NFC - NEAR FIELD COMMUNICATION

Subho Halder and Aditya Gupta
(@sunnyrockzzs and @adi1391)

INTRODUCTION

Near Field Communication at glance.

What is NFC ?

NFC or Near Field Communication is a set of standards or protocols to communicate between two devices by either touching or bringing into close proximity (less than 4 cm).

The communicating protocols of such devices are based on RFID Standards, including ISO 14443. These standards are defined and extended by the NFC Forum, which was founded on 2004 by some major companies such as Sony, Nokia, Philips, Samsung etc.

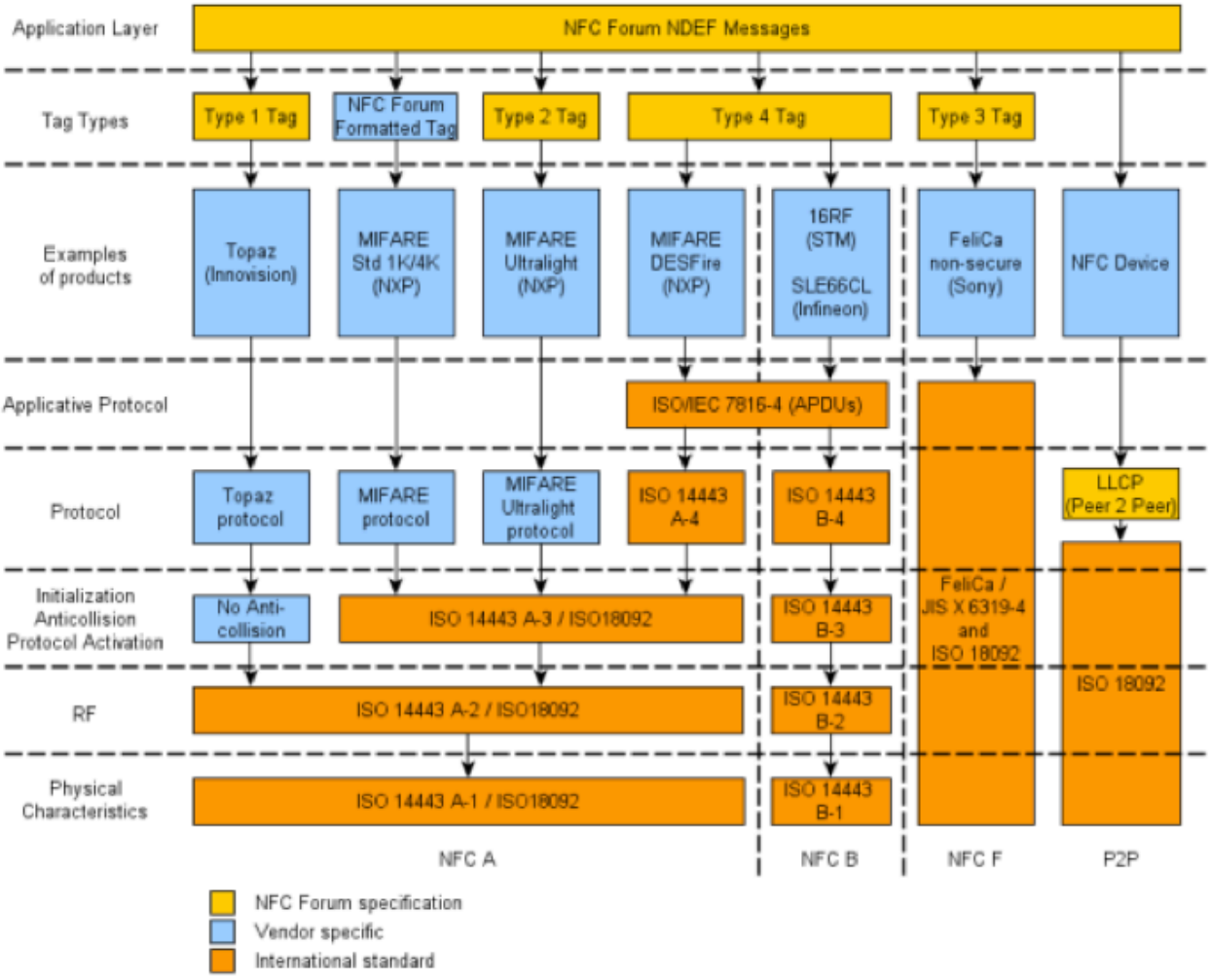
The operating Frequency of such communication is merely 13.56 MHz (+/- 7) which is very low. This gives an advantage of easily integrating into portable devices without the need of much battery power.

Types of Communication

There are basically two types of communication possible in NFC based devices

1. **Passive :** In this type of communication unpowered NFC "tags" can be read using NFC enabled devices. The initiator, that is the NFC device provides power to the "tag" which re-transmits back with the recorded data.
2. **Active :** In this type of communication, both devices simulates power to transmit data between each other. It can be more or less generalized as a Peer-to-Peer (P2P) transmission. In this way, binary/multimedia files can be transmitted with ease.

NFC Stack



The above figure shows the basic protocol layout of the NFC Stack.

For the purpose of this report, we will be discussing more about the Protocol Layer of this stack which are focussed on physical aspect of starting communication and the Application Layer which are focussed on how the data are transmitted during the communication.

The Protocol Layer

There are basically 6 division based on the protocol layers. We will be discussing about the major 4 types, namely Type 1 (Topaz), Mifare Classic, Mifare Ultralight, LLCP (P2P)

TYPE 1 (TOPAZ)

Type 1 tags use a format sometimes called the Topaz protocol. It uses a simple memory model which is either static for tags with memory size less than 120 bytes or dynamic for tags with larger memory. Bytes are read/written to the tag using commands such as RALL, READ, WRITE-E, WRITE-NE, RSEG, READ8, WRITE-E8, WRITE-N8.

MIFARE CLASSIC

MIFARE classic tags are storage devices with simple security mechanisms for access control. They use an NXP proprietary security protocol for authentication and ciphering. This encryption was reverse engineered and broken in 2007.

MIFARE-ULTRALIGHT

These tags are similar to Topaz tags. They have a static memory layout when they have less than 64 bytes available and a dynamic layout otherwise. The first 16 bytes of memory contain metadata like a serial number, access rights, and capability container. The rest is for the actual data. Data is accessed using READ and WRITE commands.

LLCP (P2P)

The previous protocol layers have all had initiators and targets and the protocols are designed around the initiator being able to read/write to the target. Logical Link Control Protocol (LLCP) is different because it establishes communication between two peer devices.

NFC Application Layer

This layer is focussed mainly on the format through which the data are exchanged between NFC devices or between an NFC device and Tags.

NFC uses NDEF or NFC Data Exchange Format, a format which was standardized in NFC Forum, is used to transmit data. This is a simple binary message format. There are many types of message format such as text, url, etc.

One example NDEF is given in the next section. For clarity, and because the NDEF format is so important for NFC, we provide another couple of examples here. We start with a "text" which is basically a text type data.

```
0000: 03 17 D1 01 13 54 02 65 6E 68 65 6C 6C 6F
0001: 20 63 6C 75 62 68 61 63 6B 20 21 FE
```

Now lets decode the NDEF message from the above example

```
03 - NDEF message start
     17 - Payload Length
     D1 - MB, ME, SR, TNF= "NFC Forum well-known type"
     01 - Length of Type, in this case =1
     54 - Type of message, in this case ="T" which is text
     02 - Length of Language code, in this case =2
          65 6e - Language Code, in this case ="en"
          68 65 ... 21 - "hello clubhack" text
FE - NDEF Terminator
```

The previous NDEF example had a single byte devoted to the length of the payload. To support payloads longer than 255 bytes, a longer form of NDEF is used. (You can tell which variant to expect by whether the SR bit is set in the first byte of the NDEF record or not).

Android NFC Stack

Our scope will be limited to Android based NFC devices.

The above figure shows you the different libraries which are present in the Android stack.

Android NFC stack could be divided into three components - Kernel, NFC Services and the tag(or device) itself.

The kernel contains the NFC driver named *libpn544_fw.so*, which will respond and interact with the necessary NFC signals. The NFC service present in the android device is named as *com.android.nfc*. It relies on 3 main driver components : *libnfc.so*, *libnfc_jni.so* and *libnfc_ndef.so*. The components are divided on the basis of which component will contain which part of the NFC data (JNI, NDEF or any other).

So, in the real scenario, once the tag(or other NFC enabled device) is brought close to a Android NFC device, the kernel component *libpn544_fw.so* calls the NFC services. Once the NFC services are called, they receive the NFC data and store the information dividing it into proper categories. The most interesting part among all is the *libnfc_ndef.so*, which is responsible for the NDEF part. So, if suppose, we want to fuzz the NFC driver, we would be modifying the NFC data, making some modifications in the hex data stored in it, or changing the value of the length of the message, all of which is contained in the NFC component

NFC ATTACK SURFACE

Common NFC based Attack Vectors

ATM Card Skimmers

In countries where ATM cards are NFC enabled, the ATM cards could be used to complete transactions using the NFC functionality. So once the card gets in contact with the NFC card reader, the card reader retrieves some of the information from the card, and uses it to complete the payment.

What an attacker could do in this scenario, is install his custom NFC enabled card reader in any of the ATMs, which accept NFC enabled credit cards. So, once the user goes to the ATM and uses the NFC enabled ATM card, the attacker's card reader would retrieve the information, and then pass it to the original card reader machine. So, this could be seen as an example of Man in the Middle Attack. Also, at the end of the day, the attacker could come to the ATM Machine, and take away his installed card reader, and get the information of all the cards used on that ATM on that day. He could then further use those card information, to perform malicious transactions and other activities.

NFC Poster Skimming

Another attack vector using NFC could be seen regarding the NFC based POSTERS. The posters are used to provide advertisement where when an NFC enabled device is tapped to the specified location in the poster the information is transferred, for example we came across an NFC enabled Poster advertising a newly released track by a famous artist in Chicago International Airport.While tapping my GalaxyS3 with that poster

i go redirected to a signup form, upon completion i was able to download the trailer of the music video.A phone is a place where most of our private information are stored , an attacker can use this poster to transmit applications to your device hereby compromising security.

NFC Relay Attack

This dangerous form of attack compromises the security of all organizations depending upon NFC cards as a proof of identification of customers or employees. An arbitrary example is the company provided id card which we use to get access to the building , an active device can read and copy the data from the passive cards and store it, thereby becoming a clone of the card, now instead of swiping the card , we can swipe the phone containing the data ACCESS GRANTED!

This was the companies security at risk, now on personal security, an NFC enable credit or debit card will cause enough damage, imagine even without stealing the card from the user data will get transferred whenever an active device is close enough. A harmless dash against a stranger on a busy street will be enough to loose all your bank balance!

LEVERAGING NFC FOR ANDROID BASED VULNERABILITY

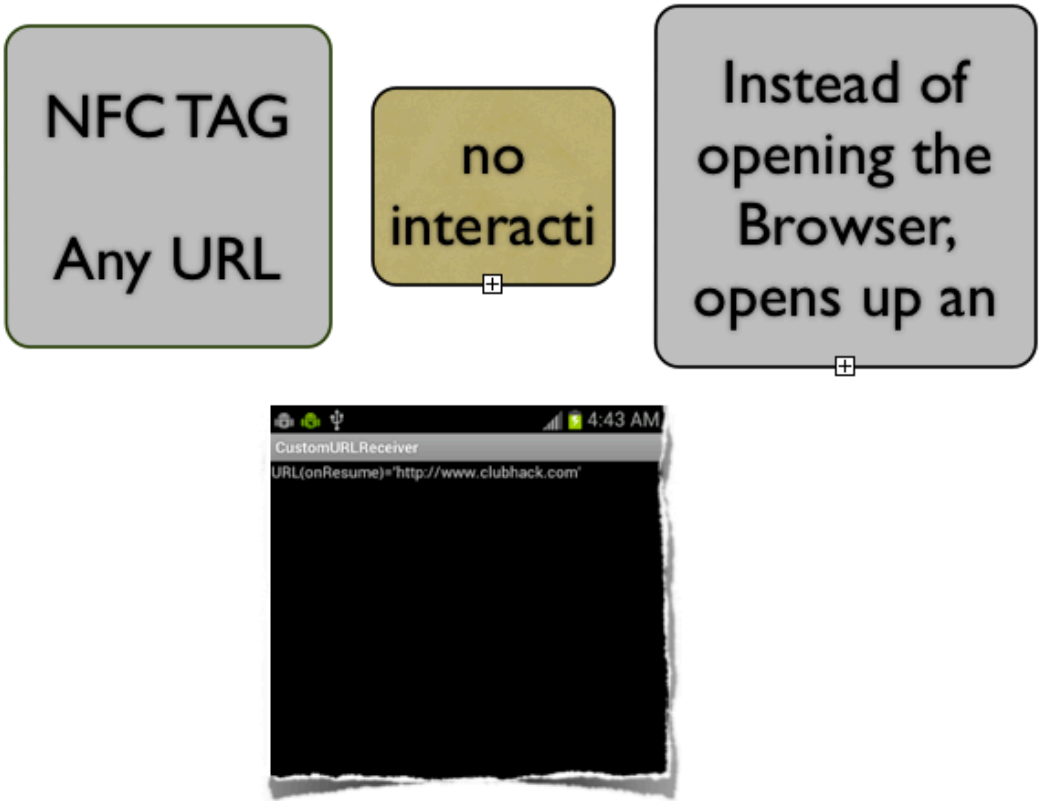
NFC AWARE ANDROID MALWARE?

For well known type Tags, the applications are called directly instead of the *com.android.tag*

- www data fires up the Browser
- mailto: protocol fires up the email client
- unexpected values in NDEF crashes NFCService.java

NFC Aware Malware

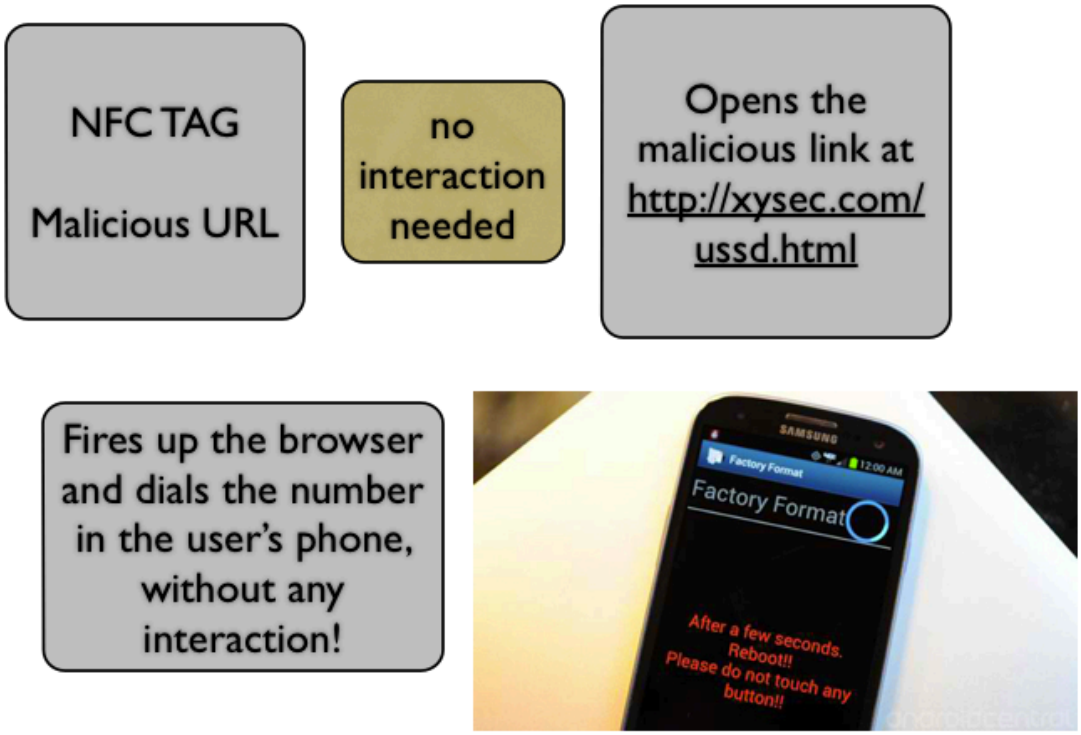
Leveraging the NFC based protocol, a new breed of NFC aware Malware can arises. These kind of Malware can proxy through the request through themselves before the correct application can get activate. One such example is proxy-ing any URL which are stored in an NFC tags, when parsed, fires up the malware instead of the Browser.



This application is hosted at the github repo : https://github.com/subho007/HTTProxy

USSD Based Attack using NFC

Well known USSD vulnerability in Samsung Galaxy devices which resets the complete Device can also be done through a simple NFC tag, which automatically opens up the browser without any user interactions, which in turns dials up the USSD code, which in turn resets the device to factory setting !



For any suggestions, or other bugs/improvements, mail us at **security@xysec.com**

REFERENCES

http://developer.android.com/guide/topics/connectivity/nfc/index.html

http://media.blackhat.com/bh-us-12/Briefings/C_Miller/BH_US_12_Miller_NFC_attack_surface_WP.pdf

http://nakedsecurity.sophos.com/2012/09/24/android-nfc-hack-lets-subway-riders-evade-fares/

http://blackwinghq.com/assets/labs/presentations/EddieLeeDefcon20.pdf

http://www.slideshare.net/the_netlocksmith/defcon-2012-nearfield-communicationrfid-hacking-miller

http://www.mulliner.org/nfc/feed/nfc_ndef_security_ninjacon_2011.pdf

http://media.blackhat.com/bh-us-12/Briefings/C_Miller/BH_US_12_Miller_NFC_attack_surface_WP.pdf