Sécurité des Applications NFC et Emergence du Cloud Of Secure Elements (CoSE)

Pr Pascal Urien
Telecom Paristech
Co-Founder of the EtherTrust Company
Agenda

• Introduction to NFC technologies
• About NFC standards
• NFC in mobile phones
• The Emergence of the Cloud Of Secure Elements (CoSE)
• Security for the NFC, LLCPS
Introduction to NFC Technologies
Smartcard Genesis

• 1980, First BO’ French bank card, from CP8
• 1988, SIM card specification
• 1990, First ISO7816 standards
• 1991, First SIM devices
• 1995, First EMV standards
• 1997, First Javacard
  – The javacard is a subset of the java language
  – Patent US 6,308,317
• 1998, JCOP (IBM JC/OP)
• 1999, Global Platform (GP)
• 2002, First USIM cards

1988, the 21 (BO’) chip
NFC Genesis

• 1994, Mifare 1K
  – In 2011 Mifare chips represent 70% of the transport market.

• 2001, ISO 14443 Standards (13,56 Mhz)
  – Type A (Mifare)
  – Type B
  – Type F (Felica)

• 2004, NFC Forum
  – Mifare (NXP), ISO14443A, ISO14443B, Felica (Sony)
  – Three functional modes :
    • Reader/Writer, Card Emulation, Peer to Peer

• NFC controllers realize NFC modes

Pascal Urien, Porquerolles, 28 mai 2013
From ISO 7816 to ISO 14443

• The basic idea of Wi-Fi design was Wireless Ethernet.
• The basic idea of ISO 14443 design was Wireless (ISO 7816) Smartcard.

  – Contrary to IEEE 802.11 there is no security features at the radio frame level.

ISO 7816
Contact Mode

ISO 14443
Contactless Mode

\[ V = 2 \pi f_c S \mu_0 H \]

\[ H = 5 \text{ A/m} \]
\[ f_c = 13.56 \text{ Mhz} \]
\[ S = 40 \times 10^{-4} \]
\[ V = 2.2 \text{V} \]

Pascal Urien, Porquerolles, 28 mai 2013
What is a Secure Element?

A Secure Element (SE) is a Secure Microcontroller, equipped with host interfaces such as ISO7816, SPI or I²C.

- OS: JAVACARD, JCOP
- GP: Global Platform
- ROM: 160 KB
- EEPROM: 72 KB
- RAM: 4KB
- Crypto-processor: 3xDES, AES, RSA, ECC
- Certification: CC EAL5+
- Security Certificates: EMVCo
NFC and Secure Elements

- Some NFC Controllers embed a Secure Element
  - In that case the card emulation mode may be managed by the embedded secure element
  - This is the Google Secure Element Android Model
The SIM card becomes an NFC device: the Contactless Front-end (CLF)

The ETSI TS 102 613 Standard

A simplified HDLC protocol: SHDLC

A physical Link: Single Wire Protocol (SWP)
NFC Reader/Writer & Card Emulation

Reader/Writer

Card Emulation

Pascal Urien, Porquerolles, 28 mai 2013

OTA (Over The Air) Administration
SMS, GSM 3.48

NFC CONTROLER

SECURE ELEMENT

Secure Element Administration

SOFTWARE

Pascal Urien, Porquerolles, 28 mai 2013
NFC P2P Mode  Illustration

- Android NDEF Push Protocol Specification
  - Version 1, 2011-02-22
  - Proprietary protocol, Android 2.3
  - Replace by SNEP for Android 4.x

“The NDEF Push Protocol (NPP) is a simple protocol built on top of LLCP which is designed to push an NDEF message from one device to another.”

Pascal Urien, Porquerolles, 28 mai 2013
Using the Google NDEF Push Protocol (NPP)

**ATR_REQ**, NFC-MAGIC VERSION  WKS (Well-Known Service)
LTO (Link-Time-out)

**ATR_RES**, NFC-MAGIC VERSION  WKS (Well-Known Service)
LTO (Link-Time-out)

**LLCP-SYMM** [0000]

**CONNECT** [0521 060F 636F6D2E616E64726F69642E6E7070]
DSAP=1, SSAP=33, Service=“com.android.npp”

**CC** (Connection Complete) [859002020078]
DSAP=33, SSAP=16, MUI (Maximum Information Unit)

**Information**
[432100 010000000101000000F D1010B5402656E 6B657976616C7565]
DSAP=16, SSAP=33, N(S)=0, N(R)=0, NPP HEADER, NDEF RECORD, **keyvalue**

**RR(1)** [855001], SSAP=16, DSAP=33

**DISCONNECT** [4161] DSAP=16, SSAP=33

Pascal Urien, Porquerolles, 28 mai 2013
Smartcard Administration: GP

KMC-ID (6B) KMAC KDEK

Select ISD

Mutual Authentication

Secure Channel

Application Management downloading - deletion

The VISA Model*

*EMV Card Personalization Specification Version 1.1 July 2007

Pascal Urien, Porquerolles, 28 mai 2013
The In2Pay Administration Model*

* http://www.devifi.com/assets/whitepaper.pdf

Pascal Urien, Porquerolles, 28 mai 2013
The Google Platform

Reader/Writer

Card Emulation

Peer to Peer

Android Beams

NFC Tags

- EMV Magnetic Stripe Profile
- Cloud Storage

Pascal Urien, Porquerolles, 28 mai 2013
HID NFC White Paper:
SIM centric Services

- Payment
- Access Control
- Transport

NFC ecosystem with the Secure Element in the SIM and one MNO

Pascal Urien, Porquerolles, 28 mai 2013
About NFC Specifications
In the NFC Jungle
# NFC Standards Overview

<table>
<thead>
<tr>
<th>Activity</th>
<th>Technology / Device Platform</th>
</tr>
</thead>
<tbody>
<tr>
<td>Listen, RF Collision</td>
<td>NFC-A</td>
</tr>
<tr>
<td>Avoidance, Technology</td>
<td>ISO 14443-2A</td>
</tr>
<tr>
<td>Detection, Collision</td>
<td>ISO 14443-3A</td>
</tr>
<tr>
<td>Resolution</td>
<td></td>
</tr>
<tr>
<td>NFC-A</td>
<td></td>
</tr>
<tr>
<td>ISO 14443-2A</td>
<td></td>
</tr>
<tr>
<td>ISO 14443-3A</td>
<td></td>
</tr>
<tr>
<td>NFC-B</td>
<td>ISO 14443-2A</td>
</tr>
<tr>
<td>NFC-F</td>
<td>ISO 14443-3A</td>
</tr>
<tr>
<td>NFC-F</td>
<td>FELICA</td>
</tr>
</tbody>
</table>

**Device Activation**
- NFC-DEP Protocol
  - Type 1 Tag Platform
  - Type 2 Tag Platform
  - Type 4A Tag Platform
  - Type 4B Tag Platform

**Data Exchange**
- NFC-DEP Protocol
  - Type 1, 2, and 3 Tag Half-duplex Protocol
- ISO-DEP Protocol
  - Type 1, 2, and 3 Tag Half-duplex Protocols

**Device Deactivation**
- NFCIP-1
- ISO 14443-4
- NFCIP-1

*ISO/IEC_18092 standard and NFCIP-1 standards are similar*

DEP: Data Exchange Protocol (Supports Read/Write Operations for Tags)

Pascal Urien, Porquerolles, 28 mai 2013
## NFC Radio

### Standard

<table>
<thead>
<tr>
<th>ISO 14443</th>
<th>PCD to ICC</th>
<th>PICC to PCD</th>
</tr>
</thead>
<tbody>
<tr>
<td>106 kbps</td>
<td>Reader to Card</td>
<td>Card to Reader</td>
</tr>
<tr>
<td>212 kbps</td>
<td>ASK 100%</td>
<td>Subcarrier fc/16</td>
</tr>
<tr>
<td>424 kbps</td>
<td>Modified Miller</td>
<td>OOK Manchester</td>
</tr>
<tr>
<td>848 kbps</td>
<td>ASK 10%,</td>
<td>Subcarrier fc/16</td>
</tr>
<tr>
<td>NFC-A</td>
<td>NRZ-L</td>
<td>BPSK, NRZ-L</td>
</tr>
<tr>
<td>NFC-B</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Bit Rate

<table>
<thead>
<tr>
<th>Bit Rate</th>
<th>Initiator</th>
<th>Target</th>
</tr>
</thead>
<tbody>
<tr>
<td>106 kbps</td>
<td>ASK 100% Modified Miller</td>
<td>Subcarrier fc/16 OOK Manchester</td>
</tr>
<tr>
<td>212-424 kbps</td>
<td>ASK 8-30% OOK Manchester</td>
<td>ASK 8-30% OOK Manchester</td>
</tr>
</tbody>
</table>

### NFCIP-1

#### Passive Mode

- ISO 14443-2A
- NFC-A

#### Active Mode

- ISO 14443-2B
- NFC-B

---

Pascal Urien, Porquerolles, 28 mai 2013

---

19 /47
LLCP: a Bridge to LAN Technologies

<table>
<thead>
<tr>
<th>LLC</th>
<th>MAC</th>
<th>PHY</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

IEEE 802 LAN Reference Model

<table>
<thead>
<tr>
<th>Physical layer</th>
<th>Data Link layer</th>
<th>LLC-MAC mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>NFC Forum Digital Protocols NCPIP-1</td>
</tr>
</tbody>
</table>

IEEE 802.15.4 MAC Payload Formats

Table 3: PDU Type Values

<table>
<thead>
<tr>
<th>PDU Type</th>
<th>PTYPE</th>
<th>Link Service Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYMM</td>
<td>0000</td>
<td>1, 2, 3</td>
</tr>
<tr>
<td>PAX</td>
<td>0001</td>
<td>1, 2, 3</td>
</tr>
<tr>
<td>AGF</td>
<td>0010</td>
<td>1, 2, 3</td>
</tr>
<tr>
<td>UI</td>
<td>0011</td>
<td>1, 3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PDU Type</th>
<th>PTYPE</th>
<th>Link Service Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONNECT</td>
<td>0100</td>
<td>2, 3</td>
</tr>
<tr>
<td>DISC</td>
<td>0101</td>
<td>1, 2, 3</td>
</tr>
<tr>
<td>CC</td>
<td>0110</td>
<td>2, 3</td>
</tr>
<tr>
<td>DM</td>
<td>0111</td>
<td>1, 2, 3</td>
</tr>
<tr>
<td>FRMR</td>
<td>1000</td>
<td>2, 3</td>
</tr>
<tr>
<td>SNL</td>
<td>1001</td>
<td>1, 2, 3</td>
</tr>
<tr>
<td>reserved</td>
<td>1010</td>
<td></td>
</tr>
<tr>
<td>reserved</td>
<td>1011</td>
<td></td>
</tr>
<tr>
<td>I</td>
<td>1100</td>
<td>2, 3</td>
</tr>
<tr>
<td>RR</td>
<td>1101</td>
<td>2, 3</td>
</tr>
<tr>
<td>RNR</td>
<td>1110</td>
<td>2, 3</td>
</tr>
<tr>
<td>reserved</td>
<td>1111</td>
<td></td>
</tr>
</tbody>
</table>

Figure 1: Relationship to OSI Reference Model

<table>
<thead>
<tr>
<th>LLCP Header</th>
<th>LLCP Payload</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DSAP</th>
<th>PTYPE</th>
<th>SSAP</th>
<th>Sequence</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>6 bits</td>
<td>4 bits</td>
<td>6 bits</td>
<td>0 or 8 bits</td>
<td>M x 8 bits</td>
</tr>
</tbody>
</table>

Pascal Urien, Porquerolles, 28 mai 2013

20 /47
**NDEF: NFC Data Exchange Format**

1 = Record Start
1 = Record End
1 = First Chunk
1 = Payload Length
1 = ID Length

**Structure of TYPE Field**

1 = Well Known Identifier
describing the TYPE of the payload

URI reference (RFC 3986)

---

**NDEF Record Example:**

(NFC Text Record Type Definition)

D1: 1 1 0 1 0 001
01: Type Length
0A: Payload Length
54: Type= ‘T’, Text
02: ID= UTF8
65 6E: “EN”
5D 61 6D 70 6C 65 20: "Sample 

A summary of record TYPE may be found in “NFC Tags A technical introduction, applications and products Rev. 1.3 - 1 December 2011 White paper”, NXP Semiconductors.

Pascal Urien, Porquerolles, 28 mai 2013
SNEP, Android 4.x

NFC Initiator

**ATR_REQ**, NFC-MAGIC VERSION WKS LTO

**ATR_RES**, NFC-MAGIC VERSION WKS LTO

**LLCP-SYMM** [0000]

**LLCP-SYMM** [0000]

**LLCP-SYMM** [0000]

**CONNECT** [1120], DSAP=4, SSAP=32

**CC** [818402020078], DSAP=32, SSAP=4, MUI

**Information, SNEP PUT**

[132000 10020000000F D1010B5402656E 6B657976616C7565] DSAP=4, SSAP=32, NS=0, NR=0, SNEP HEADER, NDEF RECORD, keyvalue

**RR(1)** [834401], SSAP=4, DSAP=32

**Information, SNEP Success** [830401 108100000000], SSAP=4, DSAP=32, NS=0, NR=1

**RR(1)** [13600], DSAP=4, SSAP=32

**DISCONNECT** [1160] DSAP=4, SSAP=32

**DM** [C400], DSAP=16, SSAP=32

**LLCP-SYMM** [0000]

NFC Target

Pascal Urien, Porquerolles, 28 mai 2013
NFC In Mobile Phones
NFC and Smartphones

- Nokia
  - Card Emulation and SWP
  - NOKIA 6131
- Android 2.3 (Gingerbread), Android 4.0
  - Reader/Writer and P2P
  - Nexus S (v2.3), Galaxy Nexus (v4.0), Galaxy S2 (v2.3)
  - NXP NFC Controller PN65N
- RIM JDE 7.0.0, Blackberry 10
  - Reader/Writer and Card Emulation
  - JSR 177 (SIM Access)
  - Blackberry Bold 9900, 9930
  - INSIDE SecureRead NFC Controller
- IPHONE
  - External NFC Reader
  - Rumors for the NFC support

Pascal Urien, Porquerolles, 28 mai 2013
Hardware and Software Architecture of the Nexus S Android Phone

- Application Processor (1GHz) + GPU
- RAM 512 Mo
- Flash 1+15 Go
- CPU core, "Hummingbird"
- Flash
- WiFi
- GPS
- NFC
- NFC CTRL
- Baseband Processor
- Radio Layer Interface (RIL)
- Baseband
- SIM
- Front End Module (FEM)
- COM PASS
- USB
- ACCELEROMETER
- CAMER A

- Applications
- NFC
- Telephony
- FRAMEWORK /java/android/JINI
- DALVIK
- LIBRARIES
- NFC
- RIL
- LINUX KERNEL /dev/nfc /dev/baseband
- NFC CTRL
- BASE BAND

Pascal Urien, Porquerolles, 28 mai 2013
## Proprietary Libraries

<table>
<thead>
<tr>
<th>Hardware Component</th>
<th>Company</th>
</tr>
</thead>
<tbody>
<tr>
<td>Orientation sensor</td>
<td>AKM</td>
</tr>
<tr>
<td>Wi-Fi, Bluetooth, GPS</td>
<td>Broadcom</td>
</tr>
<tr>
<td>Graphics</td>
<td>Imagination</td>
</tr>
<tr>
<td>NFC</td>
<td>NXP</td>
</tr>
<tr>
<td>GSM</td>
<td>Samsung</td>
</tr>
</tbody>
</table>
RIL Details

Telephony services (ingoing call, outgoing calls are managed through RIL packets)

http://www.kandroid.org/online-pdk/guide/telephony.html
2011, Open Mobile API
Open Mobile API & Security Policy

- The API defines a generic framework for the access to Secure Elements in a mobile environment. It is based on four main objects.
  - The **SEService** is the abstract representation of all SEs that are available for applications running in the mobile phone.
  - The **Reader** is the logical interface with a Secure Element. It is an abstraction from electronics devices which are needed for contact (ISO 7816) and contactless (ISO 14443) smartcards.
  - The **Session** is opened and closed with a Reader. It establishes the logical path with the SE managed by the Reader.
  - The **Channel** is associated with an application running in the SE and identified by an ID (the AID= Application IDentifier)

- In order to protect the USIM from a non-authorized Android application, an access control (AC) mechanism based on the PKCS#15 standard is used.
  - The PKCS#15 repertory (hosted by the SIM) contains three files defined for the access rules
  - The Access Control Main File (EF-ACMF) gives a reference to the Access Control Rules File (EF-ACRF)
  - The Access Control Rules File (EF-ACRF) stores a list of Access Control Conditions File (EF-ACCF), each of them being associated to a particular AID.
  - Each Access Control Conditions File (EF-ACCF), contains a SHA1 digest of the mobile application whose access to embedded software running in the Secure Element (and identified by its AID) is authorized.
The Emergence of the Cloud Of Secure Elements (CoSE)
About NFC Payments

• Some NFC payments are based on the MasterCard PayPass specification
• There is two modes
  – Mag Stripe, a four digits CVC3 (Card Verification Value) is computed from a 3xDES and various parameters (PAN, ATC counter,...)
  – Contactless EMV
• The Secure Element securely performs calculations or runs the EMV application
• Contactless payments introduce a new paradigm, the virtualization of the bank card.
• The merchant terminal doesn’t known where is running the payment application on the mobile side.

* MasterCard® PayPass™ M/Chip, Acquirer Implementation Requirements, v.1-A4 6/06

Where is running the SE application?

Pascal Urien, Porquerolles, 28 mai 2013
Some Details with EMV Mag. Stripe*

- SELECT 2PAY.SYS/DDFO1
- SELECT MasterCard Google Prepaid Card
- GET PROCESSING OPTIONS
- READ first record
- COMPUTE Cryptographic Checksum

G=A23FB35FC89AE3A9 23358939AFBFCAEA 2335893905152040
CVC3=233

*Visa Contactless Payment Specification Version 2.0.2 July 2006

Pascal Urien, Porquerolles, 28 mai 2013
Google Wallet 2

Google Acquirer

Google Issuer

Customer’s Cards

Card Not Present transaction (CNP)

Card Network

Acquirer Bank

Customer’s Issuer Bank

Google Virtual prepaid card

MasterCard

Pascal Urien, Porquerolles, 28 mai 2013
About Relay Attack

• In 2005, G. Hancke introduced the concept of the “Relay Attack”
• The basic idea is that a reader working with ISO14443 device, reads a fake card (the proxy) which is connected via radio to an other device (the mole) working with a legacy card.
• As a result the reader manages a session with a remote device.

“A Practical Relay Attack on ISO 14443 Proximity Cards” Gerhard Hancke, 2005

“Keep Your Enemies Close: Distance Bounding Against Smartcard Relay Attacks”, Saar Drimer and Steven J. Murdoch, 2007
Where is located a bank card?

- In the SIM/USIM
  - Use SWP for NFC communication
  - MNO model
- In a NFC SecureSD
  - Tyfone, DeviceFidelity
- In a NFC Controller
  - The Google Model
- Somewhere in the Cloud
  - GoogleWallet2
  - SimplyTapp
  - EtherTrust

2012 Google fees

<table>
<thead>
<tr>
<th>Amount Range</th>
<th>Fee Percentage</th>
<th>Additional Fee</th>
</tr>
</thead>
<tbody>
<tr>
<td>Less than $3,000</td>
<td>2.9%</td>
<td>+ $0.30</td>
</tr>
<tr>
<td>$3,000 - $9,999.99</td>
<td>2.5%</td>
<td>+ $0.30</td>
</tr>
<tr>
<td>$10,000 - $99,999.99</td>
<td>2.2%</td>
<td>+ $0.30</td>
</tr>
<tr>
<td>$100,000 or more</td>
<td>1.9%</td>
<td>+ $0.30</td>
</tr>
</tbody>
</table>

2017 Forecasts
- Mobile Payments 1300 billions $
- NFC payments 200 billions $

Mobile Payment Market Growth
A cloud of secure elements (CSE) comprises the following five elements:

- Applications (typically written in javacard) stored in secure elements.
- Grids of secure elements (GoSE). Secure Elements embed Issuer Security Domain, which manage the lifecycle of applications. Applications may move from a grid to another.
- A Relay-Protocol (RP) enforces security between the GoSE and the NFC proxy, thanks to a secure channel, such as TLS.
- The NFC Secure Proxy (NSP) controls the session with the NFC reader (or initiator) and the dialog with the GoSE according to the relay protocol. This software entity should manage a SE located in the smartphone.
- A NFC reader (or NFC initiator) is used by legacy applications (payment, transport,...); however, future services could work with the P2P mode.
Experimental Platform
<table>
<thead>
<tr>
<th>Request</th>
<th>Data Size</th>
<th>Contact</th>
<th>Relay</th>
<th>Replay</th>
<th>Relay - Relay - Contact</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Select AID</td>
<td>7</td>
<td>47</td>
<td>221</td>
<td>67</td>
<td>60</td>
</tr>
<tr>
<td>2 Read Result</td>
<td>41</td>
<td>47</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3 Read Record 1, File 1</td>
<td>7</td>
<td>31</td>
<td>150</td>
<td>20</td>
<td>36</td>
</tr>
<tr>
<td>4 Read record 1, File 1</td>
<td>32</td>
<td>63</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 Read Record 2, File 1</td>
<td>7</td>
<td>47</td>
<td>190</td>
<td>20</td>
<td>45</td>
</tr>
<tr>
<td>6 Read Record 2, File 1</td>
<td>36</td>
<td>78</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7 Read Record 3, File 1</td>
<td>7</td>
<td>47</td>
<td>121</td>
<td>20</td>
<td>54</td>
</tr>
<tr>
<td>8 Select AID</td>
<td>7</td>
<td>47</td>
<td>210</td>
<td>40</td>
<td>45</td>
</tr>
<tr>
<td>9 Read result</td>
<td>60</td>
<td>78</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10 Get Processing Options</td>
<td>7</td>
<td>187</td>
<td>330</td>
<td>40</td>
<td>72</td>
</tr>
<tr>
<td>11 Read Result</td>
<td>23</td>
<td>31</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12 Read Record 1, File 1</td>
<td>7</td>
<td>16</td>
<td>120</td>
<td>40</td>
<td>2</td>
</tr>
<tr>
<td>13 Read Record 1, File 1</td>
<td>30</td>
<td>62</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14 Read Record 2, File 1</td>
<td>7</td>
<td>31</td>
<td>220</td>
<td>70</td>
<td>25</td>
</tr>
<tr>
<td>15 Read Record 2, File 1</td>
<td>65</td>
<td>94</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16 Read Record 1, File 2</td>
<td>7</td>
<td>31</td>
<td>368</td>
<td>50</td>
<td>85</td>
</tr>
</tbody>
</table>

EMV transaction timings, Intranet

Network Cost
\(<T_{cost} = T_{relay} – T_{contact} – T_{replay}> = 51\) ms

Cache Operation
\(769 – 60 – 62 + 820 + 560 = 2027\) ms

Pascal Urien, Porquerolles, 28 mai 2013
Test with EMV magstripe profile

One TCP packet per ISO7816 command (APDU), RTT 70 ms (Paris – Hanover)

Network Cost (Internet) 
\[ T_{cost} = T_{relay} – T_{grid} – T_{replay} \geq 83 \text{ ms} \]

\[
T_{rc} = T_{r0} + L \times D \\
\text{where } T_{r0} \text{ is a fix delay (about 20-40 ms), } L \text{ is the length of exchanged data, } D \text{ is the NFC throughput (104 Kbits/s in our platform, i.e. } D = 0,1\text{ms/byte)}
\]

\[
T_{rr} = T_{rc} + RTT + T_{se},
\]

Where \( T_{rc} \) is the time consumed by the NFC proxy, \( RTT \) is the round trip time over internet (ranging between 50ms to 100ms), and \( T_{se} \) is a time consumed by a legacy secure element for the request (such as 440ms for DDA or 420ms for GenerateAC).
Security for NFC

LLCPS

urn:nfc:sn:tls:snep
2012 LLCPS Platform

NDEF
SNEP
TLS
LLCP
NFCIP-1

NFC-Controller(s)

OpenSSL

TLS
LLCP
SNEP
NDEF

www.youtube.com/watch?v=CVWHlxoi3eU

Pascal Urien, Porquerolles, 28 mai 2013
2013, First Tests With BB 10

http://www.youtube.com/watch?v=CWS41clZylw
The NFCIP-1 layer is usually running in a microcontroller chip (such as the PN532 from NXP) that drives the NFC radio. An NFC session occurs in four logical steps.

1) **Initialization and Anti-collision**, the Initiator periodically probes the presence of a Target.

2) **Activation and Parameters Selection**, once a Target has been detected a set of parameters are notified or negotiated; in particular LLCP services are selected.

3) **Data Exchange**, frames are exchanged via the Data Exchange Protocol (DEP); the Initiator sends (DEP) requests acknowledged by Target responses; the packets size ranges from 64 to 256 bytes; DEP provides error recovery mechanisms, so upper layers such as LLCP, exchange error free packets.

4) **De-Activation**, the initiator can release the NFC session at any time, via Release-Request/Response messages.
The LLCP Protocol looks like a light version of the IEEE 802.2 LLC standard.

LLCP packets include a mandatory two bytes header comprising the DSAP (Destination Service Access Point, 6 bits), the SSAP (Source Service Access Point, 6 bits) and the PTYPE (4 bits) indicating the class of the PDU (Protocol Data Unit).

LLCPS works in connected mode and uses the "com.ietf.tls.snep" service name, which identifies a SNEP service over TLS. It only deals with seven PTYPEs:

- **CONNECT** (connection to the "com.ietf.tls.snep" service),
- **CC** (Connection Confirm),
- **DISC** (Disconnect),
- **DM** (Disconnected Mode),
- **INFORMATION** (TLS messages in our use case),
- **RR** (Receiver Reader, i.e. the acknowledgment of an INFORMATION PDU),
- **SYMM** (Symmetry) that indicates an inactivity over LLCP and avoids timeout at the DEP level.
The LLCPS layer manages five exclusive processes in order to exchange TLS messages.

- **The connection process (CP)** and the **disconnection process (DP)** are in charge of establishing and releasing LLCP sessions with the "com.iotf.tls.snep" service.

- **The sending process (SP)** sends a requested amount of data according to a simple strategy that performs segmentation, transmits INFORMATION PDUs and waits for acknowledgments (RR).

- **The receiving process (RP)** waits for a requested amount of incoming data; the reception of each incoming INFORMATION packet is acknowledged by a RR PDU.

- **The inactivity process (IP)** periodically generates SYMM symbols which may be echoed by other processes such as IP, SP or RP. SYMM generation is a consequence of slow TLS processing by a secure element, or interaction between the mobile operating system and its user.

Pascal Urien, Porquerolles, 28 mai 2013
SNEP - Simple NDEF Exchange Protocol & NDEF - NFC Data Exchange Format

- Once a TLS session is established, SNEP packets are securely exchanged.
- In our demonstration we use only SNEP-Put and SNEP-Success packets.
- A value, i.e. as a key encoded according to the NDEF format, is pushed from the target to the virtual lock

SNEP Put Packet

10  SNEP Version
02  Put
00 00 00 0E  Payload Length

NDEF Record :
(NFC Text Record Type Definition)

D1: 1 1 0 1 0 001
01: Type Length
0A: Payload Length
54: Type= ‘T’, Text
02: ID= UTF8
65 6E: “EN”
53 61 6D 70 6C 65 20: "Sample "

Pascal Urien, Porquerolles, 28 mai 2013
Thank You

Questions

Pascal Urien, Porquerolles, 28 mai 2013