, because the RF field generated by Alice is still there, so Eve has to generate a second RF field. This however, causes two RF fields to be active at the same time. It is practically impossible to perfectly align these two RF fields. Thus, it is practically impossible for Bob to understand data sent by Eve. Because of this and the possibility of Alice to detect the attack much earlier we conclude that in this setup a Man-in-the-Middle attack is practically impossible. [**?**, chap. 3.5]

Even when Alice does not check her RF field for interception, it proves difficult for Eve to successful send a comprehensible message to Bob. Thus one can implement a simple unauthenticated version of the Diffie-Hellmann key-exchange protocol for the above reasoning.

5 Implementation

The following section will discuss upper level ideas to solve the potential problems of data integrity and validation and outline the secure channel. For this research, I am assuming that the UID and hash I assign to the MIFARE 1K card are un-writable after the initial write. This is a valid assumption because there are a few blocks that is read-only.

5.1 Data Format

To conform, data written to NFC tags must use this format:
UID=UID_NUMBER?MD=MESSAGE_DIGEST?URL=URL
Thus the characters ? and = must be escaped with &#61 and &#63 respectively if they appear in any values.

5.2 Protocol Outline

5.2.1 Data Integrity and Validation

To ensure that the data read from the tag is what the original tag owner wanted, we will use the idea of a cryptographic hash digest of the data. Stored will be the tags UID number from a remote database and the the salted hash of the url. The salt will be retrieved from a remote site and used to validate the store data. I will use SHA1 as the algorithm due to its weak and strong collision resistance, high avalanche effect, and speed.

The basic writing algorithm is as follows:

1. Register tag with service to receive a UID for the tag, and a salt.
2. Write tag with the UID, the MD of the URL, and the URL itself.

The reading algorithm is as follows:

1. Read tag
2. Extract UID, MD, URL and store locally
3. Lookup UID and get salt.
4. Hash(URL + salt)
5. Compare to given MD
6. Warn user or continue operations.

Here is a sample run of some test code. It reads the data from the card and sends it to a integrity checker. The first example is the URL hashing to the stored value on the card. The first item in each list is the record read from the card.

UID=123456789?MD=3b2b133bbcd76d6770a885d1e2e8faea7fc3c4b7?URL=www.wlaurance.com

3b2b133bbcd76d6770a885d1e2e8faea7fc3c4b7

3b2b133bbcd76d6770a885d1e2e8faea7fc3c4b7

This second set is an example of the stored hash value not matching the hash value of current URL stored on the card. Thus this card is invalid.

UID=123456789?MD=3b2b133bbcd76d6770a885d1e2e8faea7fc3c4b7?URL=www.evilsite.com

3b2b133bbcd76d6770a885d1e2e8faea7fc3c4b7

8aa1a54d45e83a8cb3322f1af59446658c0dedf5

5.2.2 Secure Communication Channel

When NFC interects between two smart parties a new type of connection is made. Peer-to-Peer NFC requires a more sophisticated security implementation because of the sensitive data the two or more parties might be exchanging. In our implementation we will assume that such contents are financially related such as access numbers, pin codes, credit card creditials, or other secure tokens that should be secret. To be PCI compliant, we must, "Use strong crytpography and security protocols such as SSL/TLS or IPSEC to safegaurd sensitive cardholder data during transmission over open, public networks."[**?**, sec. 4.1] This secure communication channel will make the assumption the the client is a network enabled smartphone and the server is a payment terminal with internet access. Thus both parties can quickly communicate to third party servers.

1. The first thing the client needs to do is authenticate the server in which it is communicating to. Using an enhanced SSL handshake
   (a) Client will send server a hello messsage with a random number R_1
   (b) Server will reply with its Certificate, and random number R_2
   (c) Client will verify the server's public key
   (d) Client will send the server $[PreMasterSecret]_{E_{ks+}}$
   (e) Both compute the MasterSecret = $f(R_1, R_2, PreMasterSecret)$
2. Once a master secret is shared by both parties a session key can be devised.
3. The channel should use 3DES or another stream cipher encryption algorithm that is compliant with PCI standards.
4. The session keys should be used only once per transaction.

Things to consider are the exspensive network calls this protocol uses and the amount of data transferred via NFC transmissions. It might be too costly to implement this because it is not quick enough for fast and secure NFC payments. However, I believe that a protocol similar to SSL is the currently the best solution to authenticate the server and to have secure over the air transmission of sensitive data.

6 Conclusion

Although the technology is available to implement a NFC based payment system, many experts believe it will be atleast three to five years before we see such systems. Verizon Wireless confirmed that they will not support Google Wallet for the Galaxy Nexus, Google's NFC based transaction system, in early December 2011. There are still many debates about which NFC protocols are acceptable and secure. Many different wireless carriers, Google, and other NFC related companies are involved in these debates and a conclusion does not seem likely any time soon. [**?**]

I recently attended the Mobile Monetization Forum in Washington D.C where I heard many panelists including Mark Murphy from Commonsware and Larry

McDonough from RIM express their thoughts and concerns regarding NFC based transactions. They felt like NFC would be used more for social interaction than financial transactions. A good example of NFC for social integration would be a NFC tag in a building and upon scanning it with a mobile device, it would automatically check you in on Foursquare. Assuming devices are networked enabled there exist other feasable and viable solutions to mobile payments. [**?**] Such solutions include wipit, a mobile payment system designed for the "cash preferred" customer in mind.

NFC is not a new technology, but it is yet to have a place in American culture. Many business and organizations are working to standardize NFC and soon it will have a new home.

7 Future Research

I will continue to read about NFC's standards progress and continue my implementation of a secure based channel. I will also consider other alternitives that can speed up the transaction progress. Once I have an NFC enabled Android device, I will port this code to be able to run on Android.

8 Source Code

The source for this project is available at `https://github.com/wlaurance/NFC-Research` An online version of this pdf can be found on my website.