

La carte à puce

Damien Sauveron
damien.sauveron@unilim.fr
<http://damien.sauveron.fr/>

(et tous les auteurs cités en bibliographie)

1

Intervenants dans le module

Damien Sauveron (Cours, TD et TP) :

- Membre de l'équipe « Cryptis » en charge du projet « Sécurité des Systèmes et Réseaux » (P.F. Bonnefoi, E. Conchon, K. Tamine)
- Thème de recherche : sécurité des systèmes embarqués et des réseaux mobiles

Pour me contacter :



Damien Sauveron
XLIM UMR 7252 CNRS -- Université de Limoges
123 avenue Albert Thomas
87060 Limoges Cedex, FRANCE

Email: Damien.Sauveron@unilim.fr
Web: <http://damien.sauveron.fr/>
Phone: +33 (0) 5 87 50 67 93

Délégation CNRS (2015/2016)

MCF à l'Université de Limoges depuis septembre 2006 – Habilité à Diriger des Recherches (2014)

Visite postdoctorale au Smart Card Centre de l'Information Security Group du Royal Holloway University of London

Thèse au LaBRI – Université Bordeaux 1 sur la Sécurité de la Technologie Java Card

Ingénieur R&D dans le CESTI de SERMA Technologies (Pessac)

Organisateur des conférences WISTP

Président du groupe de travail de l'IFIP : Pervasive Systems Security

2

Plan

La carte à puce et les concepts de base

Qu'est-ce qu'une carte ?

Historique

Deux classifications:

- Par technologie pour la puce
- Par technologie de communication

Standard ISO 7816

- Protocole de communication

Chaîne de fabrication

Acteurs

Applications

Les sécurités

Au niveau physique

Au niveau logicielle

Au niveau de l'environnement de production

Méthodes formelles & Test

Évaluation sécuritaire et certification

Les attaques

Non invasive

Invasive

Les cartes du futur

Bibliographie

Rendez vous

Java Card Grid Projet

3

Et si on se commençait par une vidéo ?

On ne rit pas !!!!

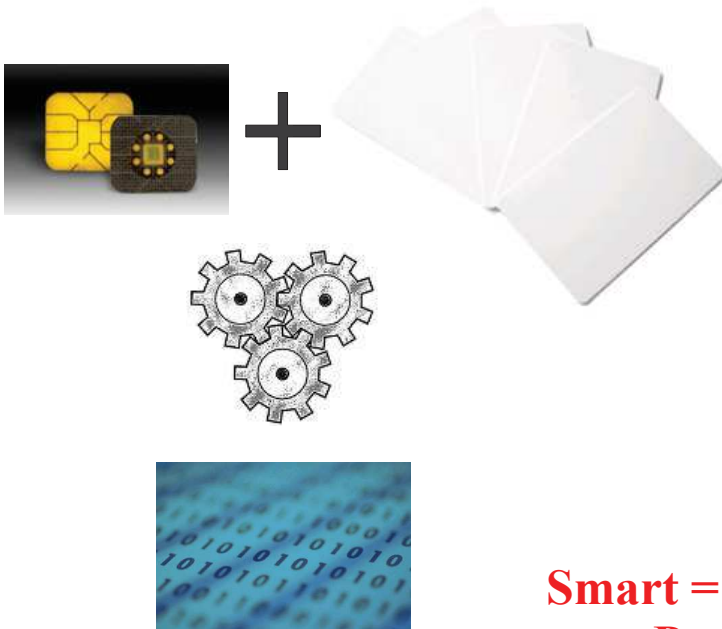
Roland Moreno La carte a puce (prod Le sabre).flv

4

Qu'est ce qu'une carte à puce

un morceau de plastique de la taille d'une carte de crédit

un circuit électronique capable de manipuler (stocker, calculer, etc) des informations



**Smart = intelligente !
Pourquoi ?**

5

Historique



**La carte à puce,
une innovation française ?**

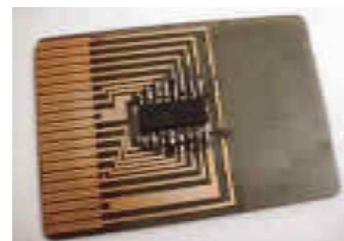
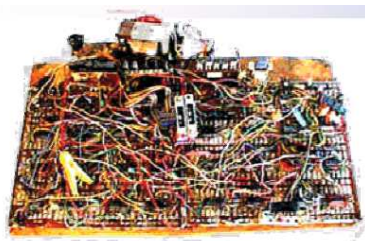
En France : OUI

Dans le reste du monde : NON

En 1967-1968, deux ingénieurs Allemands **Jürgen Dethloff** et **Helmut Grötrupp** introduisent un circuit intégré dans une carte plastique (déposent un brevet en 1969)

En mars 1970, Kunitaka Arimura au Japon dépose un brevet sur la carte à puce.

En 1971, Paul Castrucci de IBM dépose aux USA un brevet intitulé *Information Card*



6

Historique

Entre 1974 et 1978, le français Roland Moreno, le père de la carte à puce dépose 47 brevets dans 11 pays.
(créée ensuite la société **Innovatron**)

Implication industrielle de Bull et Schlumberger

Prototype de puce portable imaginé par Moreno :



11/06/1945 – 29/04/2012

En 1968, René Barjavel dans la “La nuit des temps”

« Chaque fois qu'un Gonda désirait quelque chose de nouveau, des vêtements, un voyage, des objets, il payait avec sa clé. Il pliait le majeur, enfonçait sa clé dans un emplacement prévu à cet effet et son compte, à l'ordinateur central, était aussitôt diminué de la valeur de la marchandise ou du service demandés. »

Mise au point de moyens inhibiteurs (par Moreno) logés sur la puce :

- comparaison interne du code confidentiel ;
- compteur d'erreurs, qui provoque l'autodestruction de la puce en cas de soumission répétée d'un code faux ;
- moyens de traitement ;
- lecture impossible de zones prédéterminées, notamment code confidentiel, clés, etc. ;
- écriture, modification, effacement impossibles de zones prédéterminées de la mémoire.

En 1997, Roland Moreno est entré au National Museum of American History.

7

Historique

En 1977, le Français Michel Ugon (de la compagnie Bull) propose de mettre les moyens inhibiteurs sur un micro-processeur.



Mars 1979, CII Honeywell Bull et Motorola :

Deux puces: une mémoire 2716 EPROM et un microprocesseur 8 bits 3870.

Octobre 1981, puce monolithique CII-Honeywell Bull
(CP8 : Circuit Portatif des années 80)

Historique

En 1979, la première carte est créée par Bull CP8 (1Ko de mémoire programmable et cœur à base de microprocesseur 6805).



En 1983, apparition des premières cartes téléphoniques à mémoire (commercialisation des premières télécartes par France Télécom fabriquées par Schlumberger)



En 1984, adoption par le G.I.E carte bancaire de la « carte bleue » (carte à microprocesseur)

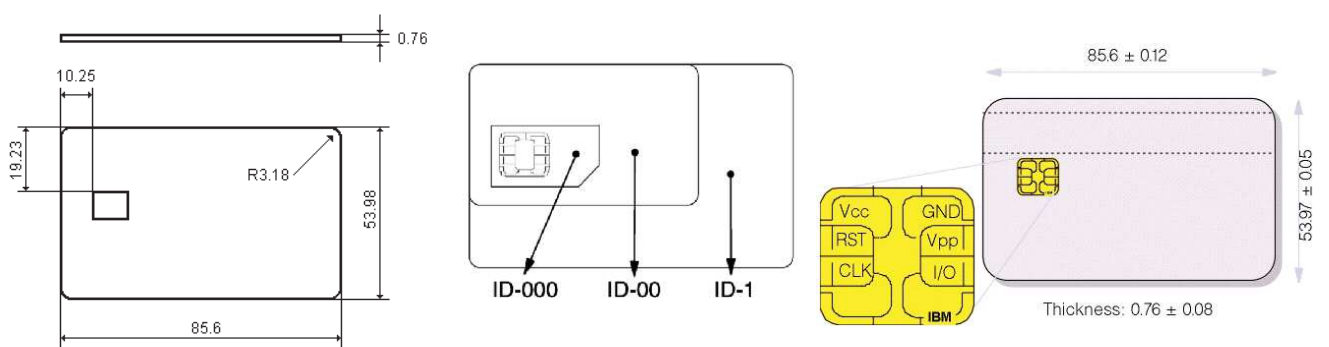


9

Historique

Entre 1984 et 1987, normes ISO 7816 (carte à puce à contact)

- **Objectif : Permettre aux cartes de fonctionner partout dans le monde !**



En 1997, apparition des premières Java Cards

- 30 ans après des nouveautés dans le monde de la carte. Enfin !!!



Premières entreprises de la carte

- La société Innovatron créée par Moreno pour exploiter ses brevets.
 - En 1979, Schlumberger (géant des services pétroliers) entre au capital d'Innovatron, pour 34 %.
 - Schlumberger devient le numéro 1 mondial de la carte à puce (cartes, lecteurs, cabines téléphoniques, systèmes),
 - En 1997, Schlumberger acquiert le concurrent français : SOLAIC.
 - En 2001, Schlumberger acquiert le concurrent français Bull CP8.
-
- En 1981, le GIE « Carte à Mémoire » lance trois expérimentations de la carte à puce avec: Bull, Philips, et Schlumberger.
 - En 1984, le GIE devient « Cartes Bancaires ».
 - Fin des années 80, GIE « CB » commande 16 millions de cartes CP8, généralisant l'utilisation de la carte à puce en France en 1992.

Vers les géants de la carte à puce

- En 1988, Marc Lassus crée Gemplus en France
- En 2001, l'activité « Cartes à puce » est cédée à Schlumberger qui externalise cette activité à Axalto
- En 2004, Schlumberger qui veut se recentrer sur les métiers du secteur pétrolier, se sépare d'Axalto en l'introduisant à la bourse de Paris.
- En 2006, fusion de Axalto et de Gemplus =>
Gemalto (numéro 1 mondial de la carte à puce)
- Exemples d'entreprises concurrentes aujourd'hui : Oberthur Technologies,
Sagem Orga

Quelques dates/1

Année	Événement
1979	Première carte fabriquée par Motorola pour Bull CP8
1981	Sortie de la première carte mono-composant à micro-circuit
1983	Premières télécartes françaises
1983	Le CNET spécifie la première carte bancaire à puce
1984	Naissance du GIE « Cartes Bancaires » successeur du GIE Carte à mémoire
1985	Le GIE CB commande 16 millions de cartes à puce
1987	Premières normes ISO
1989	Premières cartes GSM pour téléphones mobiles
1992	Le GSM devient un produit commercial

jeanla.boutefrance@enam.fr CEDRIC / CN

13

Quelques dates/2

Année	Événement
1992	Toutes les cartes CB sont dotées d'un micro-circuit
1994	Développement de la carte de SS en Allemagne (près de 70 millions en 1997)
1995	Début du projet « SESAM Vitale »
1996	Développement important du porte-monnaie électronique
1999	Généralisation de la carte vitale (près de 40 millions)
1996	Premières normes EMV
2004	Le système CB passe au standard international EMV
2004	Premières CB sans contact

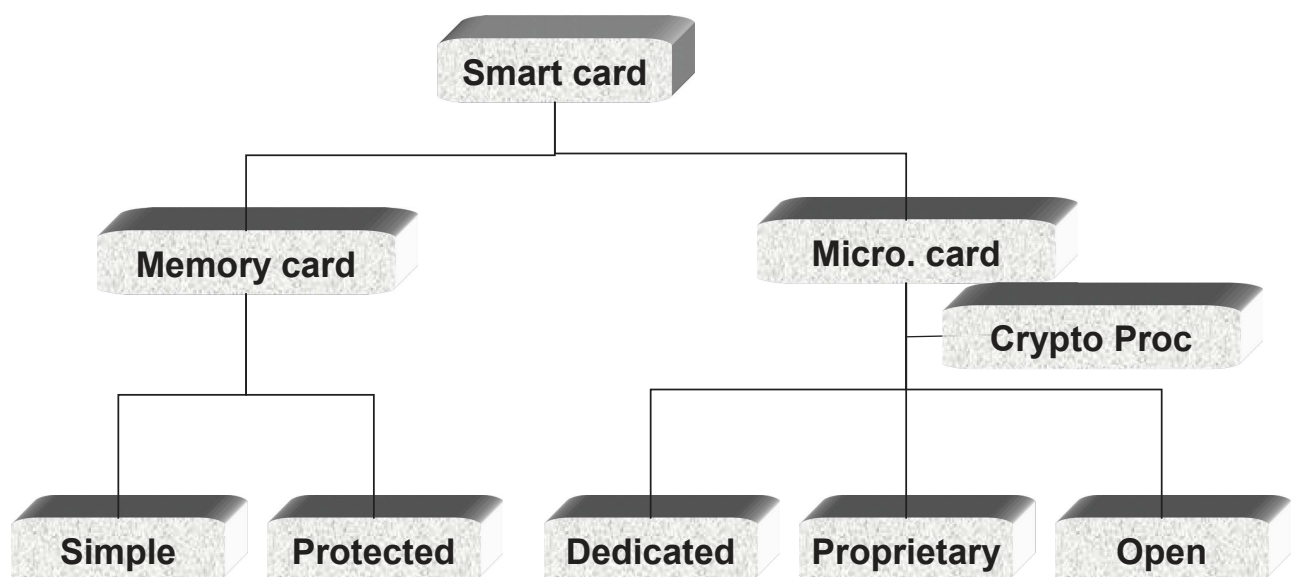
jeanla.boutefrance@enam.fr

14

**les cartes à mémoire
versus
les cartes à microprocesseur**

**les cartes à contact
versus
les cartes sans contact
versus
les cartes dual-interface**

15



16

La carte à mémoire

Premier modèle de cartes à puce

Majorité des cartes vendues dans le monde en 1999 (on va voir quelques chiffres plus récents)

Elle possède :

- une puce mémoire de 1 à 16 Ko
- une logique câblée non programmable

Avantages :

- sa technologie simple
- son faible coût (1€)

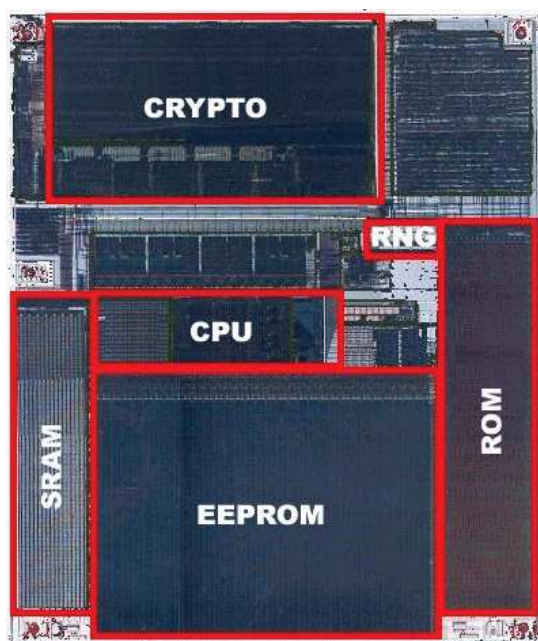
Inconvénients :

- sa dépendance vis-à-vis du lecteur de carte
- « assez » facile à dupliquer

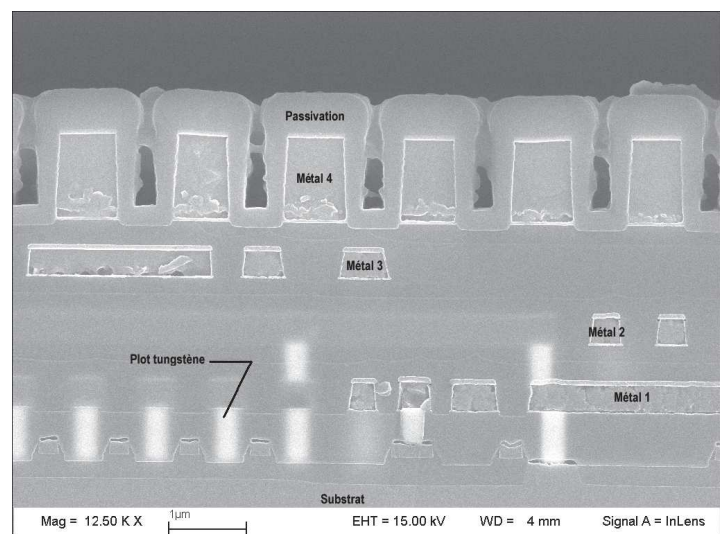
17

La carte à microprocesseur

Taille de la puce : $25\text{mm}^2 * 200\mu\text{m}$ ($6\mu\text{m}$)



Vue de dessus



Vue en coupe

Coût : entre 1€ et 20€ (acceptable pour tant de sécurité).

18

La carte à microprocesseur

Microprocesseur : 8, 16 ou 32 bits (à architecture CISC ou RISC)

ROM : 32 à 256 Ko

Stocke le système d'exploitation et des données permanentes
Figée en usine

EEPROM : 32 à 256 Ko

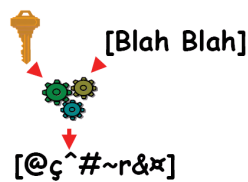
Mémoire persistante => stocke les données applicatives
Problèmes : durée de vie limitée et temps d'accès lent

RAM : 1 à 4 Ko

Mémoire de travail

Avantages : durée de vie illimitée et temps d'accès rapide

Coprocasseur cryptographique



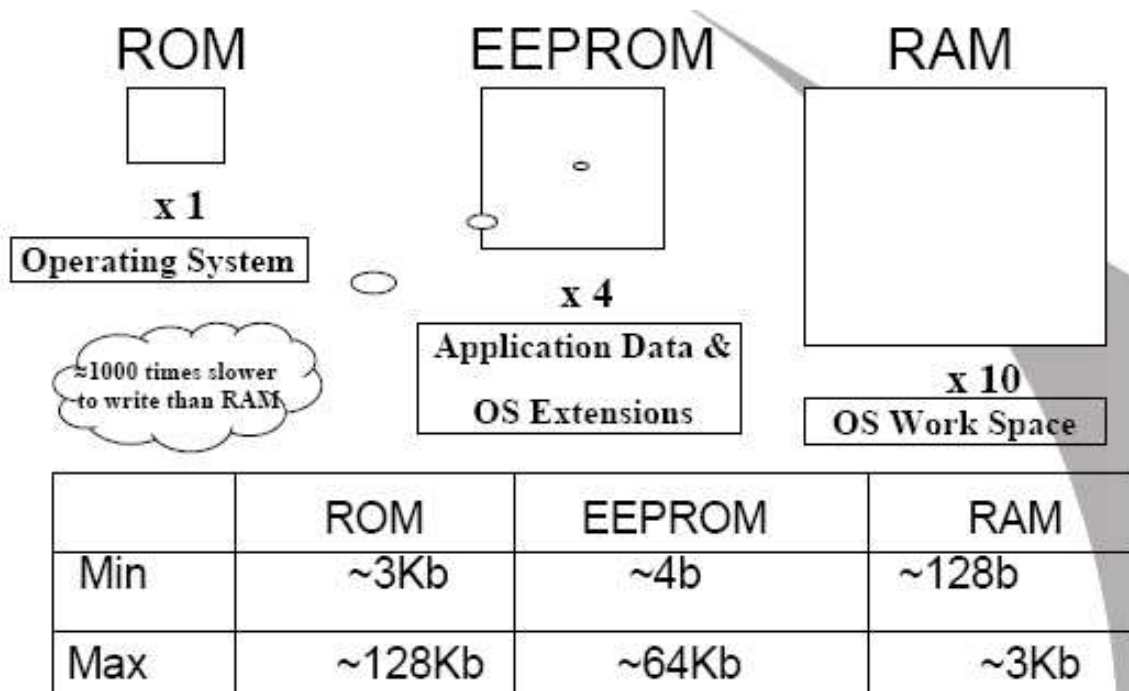
Générateur de nombres aléatoires (RNG)

Vers de nouvelles mémoires (+ rapide, + grosse) : FeRAM, Flash, MRAM



19

Les différents types de mémoires



EPROM : Write once, read FOR EVER !
– Used for initialization area (eg. Lock bytes)

20

Les différents types de mémoires

	RAM Statique	EEPROM	Flash-RAM	Fé-RAM (expérimentale)
Persistance	Volatile	Non Volatile	Non Volatile	Non Volatile
Tps Accès Read	0.1 μ s	0.15 μ s	0.15 μ s	0.1 μ s
Tps Accès Write	0.1 μ s	10 μ s	10 μ s	0.4 μ s
Tps Effacement (Reset)	Sans	5 000 μ s	100 000 μ s	Sans
Granularité Effacement	Sans	4 Octets	64 Octets	Sans
Nombre de Cycles garanti	Très grand	10^5 en Ecriture	10^5 en Ecriture	10^{10} en Ecriture/Lecture
Taille du point mémoire	> 100 μ m ²	> 30 μ m ²	< 10 μ m ²	< 10 μ m ²

21

Comparatif entre la carte et ...

L'ordinateur de pilotage d'Apollo 11 (1969)

- 2 MHz CPU
- 16 bits
- 2 Ko de RAM
- 36 Ko de ROM



Un PC de la fin des années 90

	Carte à puce	PC	Ratio
RAM	1 Ko	128 Mo	130000
Stockage	64 Ko	6 Go	100000
Connectivité	192 Kbits	100 Mbits	500
Microprocesseur	20 Mips	500 Mips	25

22

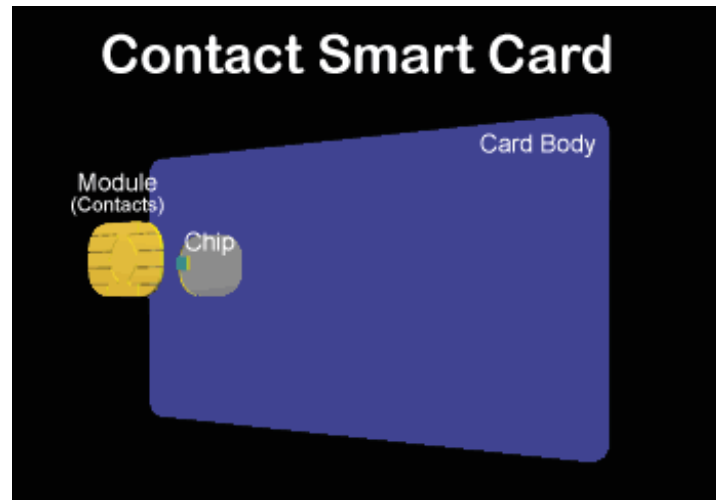
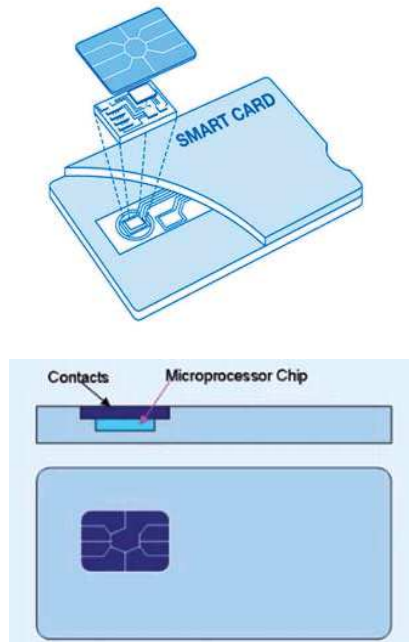
Carte à contact

Suit le standard ISO 7816

Communication série via huit contacts => insertion dans un lecteur de carte

Problèmes :

l'insertion et le retrait sont des facteurs d'usure de la carte
orientation de la carte dans le lecteur

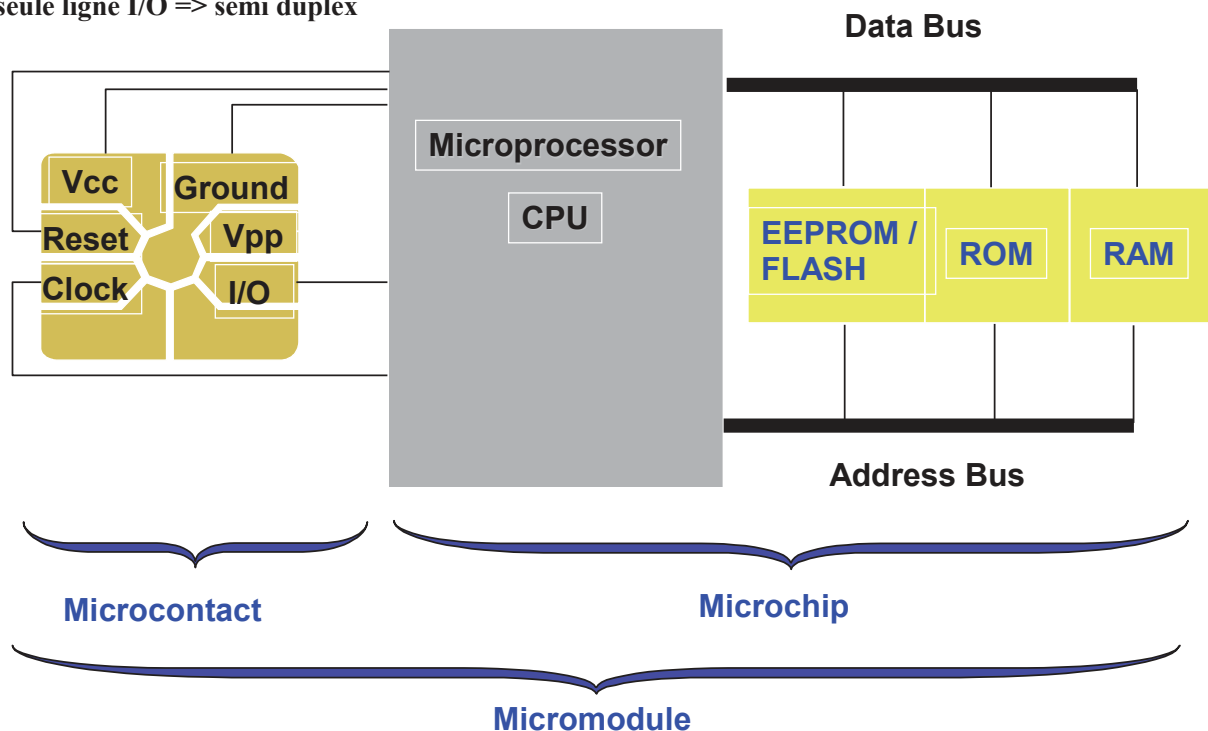


23

La carte à contact

Son fonctionnement, l'oblige à être insérée dans un lecteur de carte.
Elle utilise une communication série via huit contacts.

1 seule ligne I/O => semi duplex



24

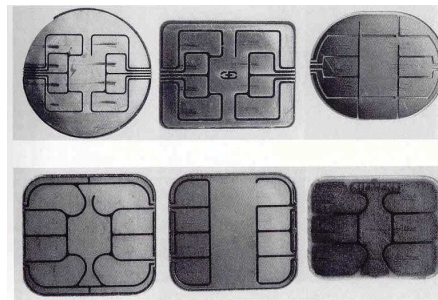
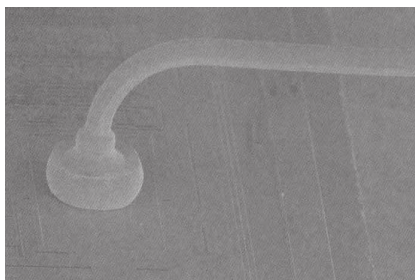
La carte à contact

Electrical connections between the chip and the module (wire bonding process)

8 contacts (C1-C8) but only 6 used

C6 used as V_{pp} while EEPROM where not embedding charge pump,

Supply voltage 2,7v (SIM) to 5,5v (standard TTL) and clock provided by the reader.



25

Carte sans contact

Communication via une antenne dans la carte

Récupère son énergie d'un couplage capacitif ou d'un couplage inductif

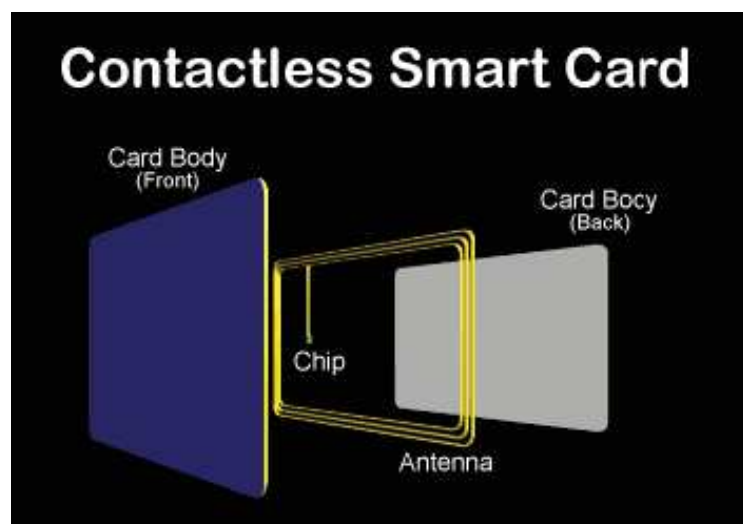
Suit le standard ISO 14443

Problèmes :

distance de communication limitée (environ 10 cm)

temps de transaction est de l'ordre de 200 ms => limite les données à échanger

le coût élevé



26

Carte sans contact

Need : modulator, demodulator, anti-collision mechanism, voltage regulator, reset generator and an antenna.

For data transfer all known digital modulation techniques can be used (ASK, FSK and PSK).

Standards : close coupling ISO/IEC 10536 (3-5MHz), proximity cards ISO/IEC 14443 (13,56MHz) and Hand Free Cards ISO/IEC 15693,

Used for public transportation, ski pass, access control, payment with GSM (NFC)...

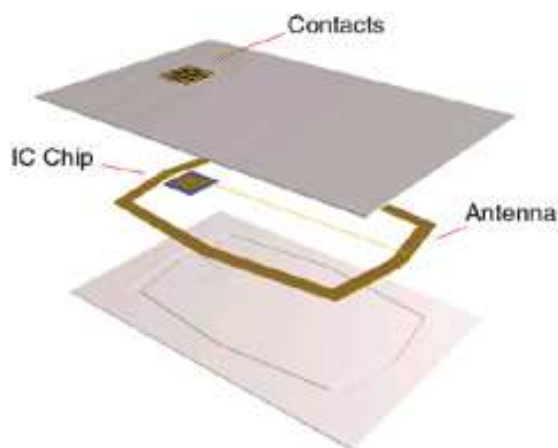
27

La carte combi

C'est une combinaison entre :

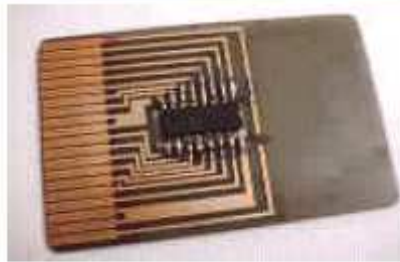
la carte à contact
et la carte sans contact

Ces deux possibilités de communication en font une carte « idéale ».



28

Quelques spécimens de mon Zoo



Cartes à Microprocesseur, 1987-2007



Les différentes formes et interfaces de communication

- Cartes avec contact (Module SIM GSM)
 - ISO 7810, 7816-1, 7816-2
 - USB ou OTG (USB pour mobile)
- Cartes sans contact
 - Plusieurs normes
- Cartes Hybrides
 - combinaison contact et sans contact
- Boutons
 - Produits iButton (bus DS 1-Wire)
 - JavaRing (iButton monté sur bague)
- Dongle
 - Série, parallèle, USB



Les terminaux cartes

- Contact / sans contact
- Simple / complexe (keypad, LCD, multi-slot, biometrie ...)

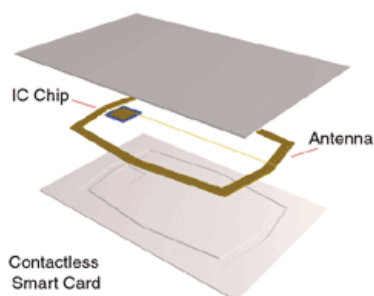


Quelques périphériques pouvant intégrer une puce sécurisée



Anneau Java

IButton



Dongles : Série, parallèle, USB

Quelques chiffres

1999 Card Shipments figures per sector

- Strong increase in microprocessor cards (+60%)
- Slight progression of memory cards (+14%)

	Memory (MU)	Microprocessor (MU)
Banking		108
Healthcare	27.5	30
Telecom	913	200
Transport	40.5	3
Pay TV / IT	1	29
Others	49	28
Total 1999	1031	398
	1429	

Worldwide Market 2004

Cards (Millions of units - Mu)			
By segments	Memory (MU)	Microprocessor	
			Growth % From 2003
Telecom	710	1050	57%
Financial services / Retail / Loyalty	35	280	37%
Government/ Healthcare	20	45	12%
Transport	60	15	25%
Pay TV	-	55	57%
Corporate Security	10	12	72%
Others	10	12	20%
Total H1 2004	845	1469	50%
Total 2004	2314		

33

Quelques chiffres

Worldwide Market 2005

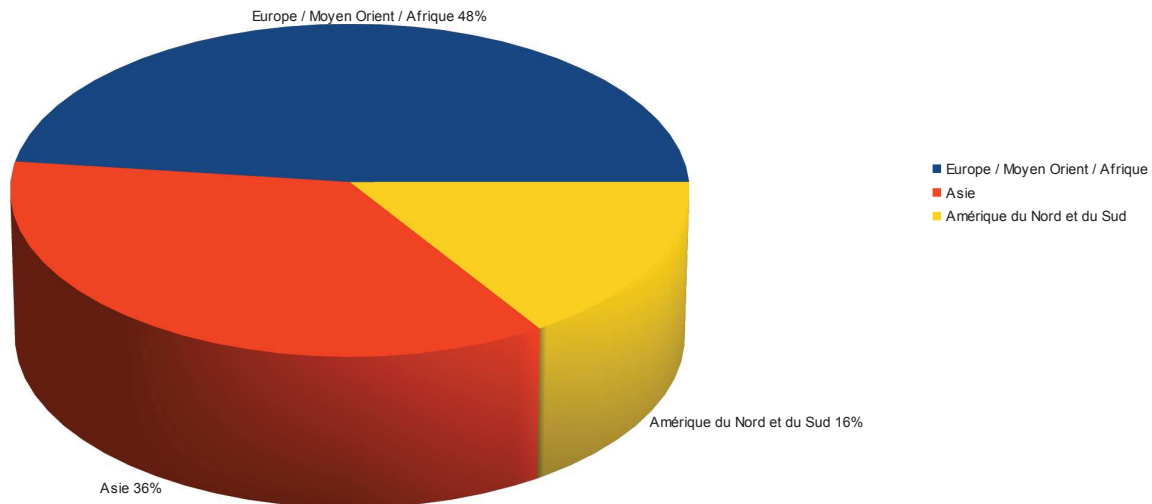
Cards (Millions of units - Mu)			
Sectors	Memory (MU)	Microprocessor	
			Growth % From 2004
Telecom	580	1390	32,38%
Financial services / Retail / Loyalty	30	336	20%
Government/ Healthcare	25	60*	33,33%
Transport	73	20	-
Pay TV	-	55	-
Corporate Security	20	15	-
Others	10	12	-
Total	738	1 888	28,52%
Total 2005	2626		

34

Répartition géographique du marché

Chiffres de 2005

Cartes %	
Zones Geographiques	à Microprocesseur
Europe / Moyen Orient / Afrique	48,00%
Asie	36,00%
Amérique du Nord et du Sud	16,00%
Total 2005	100,00%



35

Quelques chiffres

Smart Card shipments Global 2007

Cards (Millions of units - Mu)		
Sectors	Memory	Microprocessor
Telecom	440	2 600
Financial services / Retail / Loyalty	30	500
Government/ Healthcare	300	105
Transport	160	15
Pay TV	-	70
Corporate Security	20	20
Others	10	15
Total 2007	960	3325
Total 2007	4285	

36

Quelques chiffres

2007 Secure Contactless market overview (excluding access control and ticketing)

	Secure Memory Contactless (Mu)	Secure Microprocessor Contactless (Mu)
Financial Services	-	45
Government – Health care	250	45
Transport	160	15
Others	-	28
TOTAL 2007	410	133

37

Quelques chiffres

Worldwide Smart Secure Device shipment - 2011 and 2012 forecasts Millions of Units (Mu)

	2011 forecast	2012 forecast	2012 vs 2011 % growth
Telecom	4,600	5,100	11%
Financial services	1,010	1,200	19%
Government - Healthcare	240	300	25%
Transport	80	95	19%
Pay TV	125	140	12%
Others	80	90	13%
Total	6,135	6,925	13%

Telecom: including NFC secure elements (SWP UICC, μ SD, embedded SEs)

Secure Contactless market figures – 2010 and 2011 forecasts Millions of Units (Mu)

	2011 forecast	2012 forecast	2012 vs 2011 % growth
Financial services	225	290	29%
Government - Healthcare	125	160	28%
Transport	80	95	19%
Others	30	35	17%
Total	460	580	26%

38

Quelques chiffres

Eurosmart estimated worldwide Smart Secure Devices (microprocessors)
Millions of Units

	2012	2013 forecast	2013 vs 2012 % growth
Telecom	5.100*	5.350	5%
Financial services	1.200	1.480	23%
Government - Healthcare	310	360	16%
Transport	135	160	19%
Pay TV	135	145	7%
Others	90	100	11%
Total	6.970	7.595	9%

Eurosmart estimated worldwide Smart Secure Contactless Devices (microprocessor)
Millions of Units

	2012	2013 forecast	2013 vs 2012 % growth
Financial services	295	455	54%
Government - Healthcare	170	210	24%
Transport	135	160	19%
Others	60	70	17%
Total	660	895	36%

39

2014

Cartes (Millions d'unités – Mu)	
Secteur	à Microprocesseur
Télécommunications	5100
Services financiers / Fidélité	1800 (740)
Gouvernement / Santé	400 (240)
Autres (Transport, TV Payante, accès logique et physique)	410 (70) + (180 pour le transport)
Total prévision 2014	7710 (1230)

Cartes (Millions d'unités – Mu)	
Secteur	à Microprocesseur
Télécommunications	5200
Services financiers / Fidélité	2050 (880)
Gouvernement / Santé	380 (230)
Fabricant d'appareil	190
Autres (Transport, TV Payante, accès logique et physique)	400 (70) + (180 pour le transport)
Total réalisé 2014	8220 (1360)

40

Prévisions 2015

Worldwide Secure Elements shipment – 2013 & 2014 shipments and 2015 forecasts
(Million of units)

Source: Eurosmart, April 2015

WW shipments forecast	2013	2014	2015f	2015f vs 2014 % growth
Telecom	4,850*	5,200*	5400	3,8%
Banking	1,550	2050	2450	19,5%
Government	350	380	420	10,5%
Device Manufacturers	190	190	310	63,2%
Others**	390	400	430	7,5%
Total	7,140	8,220	9,010	9,6%

* Source SIMAlliance

**Others include Transport, PayTV and physical and logical access cards.

Worldwide Smart Secure Contactless market figures – 2013 & 2014 shipments and 2015 forecasts
(Million of units)

Source: Eurosmart, April 2015

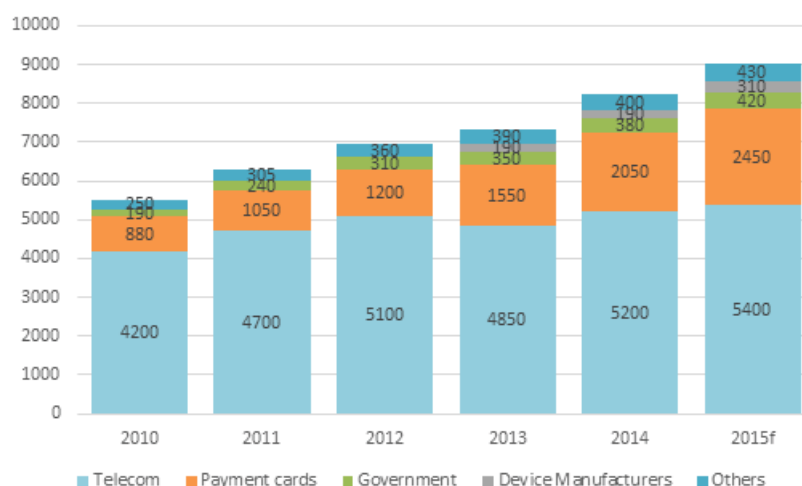
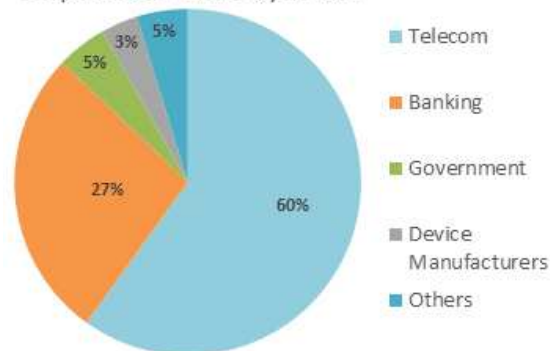
Of which contactless	2013	2014	2015f	2015f vs 2014 % growth
Financial services	590	880	1050	19,3%
Government – Healthcare	200	230	260	13,2%
Transport	160	180	210	16,7%
Others*	70	70	70	0%
Total	1,020	1,360	1,590	17%

*Others include Transport, PayTV and physical and logical access cards.

WW shipment forecast	2013	2014	2015f	2015f vs 2014 % growth
NFC Secure Elements	270	350	550	57,1%

41

Worldwide Secure Elements shipments in 2015 by Sector



42

Few attacks

- Smart card is a tamper resistant token not a tamper proof one.
- Attacks to the system can come from
 - Human error (e.g. entering wrong data)
 - Unintentional fraud (e.g. equipment failure)
 - Intentional fraud
 - Misuse of equipment
 - Passive attacks (e.g. listening without modifying)
 - Difficult to detect
 - Preventable
 - Active attacks (e.g. generation, modification of messages)
 - Generally detectable
 - Prevention difficult
 - etc.

Smart Card Fraud

- Scenario 1:

Although PIN protected stolen magnetic stripe credit cards were successfully used to withdraw money

Audit of the ATM's log file show that although the thief presented three false PIN code he could somehow get the card back and try again. The correct PIN was found by exhaustive search after approximately 5000 attempts.
- What happened ?

After stealing the card, the thief made a small hole in it, attached a wire to pull out the card after three false presentation

Smart Card Fraud

- Scenario 2

Users insert their cards to ATMs enter their PINs but get no money. The ATM swallows the card and display the message "*Invalid card contact your bank*".

Money was however withdrawn with the card later.

- What happened ?

A false ATM...

Smart Card Fraud

- Scenario 3

Same as the previous one but using smart card with an EEPROM having a retry counter limited to 3. The card is always returned to the user but its EEPROM retry counter never decrease.

The audit of the ATM's log file showed that although the thief presented three false PIN codes he could somehow try again and again. The correct PIN was found by exhaustive search after approximately 5000 attempts.

- What happened ?

In old cards EEPROM programming voltage was done using an external programming voltage (Vpp) supplied through a specific ISO contact. The thief had covered this contact with a sticker.

Smart Card Fraud

- Scenario 4:

The ATM's log file and cash do not match, money is missing

Audit of the ATM's log file showed that the same user withdrew money several times. He always forgot his banknotes that were swallowed back by the ATM after a short time-out (security features)

- What happened ?

The thief would withdraw three banknotes but take only two of them. The remaining banknote was detected by the paper sensor and swallowed back by the ATM which automatically cancelled the transaction.

The sensor could not distinguish between one, two or three banknotes.

Smart Card Fraud

- Scenario 5:

Users complain that when attempting to withdraw money they get nothing, money is however debited from their accounts

An audit of the ATM's log file shows nothing abnormal

- What happened?

The hole through which money was delivered was covered by a fake hole (piece of metal)

The back of the fake hole was covered with glue to prevent the machine from swallowing back forgotten banknotes.

After each victim's withdrawal, the thief would come to the ATM, remove the piece of metal and collect the banknote.

Skimming

- vendredi 07 novembre 2008 : un million retiré sur des comptes de l'Ouest (Ouest France)



Skimming

- Cout 8k\$
 - Wifi + sms
 - Hong kong, malaisie, USA, France



Smart Card Fraud

- Scenario 6:

Although PIN protected stolen smart card were successfully used to withdraw money.

An audit of the ATM's log file shows that the correct PIN was used in the withdrawal operation.

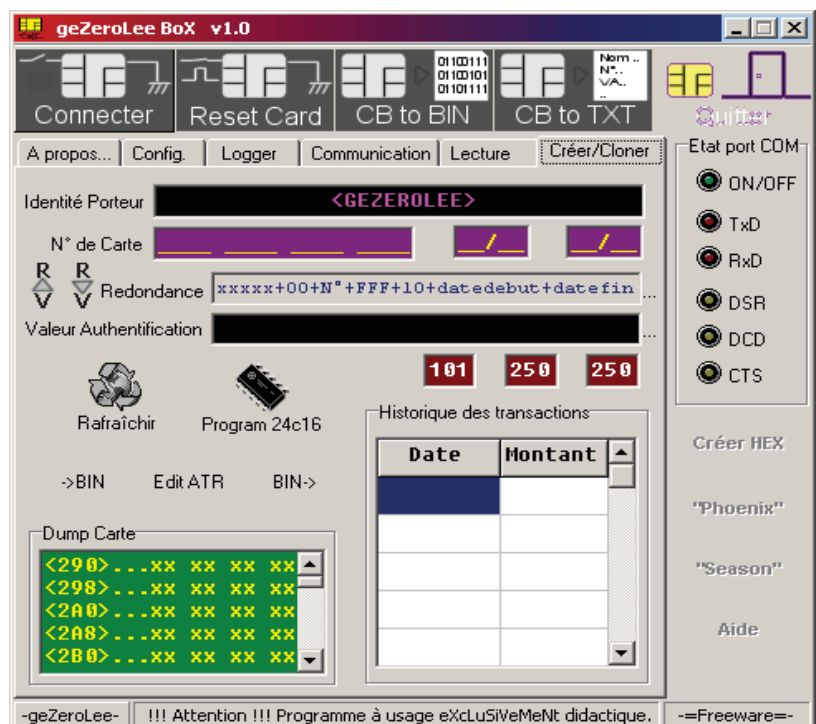
- What happened ?

The fraud was technical: the smart card's software was programmed to compare the presented PIN and if incorrect to increase the EEPROM counter.

EEPROM programming is characterized by an increased power consumption and requires 5ms.

The thief used a board that presented automatically all the PIN value (0000 to 9999) but detected the current consumption increase and powered off the card before the EEPROM retry counter could be updated.

Yescard



Le logiciel G0lee pour la fabrication de Yescard

Yescard

- Some details

Below a given threshold card and holder authentication are done locally,

Only some terminals (gasoline, transport ticket, video rental and so on...) are concerned

- ATM need always an on-line authorisation
- Merchant terminal will be detected by the merchant or he/she is himself part of the attack.

Yescard

- Context

Weakness known by industrial experts

- Off line authentication : public key
- On line authentication : secret key
- Ks stored into the card but easy to retrieve (key was only 320 bits)
- Cloning a card with forged Vs compatible with Id

Ended with the court case "Serge Humpich vs GIE-CB."

Keys have been broadcasted thanks to Usenet.

Card have migrated to EMV 5.1 and 5.2

Yescard

- Consequence

Media focused too much on cloning Banking Card

- Moved from a technical risk to industrial image

Problem related with knowledge broadcast

- Know-how used in a fraud context
- Do we need an internet based *full disclosure...*

Fraud conclusion

- All these flaws described here are at least nine years old,
- All of them of course have been corrected,
- Security is a permanent race...

ISO 7816 : le standard !

Partie 1 : Caractéristiques physiques

Partie 2 : Dimensions et position des contacts

Partie 3 : Signaux électroniques et protocoles de transmission

Partie 4 : Commandes inter-industrie pour l'échange

Partie 5 : Identificateur d'application

Partie 6 : Eléments de données inter-industrie

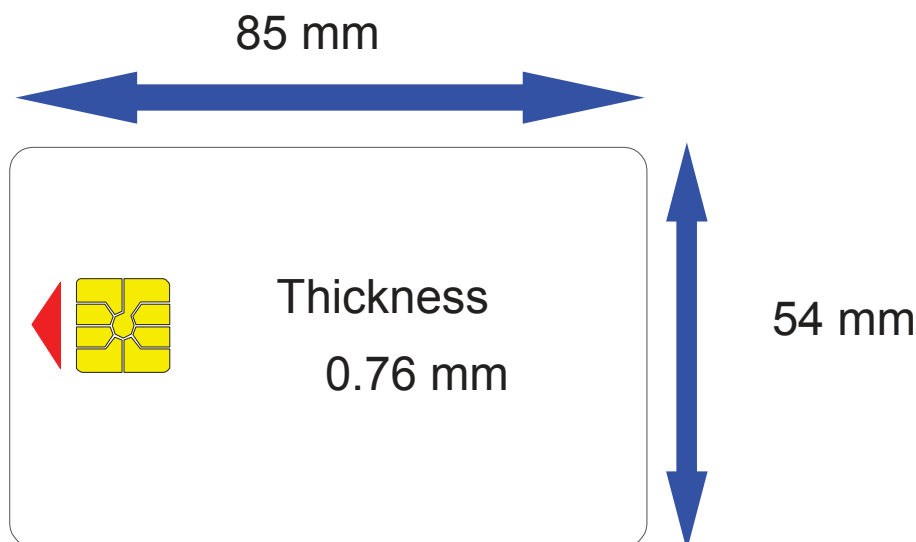
Partie 7 : Commandes inter-industrie pour SCQL (Structured Card Query Language)

Partie 8 : Sécurité de l'architecture et des commandes inter-industrie

57

ISO 7816-1

Définit les caractéristiques physiques : dimensions, flexibilité, résistivité.
Définit aussi les formats plus petits (SIM)



58

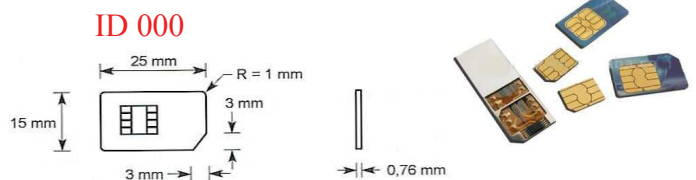
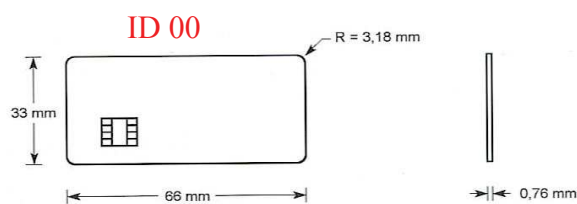
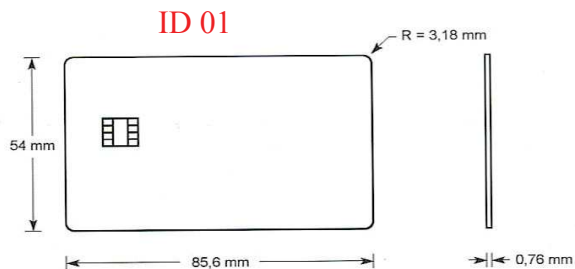
ISO 7816-1

Même si on connaît en général deux formats de la carte à puce

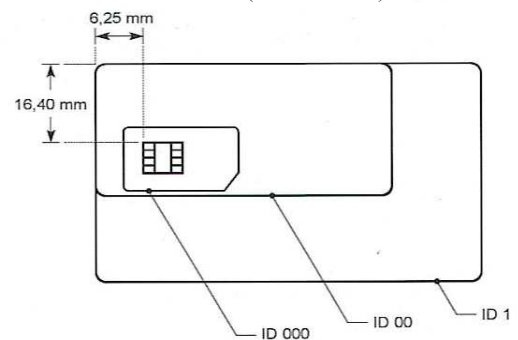
Celui de la carte bancaire

Celui de la carte SIM

3 formats normalisés : ID1, ID00 et ID000

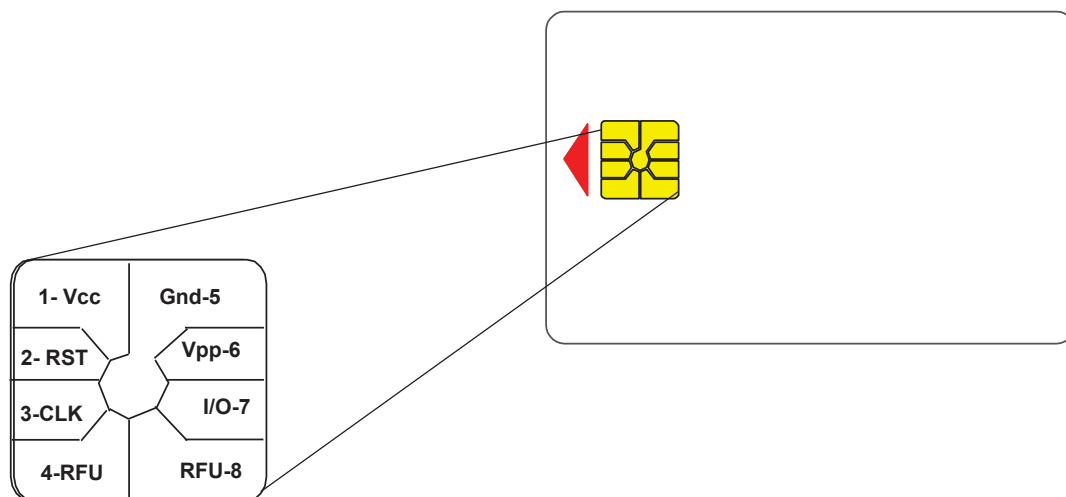


Le fabricant produit une seule taille (ID1), le client final pourra réduire ses dimensions au format ID00 ou ID000 (ex. carte SIM)



ISO 7816-2

Définit les aspects électriques et la position des contacts sur la carte.

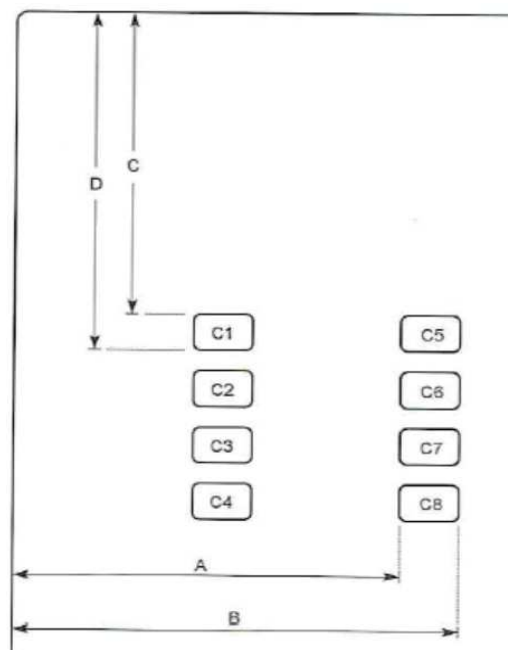


Position des contacts sur le support plastique

	A	B	C	D
C1	10,25	12,25	19,23	20,93
C2	10,25	12,25	21,77	23,47
C3	10,25	12,25	24,31	26,01
C4	10,25	12,25	26,85	28,55
C5	17,87	19,87	19,23	20,93
C6	17,87	19,87	21,77	23,47
C7	17,87	19,87	24,31	26,01
C8	17,87	19,87	26,85	28,55

Position ISO 7816

Valeurs en mm



61

Normalisation AFNOR / ISO

La position des contacts : position AFNOR et position ISO



Carte ISO

Carte AFNOR

62

ISO 7816-2

Vcc: tension d'alimentation positive de la carte fournie par le lecteur

(4.75V $\square$ Vcc $\square$ 5.25V) Vcc=3.3V pour une carte SIM

RST: c'est le « reset », initialise le microprocesseur (warm reset)

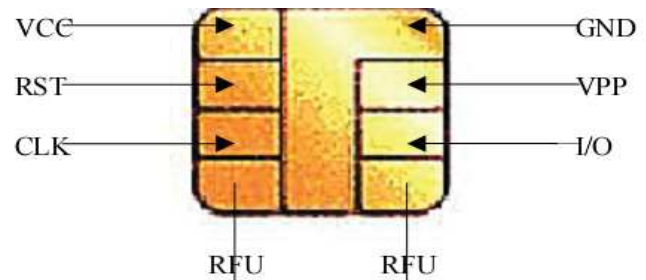
cold reset = coupure et rétablissement de l'alimentation

CLK: signal d'horloge fourni à la carte par le lecteur

rythme les échanges de données entre la carte et le lecteur

RFU (Reserved for Future Use) non utilisés

GND: masse électrique de la carte



I/O: utilisé pour le transfert des données et des commandes entre la carte et le lecteur. La communication est half-duplex.

Vpp: destiné à l'origine à recevoir la tension de programmation des mémoires non-volatiles EPROM. Cette tension était de 25 Volts à l'origine, puis 21 Volts, puis devenue inutile avec l'utilisation d'EEPROM qui possèdent leur propre générateur de tension de programmation sur la puce à partir du 5 Volts (Vcc). **Utilisé pour SWP (full duplex)**

63

ISO 7816-3

Cette partie normalise :

les protocoles de transmission (TPDU : Transmission Protocol Data Unit) :

- T=0, protocole orienté octet
- T=1, protocole orienté paquet
- T=14, réservé pour les protocoles propriétaires

la sélection du type de protocole (PTS : Protocol Type Selection)

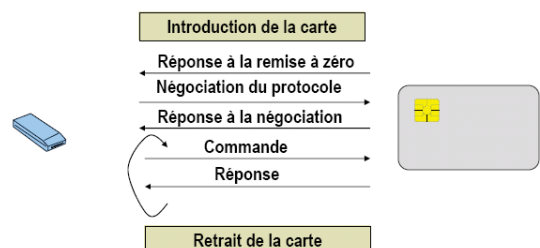
la négociation des paramètres du protocole (PPS : Protocol Parameter Selection)

la réponse au reset (ATR : Answer To Reset) qui correspond à la mise en route du prog ROM de la carte

- Min. 2 et Max. 33 caractères et 5 champs
- Permet de fixer :

Les conventions de codage des octets
Le temps de transmission d'un bit
La valeur de la tension de programmation
Le protocole de communication
Un historique qui s'affichera à la mise sous tension de la carte (ex: version de l'OS)

La carte n'est jamais l'initiateur de la communication



les caractéristiques électriques comme :

- la fréquence d'horloge (entre 1MHz et 5MHz)
- la vitesse des communications (jusqu'à 115200 bauds)

64

Exemples d'ATR

Carte Santé Vitale

ATR = 3F 65 25 00 2C 09 69 90 00

Carte Bancaire CB

ATR = 3F 65 25 08 36 04 6C 90 00

Carte Verte Monéo

ATR = 3B E6 00 FF 81 31 42 45 19 16 01 01 27 B1 37

Carte Cinema (perimé)

ATR = 3B 23 00 35 13 96

Carte GSM Itineris

ATR = 3F 2F 00 30 AF 59 02 01 02 80 00 17 0A 0E 83 1E 9F 16

GXP211_PKIS

ATR = 3F 6D 00 00 80 31 80 65 B0 05 01 02 5E 83 00 90 00

GemSafe

ATR = 3B A7 00 40 18 80 65 A2 08 01 01 52

Schlumberger Palmera

ATR = 3B 65 00 00 9C 02 02 06 01

Cyberflex Access e-gate 32K

ATR = 3B 75 94 00 00 62 02 02 00 80

65

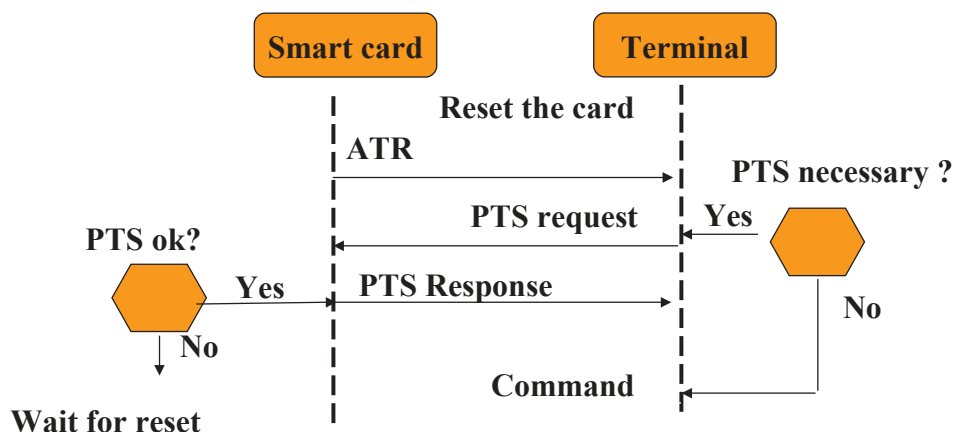
Protocol Type Selection

Needed only if the terminal wants to modify parameters,

If the card agrees, it sends the PTS back to the terminal

Otherwise the terminal execute a reset (warm => protocol change),

Only one PTS after the ATR.



66

Comportements de la carte et du lecteur lors d'un Reset

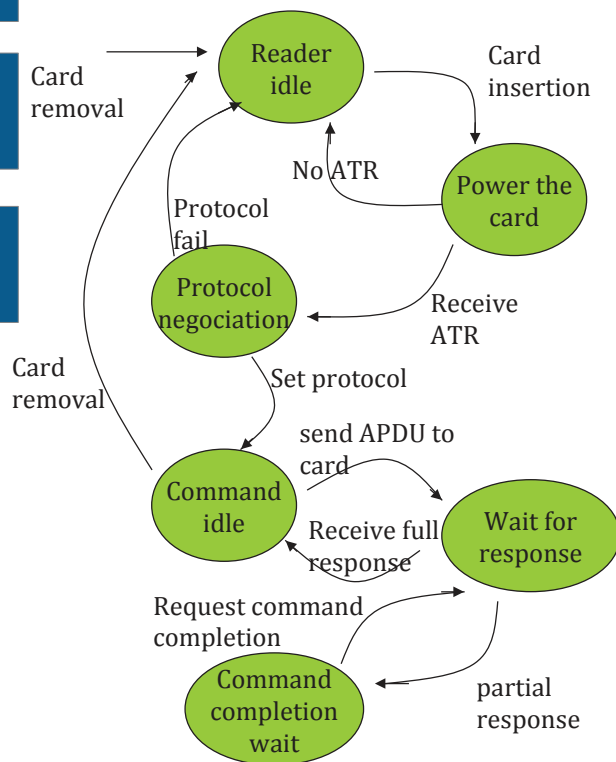


Diagramme d'état du lecteur

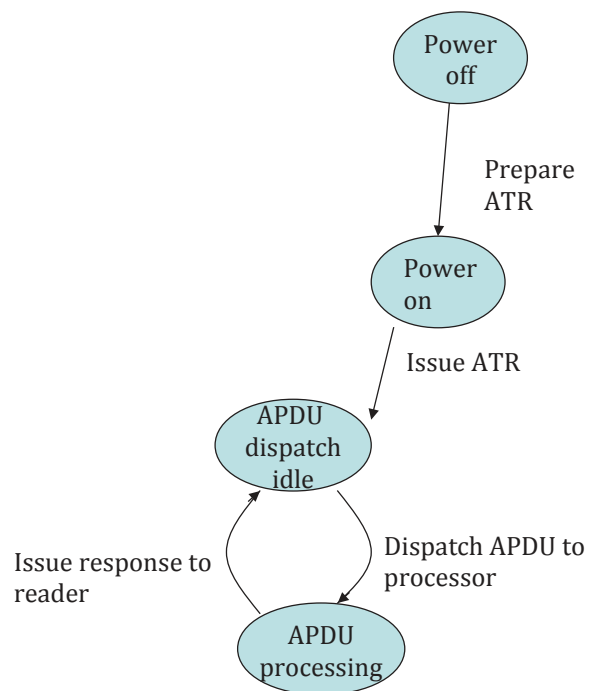


Diagramme d'état de la carte

67

Transmission protocols

T=0 most widely used (1989), **T=1** block oriented
T=14 Japan and Germany

Transmission protocol	Meaning	ISO
T=0	Asynchronous, half duplex, byte oriented	7816-3
T=1	Asynchronous, half duplex, block oriented	7816-3
T=2	Asynchronous, full duplex, block oriented, tbs	10536-4
T=14	National functions	No ISO

Transport protocols

T=0

Byte oriented, Serial transmission (1 start bit, 8 bits data, 1 parity bit, 2 stop bits)

Transmission error (parity only) 2 etu mute ("0")

T=1

Block oriented, Header : NAD, PCB, LEN; data : INF, CRC.

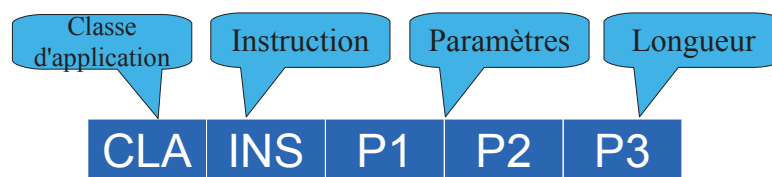
NAD 3 bits destination address, 3 bits source address

PCB define the kind of block

- I (#block, more) numbered mod 2, more = 1, another block follow
- R(#block, error) numbered mod 2, next expected bloc,
- S specific command (RESYNC, IFS, ABORT, WTX)

69

T=0 Structure d'une commande/réponse



Commande

CLA est normalisé (par exemple FF est utilisé pour le PPS)

INS ne doit pas être 9x ou 6x et doit être pair pour l'ISO

Réponse

Les valeurs sont normalisées

- 90 00 : succès
- 6E xx : classe inconnue
- 6D xx : instruction inconnue
- ...



70

CLA Class byte

b7 to b4	b3	b2	b1	b0	Meaning
			X	X	Logical channel number
	0	0			No secure messaging
	1	0			Secure messaging header not authentic
	1	1			Secure messaging header authentic
'0'					Structure and coding compliant with 7816-4
'8','9'					User specific codes
'A'					Structure and code defined in additional document GSM11.11

Class	Application
'80'	Electronic purse compliant with EN 1546-3
'8x'	Credit card compliant with EMV-2
'A0'	GSM compliant with prETS 300 608/GSM 11.11

71

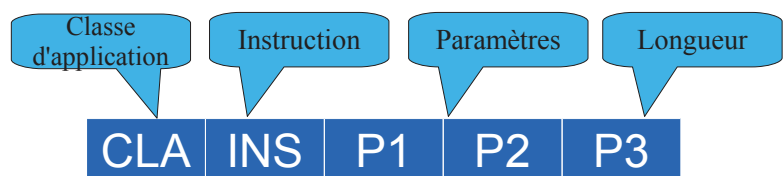
Protocole T=0 (Transmission semi-duplex de caractères asynchrones)

Commande entrante (envoi des données à la carte)

Commande sortante (récupère des données de la carte)

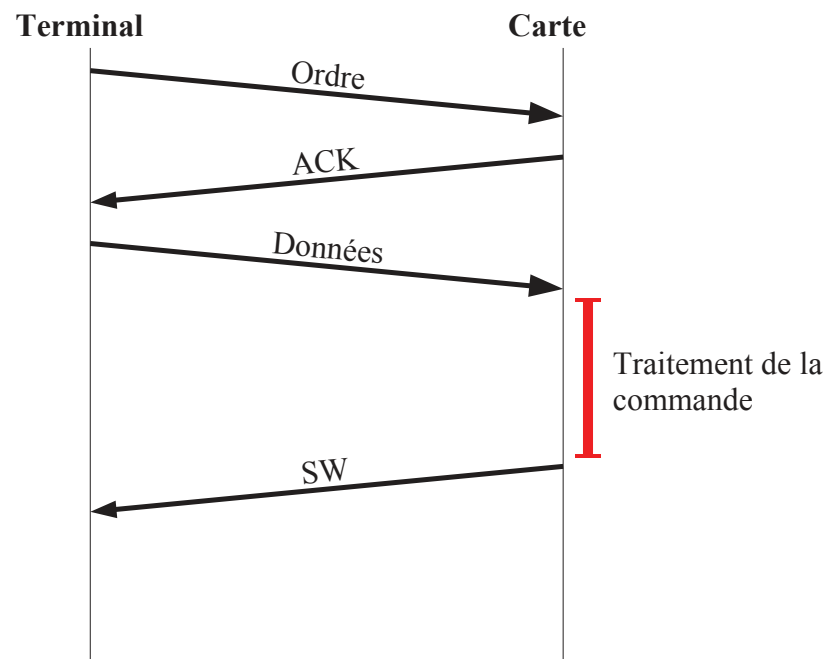
Structure de l'ordre

Octets de procédure



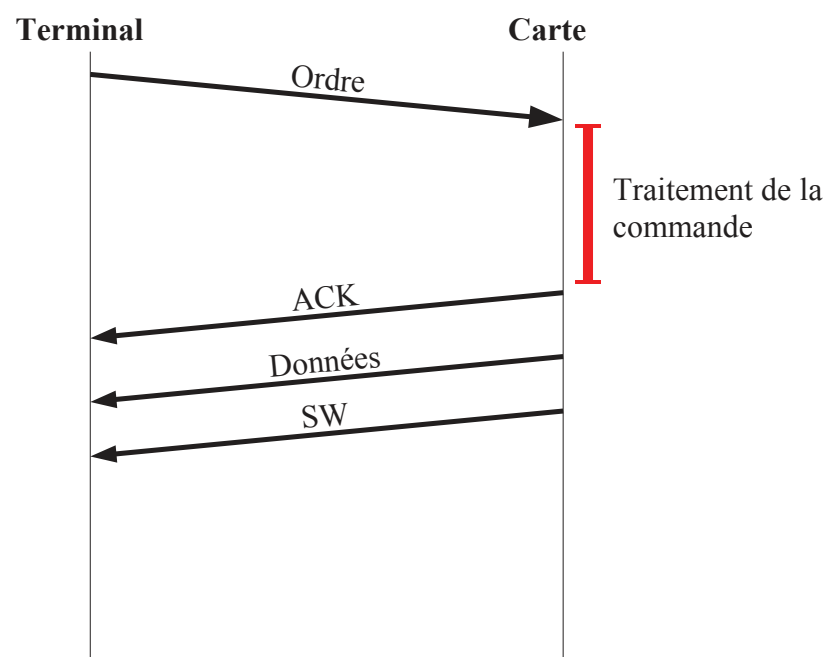
Octet	Valeur	Résultat sur VPP	Résultat en transfert de données	Puis réception de
NUL	'60'	Pas d'action	Pas d'action	Un octet de procédure
ACK	INS	État de pause	Tous les autres octets de données	Un octet de procédure
	INS @ '01'	État d'écriture	Tous les autres octets de données	Un octet de procédure
	INS @ 'FF'	État de pause	L'octet de données suivant	Un octet de procédure
	INS @ 'FE'	État d'écriture	L'octet de données suivant	Un octet de procédure
SW1	'6X' (≠'60'), '9X'	État de pause	Pas d'action	Un octet SW2

T=0 : Commande entrante



73

T=0 : Commande sortante



74

ISO 7816-4

ISO 7816-4 vise à assurer une interopérabilité.

But : indépendance des applications par rapport aux couches physique et liaison

Il spécifie :

le contenu des messages entre la carte et le lecteur

- les commandes
- les réponses

les structures des fichiers et de données :

- l'accès à ces données
- l'architecture de sécurité
- la sécurisation des communications

Le protocole APDU

Protocole de niveau application.

- La commande APDU (C-APDU) : émise par le CAD vers la carte

Entête obligatoire				Corps optionnel		
CLA	INS	P1	P2	Lc	Champs de Données (Lc octets)	Le

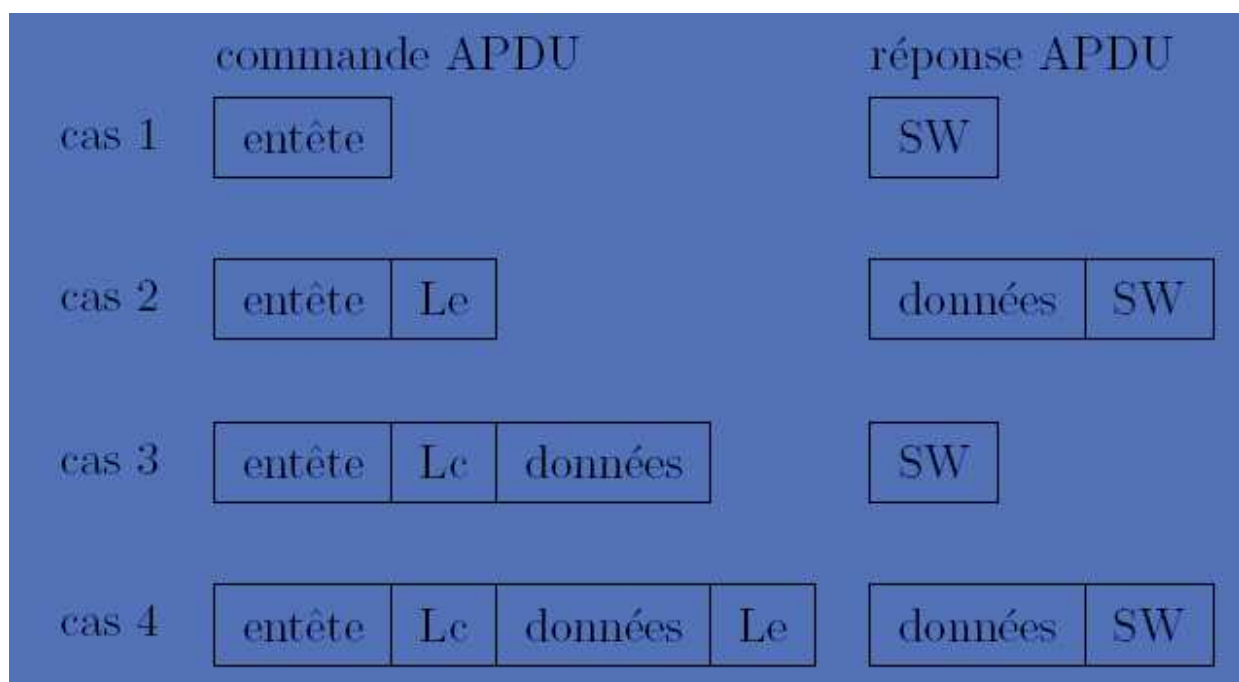
- La réponse APDU (R-APDU) : transite de la carte vers le CAD

Corps optionnel	État obligatoire	
Champs de Données (L octets <= Le)	SW1	SW2

75

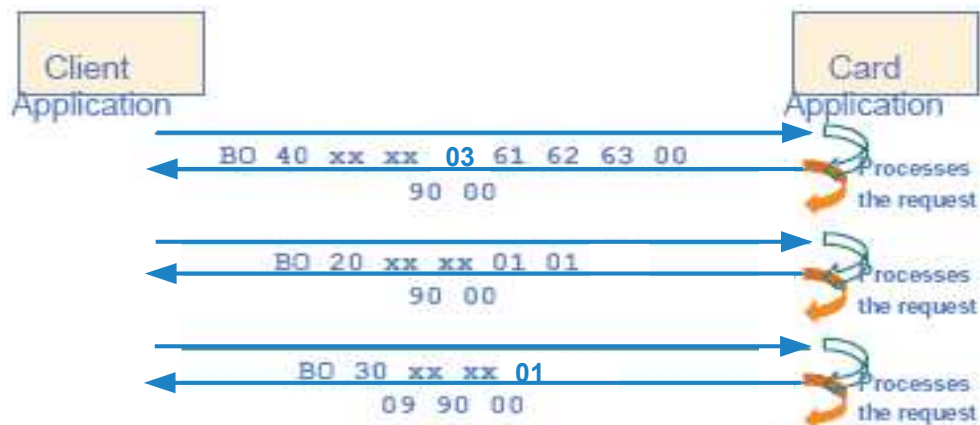
ISO 7816-4

Les différents cas d'échanges APDU :



76

Exemple de communication APDU



La transmission d'APDU en T=0 ou T=1 est détaillée dans les annexes de l'ISO7816-4

77

Complexité

Afin de pouvoir rendre transparent la communication APDU vis à vis des différents protocoles sous-jacents, on verra en Java Card qu'il sera nécessaire d'appeler les méthodes de communication dans un certain ordre.

En effet, **T=0 par exemple** ne permet pas d'avoir des commandes entrantes et sortantes ...

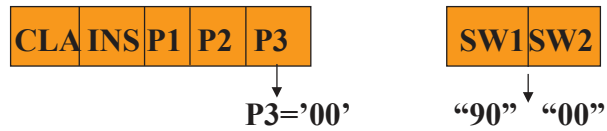
78

APDU Command-Response

Different configurations in T=0

- Command without data, response without data

Case 1



- Command without data, response with data length known

Case 2

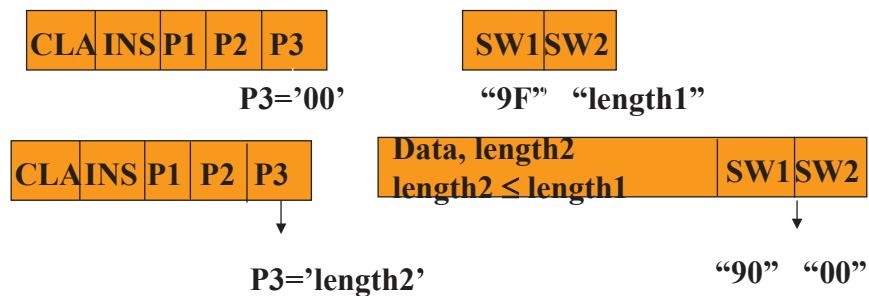


79

APDU Command-Response

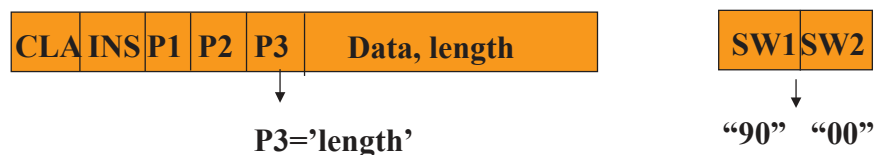
- Command without data, response with data, length unknown

Case 2b



- Command with data, response without data

Case 3

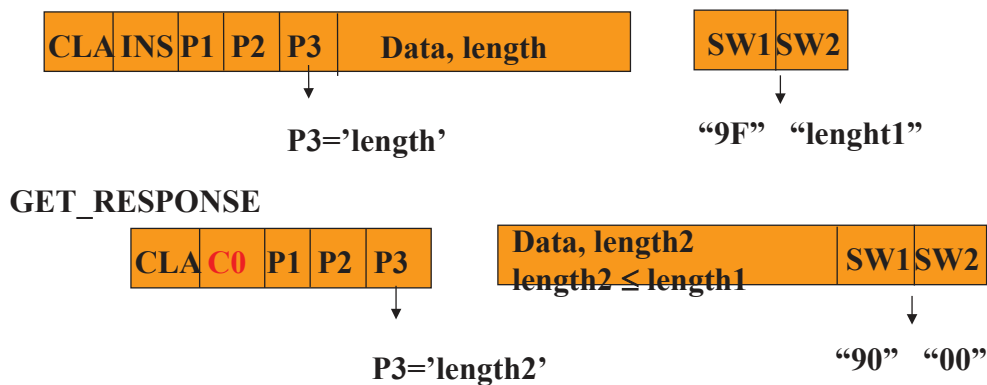


80

APDU Command-Response

- Command with data, response with data length known or unknown

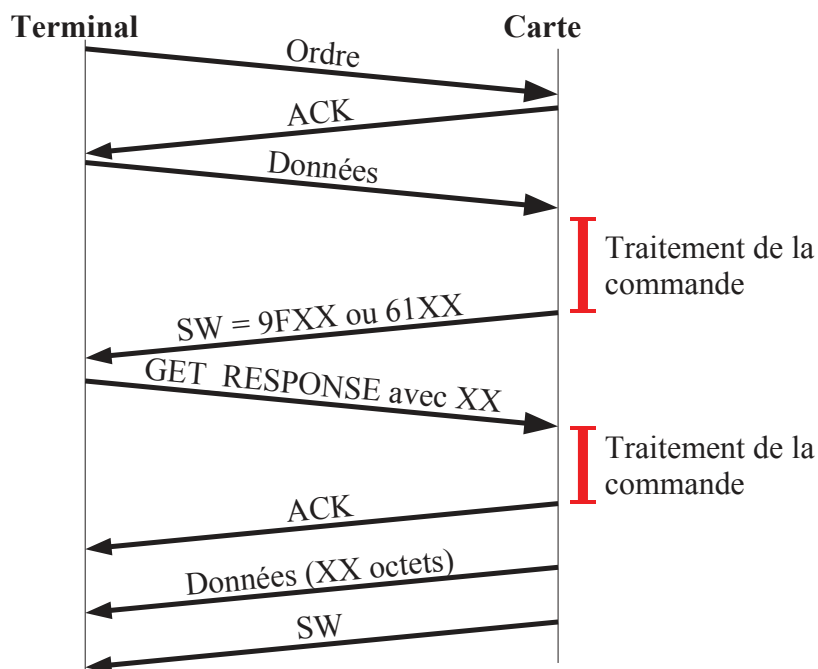
Case 4



81

Comment est traité le cas 4 APDU (données entrantes et sortantes) ?

Utilisation de la commande spéciale GET_RESPONSE



82

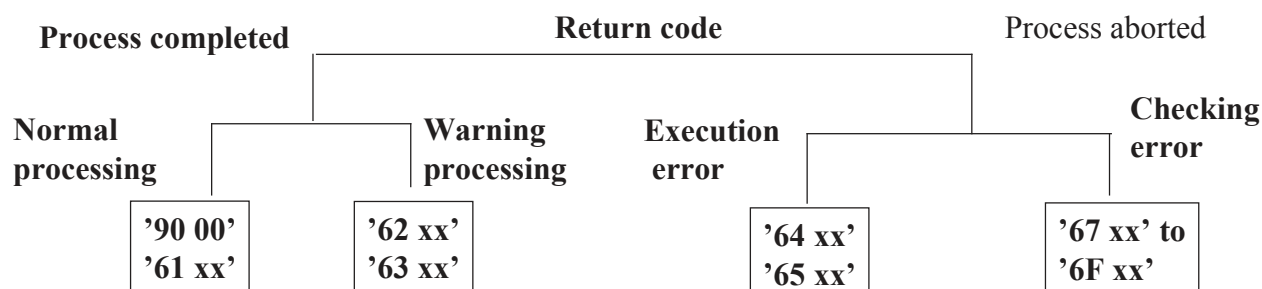
Exemples de commandes APDU

Champ de la commande APDU	Valeurs
CLA	BC = cartes de crédit françaises, cartes vitales françaises, A0 = cartes SIM (téléphonie) 00 = cartes Monéo (porte-monnaie en France), Mastercard, Visa
INS	20 = vérification du PIN, B0 = Lecture B2 = Lecture de record D0 = Écriture DC = Écriture de record A4 = Sélection du répertoire (directory) C0 = Demander une réponse (get response)
P1, P2	paramètres contenant des adresses à lire
LEN	longueur prévue pour la réponse ou bien longueur de l'argument de l'instruction
ARG	contient LEN octets (octets à écrire, PIN à vérifier, etc.)

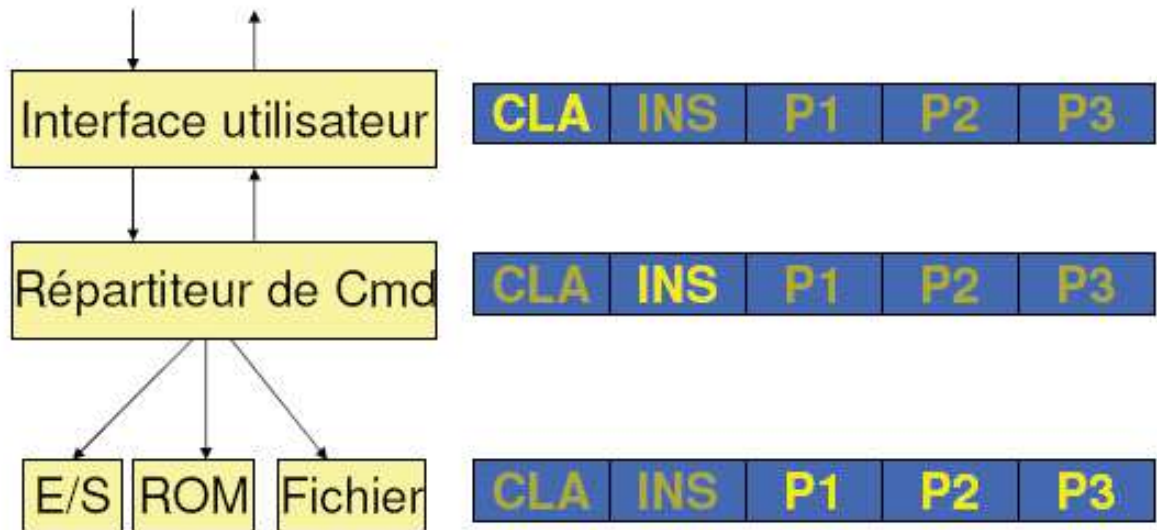
83

Return Codes

SW1, SW2 = '90 00' command successful, '63xx' or '65xx' means EEprom has been modified,
More than 50 different return codes defined by standard,
Often not respected...

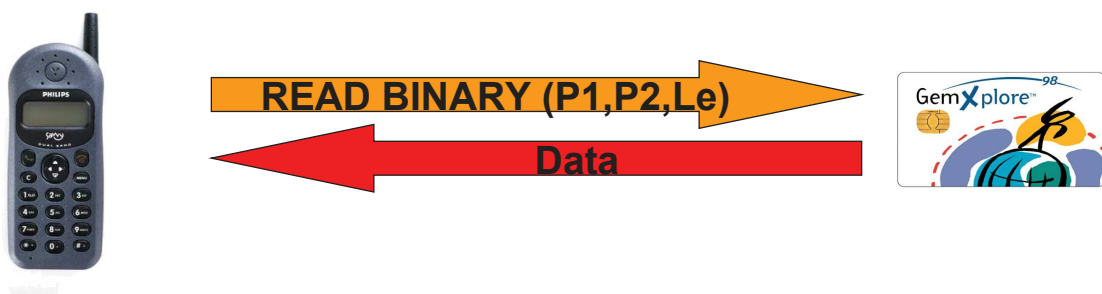


Un dispatcheur



85

Example



P1=Offset High,
P2=Offset low.

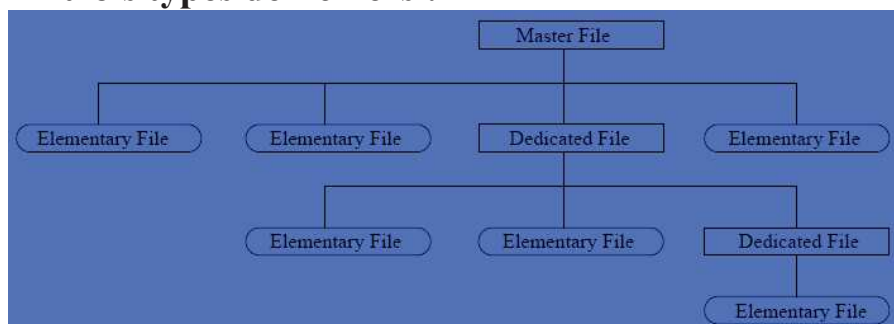
Syntax :	CLA	INS	P1	P2	Le	P1, P2 : specify the data to be retrieved Le : length of data to retrieve
	A0	B0	xx	yy	Le	

86

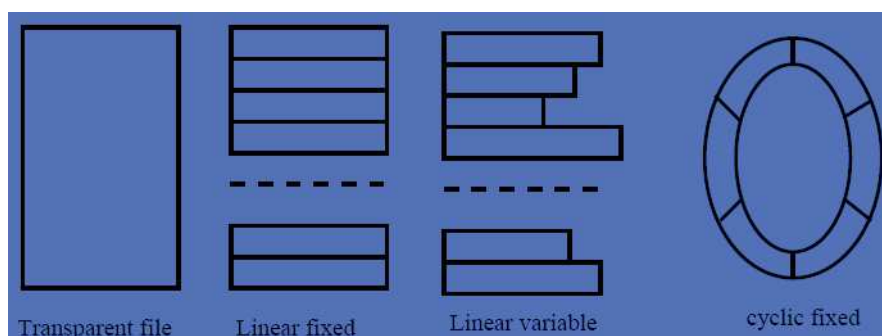
ISO 7816-4

Le système de fichiers des cartes à puce supporte un système de fichiers hiérarchique qui peut contenir trois types de fichiers :

- "Master File" (MF)
- "Dedicated File" (DF)
- "Elementary File" (EF)



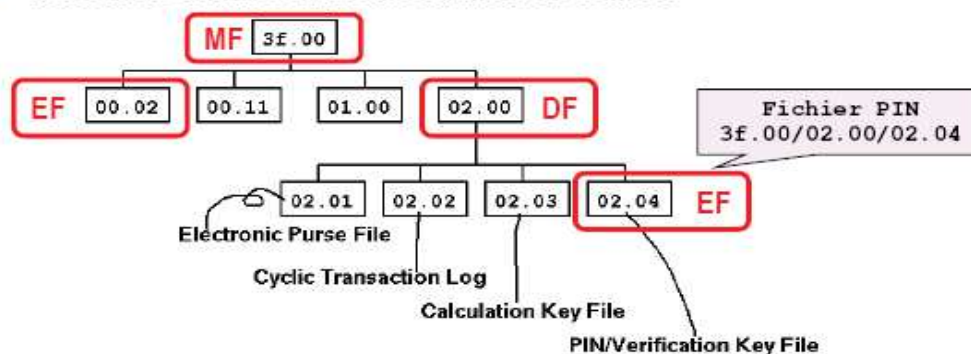
- Les différents types d' « Elementary File » (EF)



87

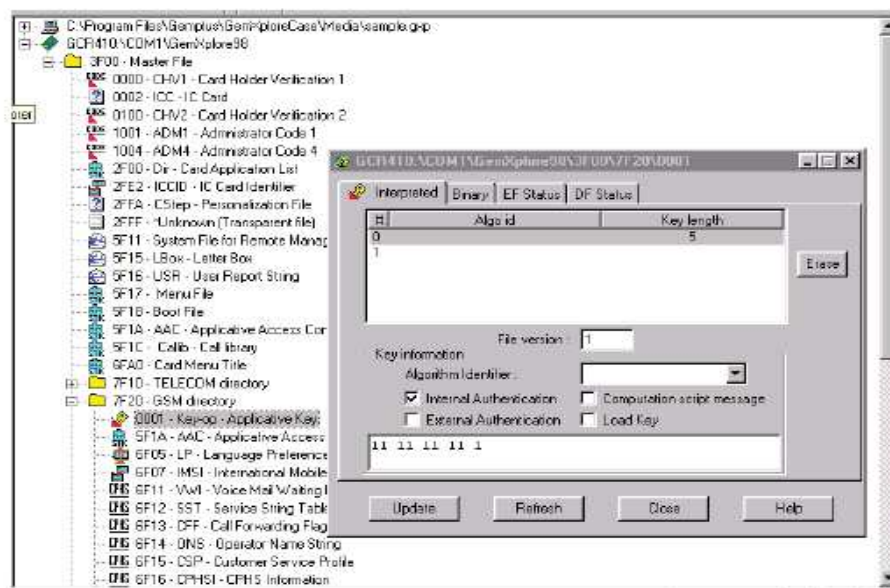
Système de Fichier ISO 7816-4

- Structure hiérarchique (MS-Dos, Unix)
 - MF : Fichier Maître : Obligatoire (Racine) 3F.00
 - Contient un fichier de contrôle d'information et de mémoire allouée
 - DF : Fichier Dédié :
 - contient des mots de passe pour accéder aux EF
 - EF Fichier Élémentaire :
 - contient des données ou des informations de contrôle



88

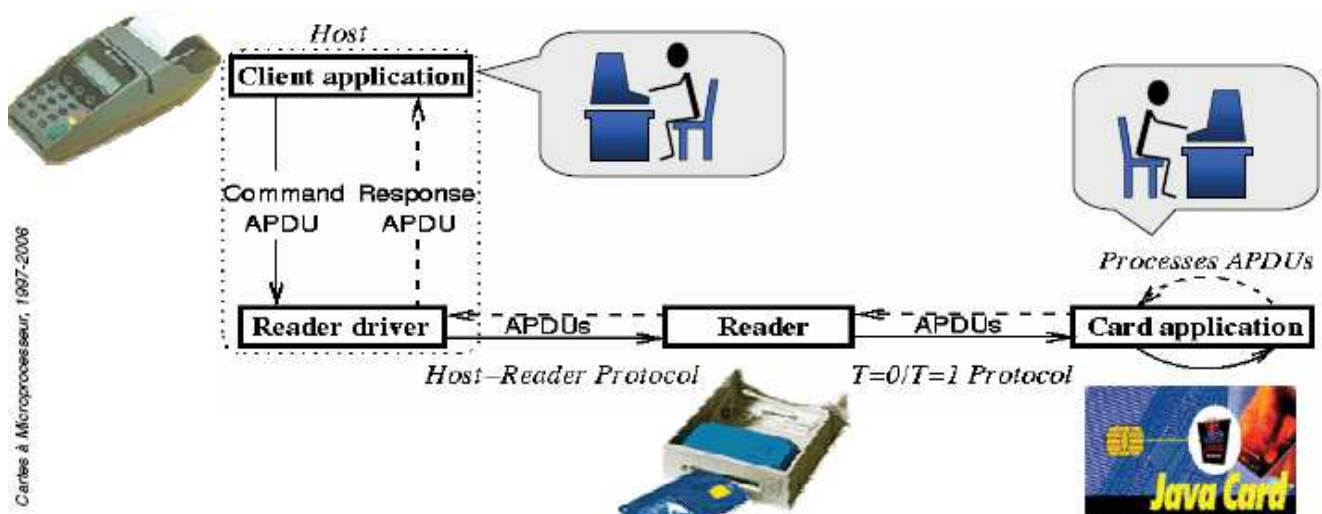
Exemple de système de fichier d'un module SIM GSM



89

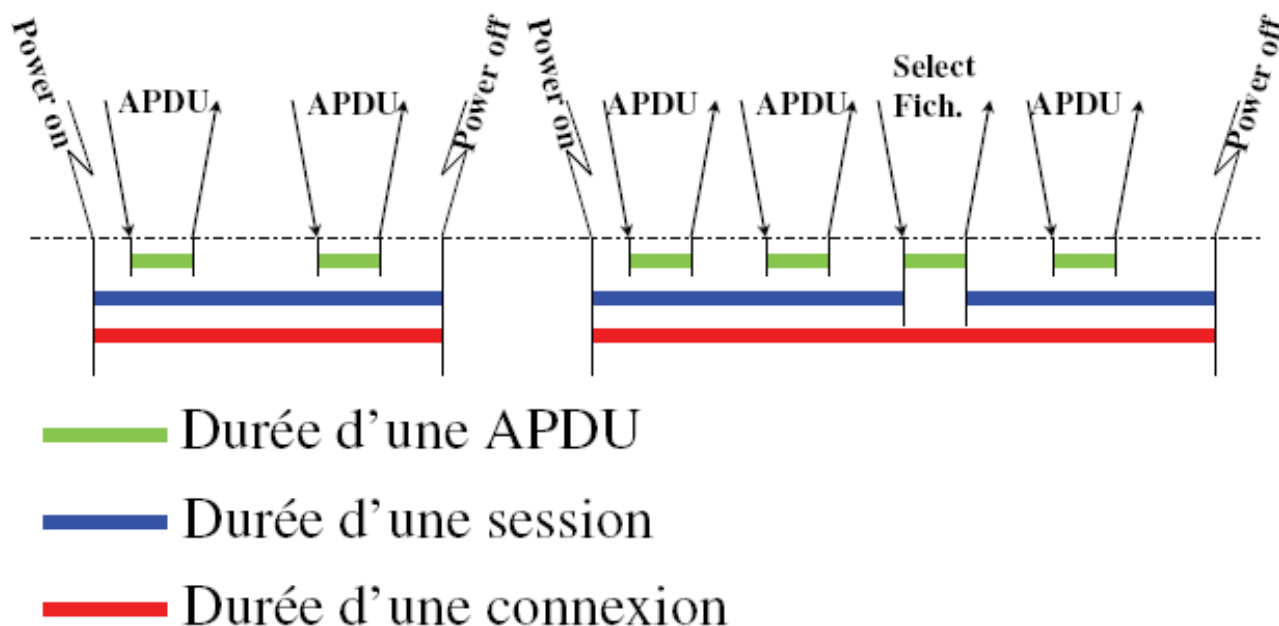
Architecture d'application carte

- Schéma général
 - Le terminal contrôle, la carte est passive
 - Dialogue terminal-carte de type requête/réponse
 - Format de messages standard : APDUs (ISO 7816-3 & 4)



90

Déroulement d'une session du point de vue de l'application hôte



91

ISO 7816-5

Spécifie le système de numérotation et les procédures d'enregistrement des identifiants d'applications (AID pour Application Identifier).

Un AID = identification unique d'une application de la carte et de certains types de fichiers

Un AID se compose de deux parties :

Ressource Identifier

RID (5 octets)

PIX (0-11 octets)

- RID is assigned by a national or international authority

RID			Meaning
D1	D2-D4	D5-D10	
X	-	-	Registration category (A international, D national)
	X		Country code according ISO 3166
		X	Application vendor number assigned by national or international body

Proprietary Identifier eXtension

- PIX is used to identify applications

En Java Card : le RID doit être le même pour le paquetage et l'applet.

92

ISO 7816-6

Spécifie les éléments de données inter-industrie tels que le nom du porteur de carte, de la date d'expiration, etc.

Exemple :

Étiquette	Longueur	Valeur				
Name	Description	Source	Format	Template	Tag	Length
Application Expiration Date	Date after which application expires	ICC	n 6 YYMMDD	'70' or '77'	'5F24'	3
Application File Locator (AFL)	Indicates the location (SFI, range of records) of the AEFs related to a given application	ICC	var.	'77' or '80'	'94'	var. up to 252
Application Dedicated File (ADF) Name	Identifies the application as described in ISO/IEC 7816-5	ICC	b	'61'	'4F'	5-16
Application Identifier (AID) – terminal	Identifies the application as described in ISO/IEC 7816-5	Terminal	b	—	'9F06'	5-16
Application Interchange Profile	Indicates the capabilities of the card to support specific functions in the application	ICC	b	'77' or '80'	'82'	2
Application Label	Mnemonic associated with the AID according to ISO/IEC 7816-5	ICC	ans with the special character limited to space	'61' or 'A5'	'50'	1-16
Application Preferred Name	Preferred mnemonic associated with the AID	ICC	ans (see section 4.3)	'61' or 'A5'	'9F12'	1-16
Application Primary Account Number (PAN)	Valid cardholder account number	ICC	cn var. up to 19	'70' or '77'	'5A'	var. up to 10

(from « EMV Integrated Circuit Card Specifications for Payment Systems », Book 3 Application Specification, Version 4.3)

ISO 7816-7

Propose des commandes inter-industrie pour SCQL.

Create table

Create view

Select

Read

Write

Update

Nom de la table: Etudiants			
Nom	Age	faculté	Année
Pierre	19	Sciences	1 ère
Marie	23	Medecine	5 ème
Alain	21	Droit	3 ème
Henri	21	Histoire	3 ème
Jean	20	Economie	2 ème
Marc	22	Droit	4 ème



ISO 7816-8

Concerne la sécurité de l'architecture et des commandes inter-industrie.

95



ISO 7816-9, ISO 7816-10, ...

A vous de chercher !

96

Des normes ... encore des normes !

Norme Europ Telecom Standard Institut:

GSM 11.11 : définition de la carte SIM

TE 9 : spécification de cartes et terminaux

EMV, (cartes de paiement)

ICAO, (agence de l'ONU, biométrie, passeport)

Santé (ISO 21 549)

...

97

Pourquoi utiliser la carte à puce ?

Avantages :

- la sécurité
- la portabilité
- la facilité d'utilisation
- la personnalisation

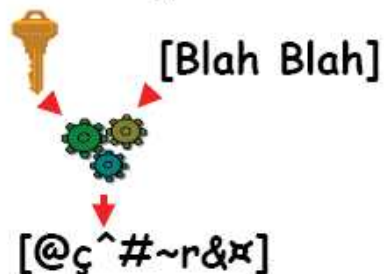
- Tamper resistance
 - Storage



- Portability



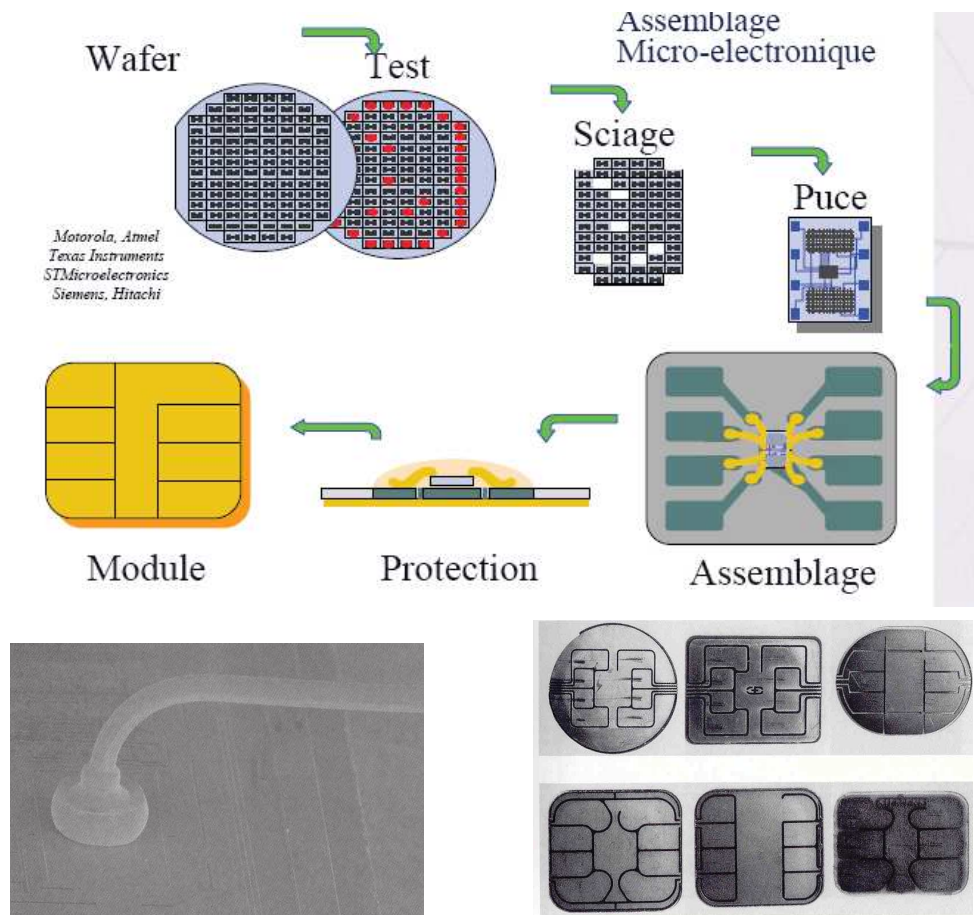
- Tamper resistance
 - Processing



- Ease of use
- Onboard key generation
- Cost

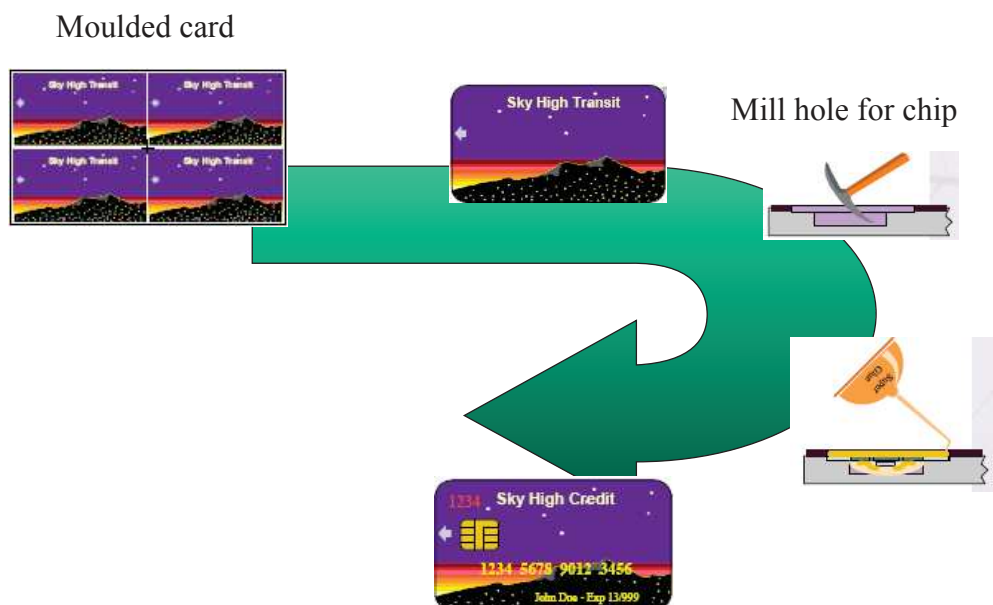
98

Cycle de fabrication



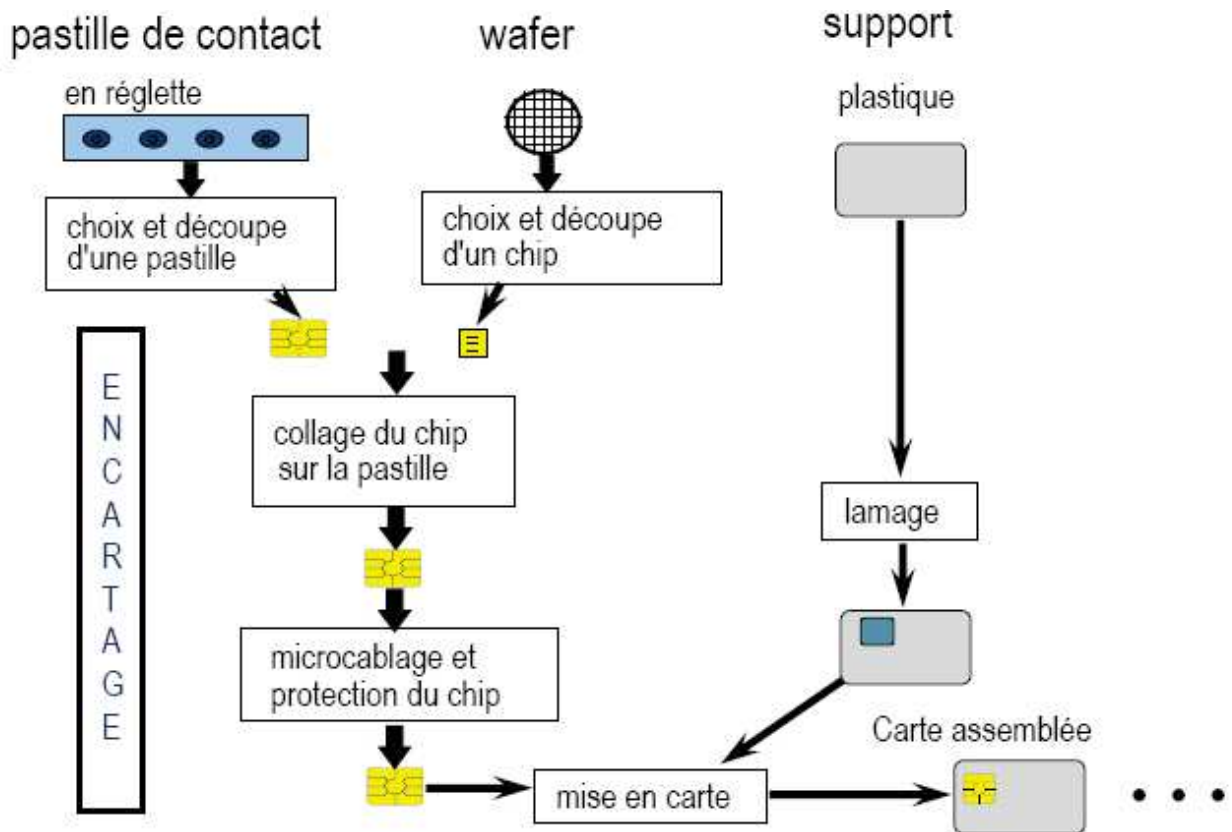
99

Cycle de fabrication

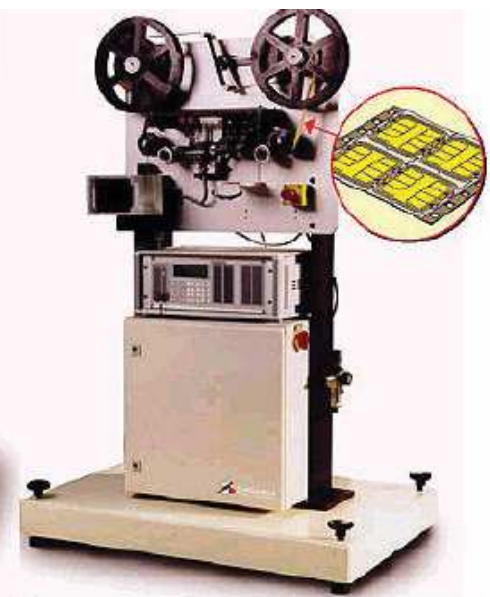


100

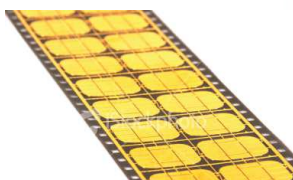
Cycle de fabrication

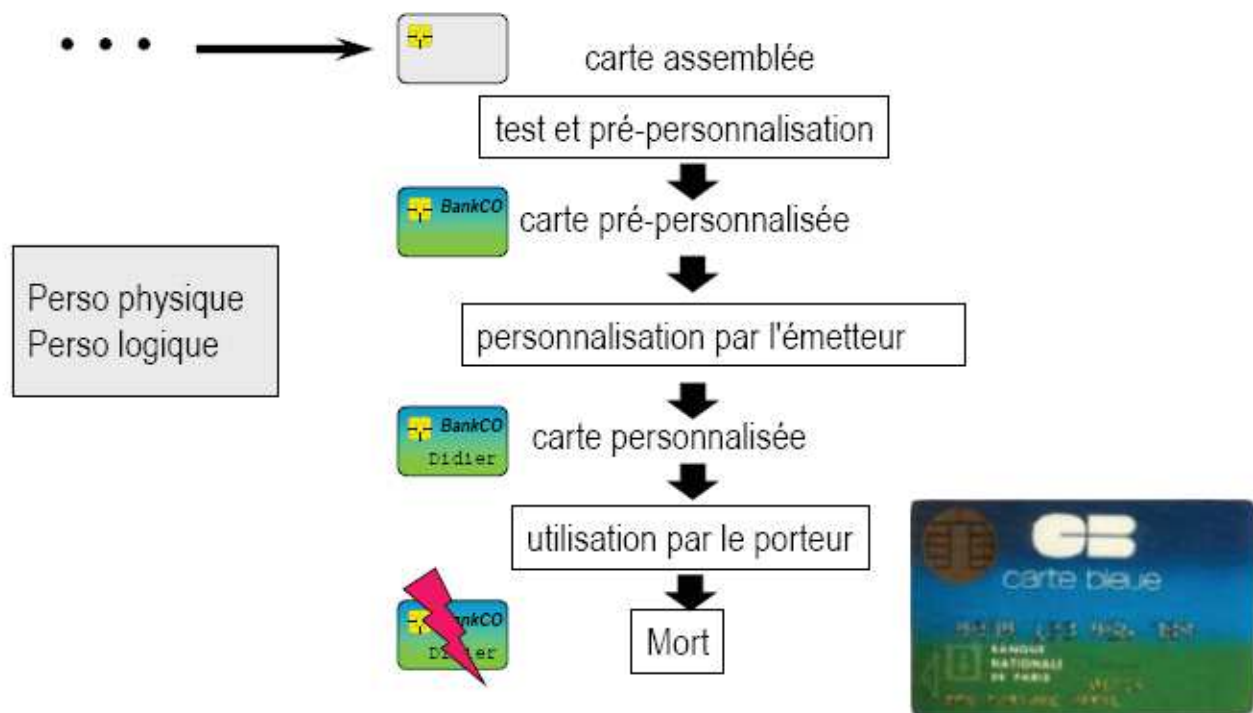


Machine pour la fabrication



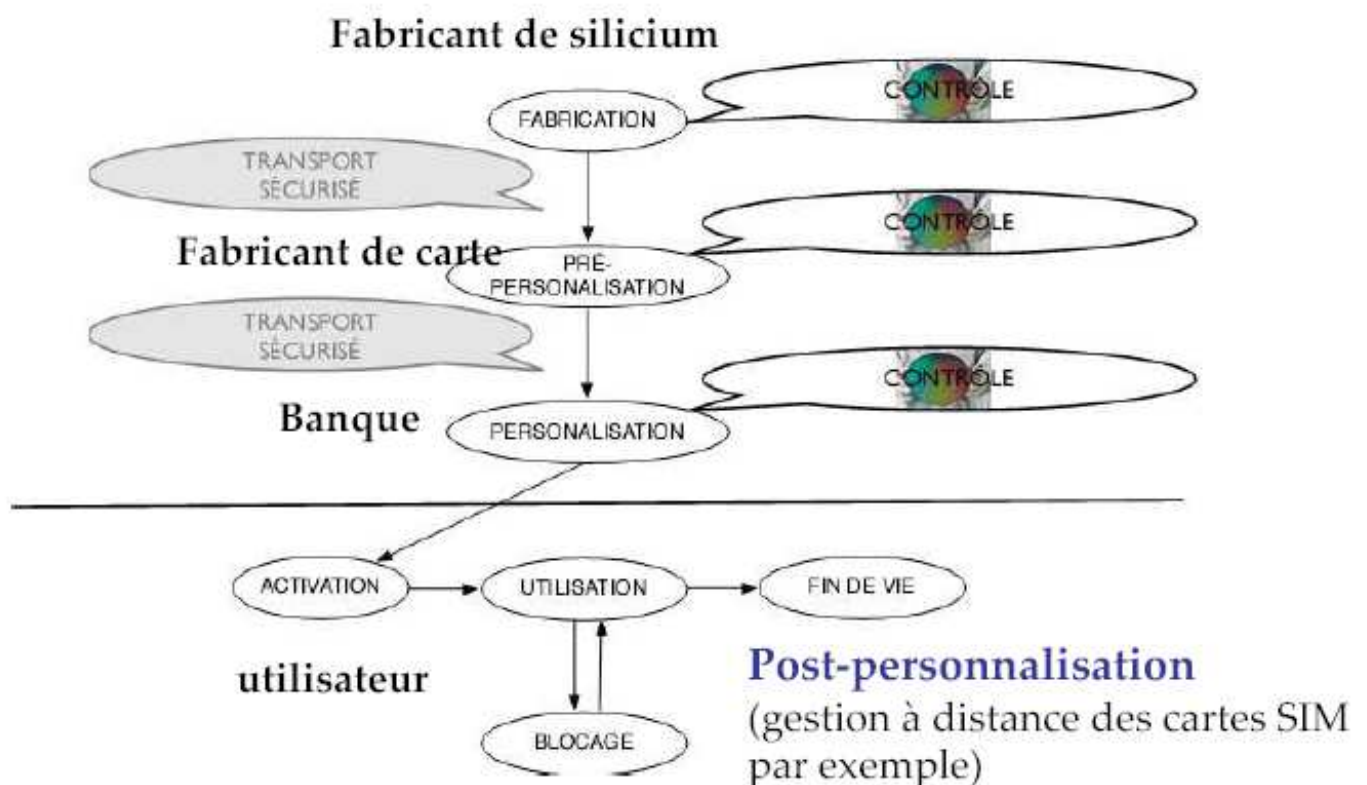
Cartes à Microprocesseur, 1987-2008





103

Cycle de vie (exemple de la carte bancaire)



Phases de Dev. D 'une appli carte

- **Choix de la carte :**
 - Puissance du processeur
 - Types et tailles des mémoires
 - Respect (ou non) des normes
- **Choix du masque**
 - Caractéristiques des « ROM » de la carte
 - OS + fonctions applicatives
 - Ecriture du masque
 - directement en assembleur
 - Dev. D 'une application carte spécifique
 - Utilisation d 'un masque « générique » + filtres
- **Initialisation+personnalisation de la carte**
 - Chargement des fonctions spécifiques de l 'application, de l 'émetteur, des utilisateurs et du porteur
- **Environnement de l 'application**

25

Systèmes d 'exploitation carte

- **Masque dédié : une seule application figée**
 - BO (CB, Sesame Vitale, CPS)
 - Monéo
- **Mono Application**
 - ISO 7816-4
 - Système de fichier sur carte
 - Authentification du porteur et contrôle d'accès
 - ISO 7816-7
 - Moteur de base de données
 - *BasicCard*
- **Multi Applications**
 - Multos
 - JavaCard
 - .NET SmartCard

106

Applications

L'industrie des télécommunications

L'industrie bancaire et monétaire (B0', EMV)

Le porte-monnaie électronique (Monéo, Proton, CEPS)

Le secteur de la santé

L'industrie audiovisuelle avec la télévision à péage, ...

Les transports en commun.

Le contrôle d'accès physique de personnes à des locaux, ...

L'identification : à des sites sur l'Internet, ...

Les "e-services"

L'identification gouvernementale (carte d'identité, ...)

Les applications de fidélité



La carte à puce aujourd'hui

➤ Aujourd'hui : près de 7 milliards de cartes en circulation

➤ **Monétique** :

- Carte bancaire : Groupement Cartes Bancaires, nouvelles cartes EMV, etc.
- Porte-monnaie : **Octopus**, **Moneo** en France, **Proton** en Belgique, **Geldkarte** en Allemagne

➤ **Identification** :

Cartes d'identité nationales (**eID** en Belgique), **E-passeports** (août 2006 en France),
Passeport biométrique (depuis la fin Juin 2009)

➤ **Éducation** (comme carte d'étudiant et/ou de restauration)

➤ **Téléphonie mobile** (carte **SIM**)

➤ **Secteur médical** (carte **Vitale** en France, carte **SIS** en Belgique).

➤ **Titre de transport** (**Passe Navigo** à Paris, **Oyster** à Londres).

➤ **Sécurité informatique** (authentification forte et signature électronique): carte dotée d'un cryptoprocésseur pour la génération des clés et le stockage de la clé privée).

Exemples des Passeports

Depuis 2006 en France :

Passeport électronique comporte une puce électronique qui stocke les données personnelles du détenteur : (son nom, sa date de naissance, sa nationalité, son numéro de passeport et la photo numérisée du titulaire).

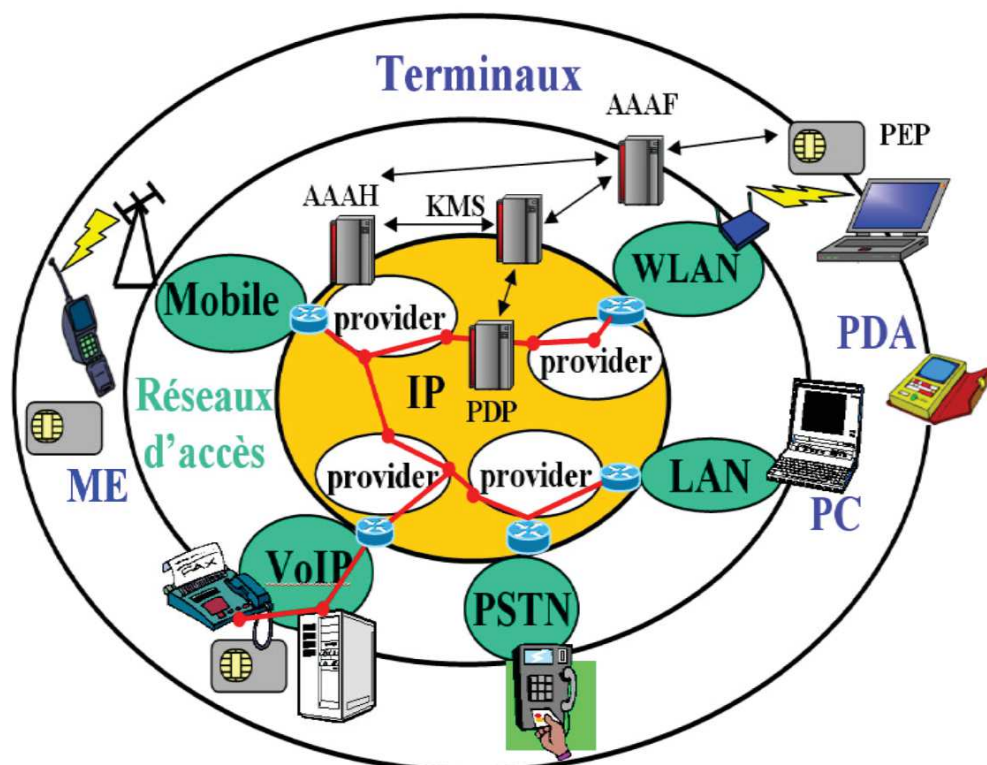
Depuis le 15 Juin 2009 :

Passeport biométrique : sur une puce RFID, qui permet de lire les informations à courte distance, sont enregistrés - outre les informations personnelles classiques et la photo numérisée - deux empreintes digitalisées des doigts du détenteur (à partir de l'âge de 6 ans).

(d'après : http://www.prefecture-police-paris.interieur.gouv.fr/demarches/passeport_elec/passeport_2006.htm)

109

La carte : omniprésente !

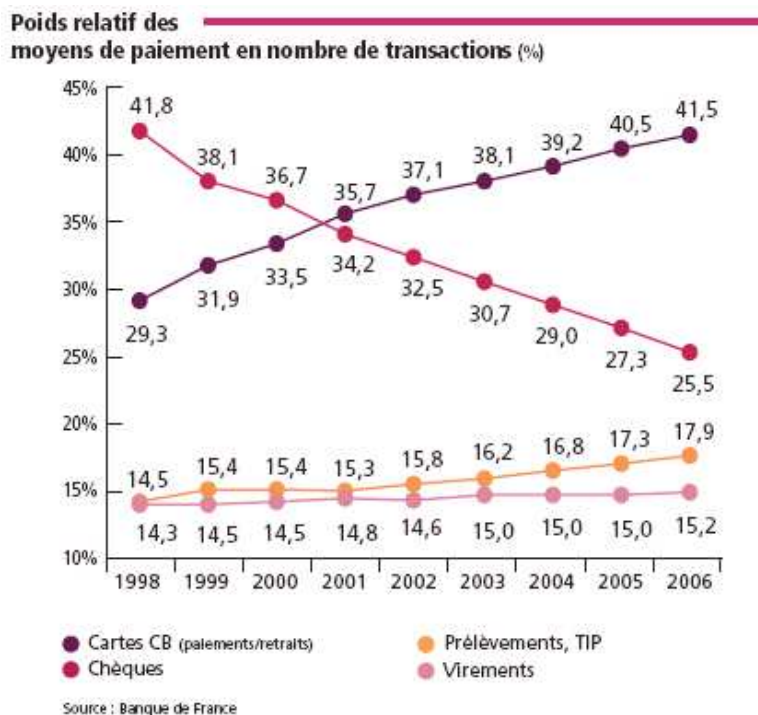


110

Applications de paiement

Cartes de Crédit : Somme débitée sur le compte du titulaire avec un taux d'intérêt fixé

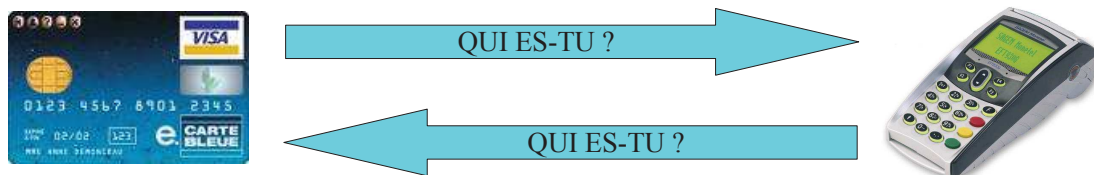
Cartes de Débit : Compte débité quelques jours après l'achat



111

Applications de paiement : Techniques

Authentification mutuelle carte et terminal commerçant



Signature Électronique



Contrôle du PIN



Contrôle (régulier ou systématique) du solde bancaire (en ligne)

112

Applications de paiement (Exemple : GIE Cartes Bancaires)

53,6 millions de cartes

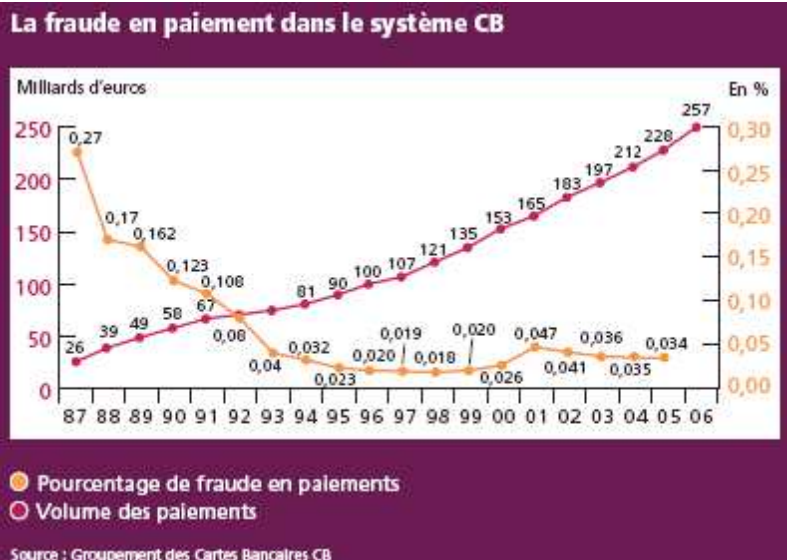
1,2 million de points d'acceptation :

- 950 000 terminaux de paiement dans les commerces
- 140 000 automates de vente
- 50 000 distributeurs de billets
- 75 000 commerçants à distance dont 15 000 sites Internet

1 Distributeur Automatique de Billets pour 1200 habitant en France

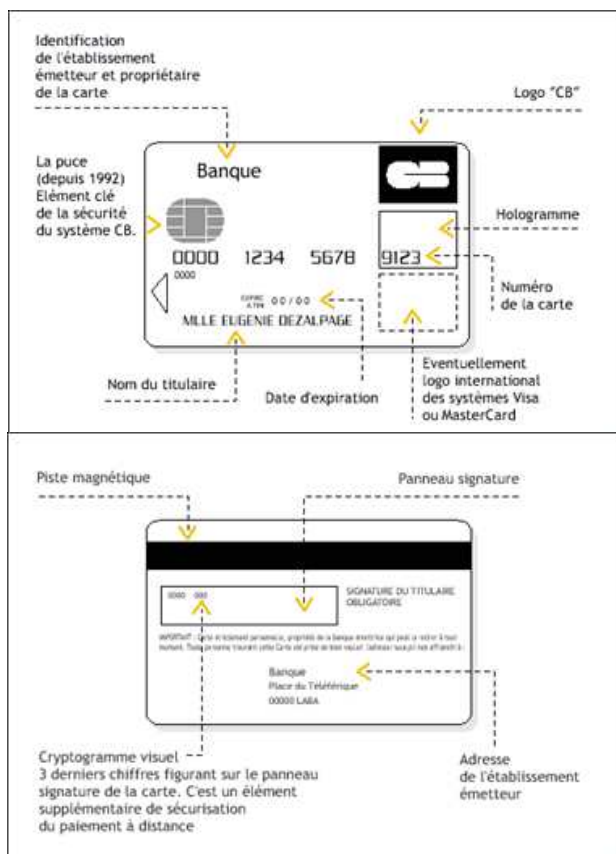
349,5 Mds€ de transactions en 2006 (257,3 Mds€ de paiements et 92,2 Mds€ de retraits)

260 M€ de fraudes du à la piste magnétique contrefaite et utilisée depuis l'étranger (en 2003)



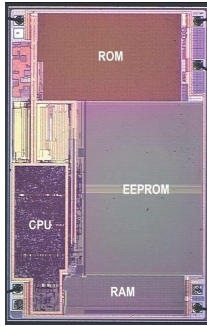
113

La carte bancaire



114

La carte à puce et sa sécurité



La Yescard : une carte « pirate » qui dit OUI à toutes les transactions !
Une remise en cause la sécurité de la carte à puce ?



NON

Il s'agit d'une mauvaise utilisation de la carte dans un système plus large, ici, le système bancaire

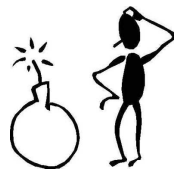
Exemple : Utilisation d'un seul des différents verrous du coffre-fort

Aujourd'hui on utilise « tous les » verrous et le système est à nouveau sécurisé.

115

Les risques

Sanitaire : à priori aucun !



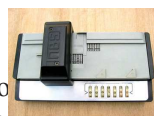
Financier :

- Sur un chèque ou sur une demande de prélèvement automatique, **la signature apposée prouve l'identité de l'émetteur**
- Dans le cas de l'utilisation d'une carte bancaire, **la preuve est électronique et fournie par le PIN !**
- Dans certains pays qui n'utilisent pas le PIN
la puce est interrogée
OU
la piste magnétique de la carte est lue
OU
une empreinte physique de la carte est faite

ET

une signature est demandée sur le reçu émis
SAUF
dans les automates où on ne demande rien !
(voilà une source de fraudes)

- À distance et ou par téléphone



Le vendeur ne peut fournir aucune preuve irrefutable que l'acheteur est bien qui il prétend être.
Par conséquent si la banque le laisse se servir sur votre compte, il en relève de sa responsabilité.
La loi oblige d'ailleurs les banques à rembourser les sommes prélevées dans le délai d'un mois.

N'ayez plus peur de commander sur l'Internet !

116

Plan de Migration de VISA vers les SC

Feb 1998	Launch of Chip Migration program
Jan 1999	New Visa and Electron chip cards to be EMV and VIS compliant
Jan 2001	All new Visa Cash programmes to be CEPS compliant
Oct 2001	All Acquirer hosts certified to EMV, All new Terminals EMV+VIS compliant (POS, EPOS, ATM)
Jan 2002	Regional review to assess EMV and PIN readiness
July 2002	Visa, Electron and Visa Cash to be EMV, VIS and CEPS compliant
Jan 2005	All Visa Electron terminals to be EMV and PIN enabled, Proposed Intra regional liability shift

Royaume Uni: PIN obligatoire depuis le 14/02/2006

117

Applications de paiement (2/3)

- Carte Prépayée
 - Cas particulier du PME : la carte contient des jetons (et non de l'argent)
 - Ex : Abonnement autoroute (Marseille), Places de cinéma, ...
 - Une réussite : la Télécarte

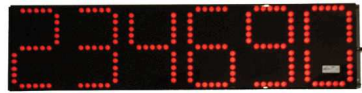
118

Analyse du succès de la télécarte

C'est une carte prépayée.
Elle contient des jetons (et non de l'argent)

Les raisons de sa réussite :

- Fin du vandalisme des cabines téléphoniques
- Temps de communication +50%
- Support publicitaire



119

Le porte-monnaie électronique (e-Purse)

Remplace les pièces et les petites coupures de billets

Carte rechargeable dans des guichets (ou chez le commerçant)

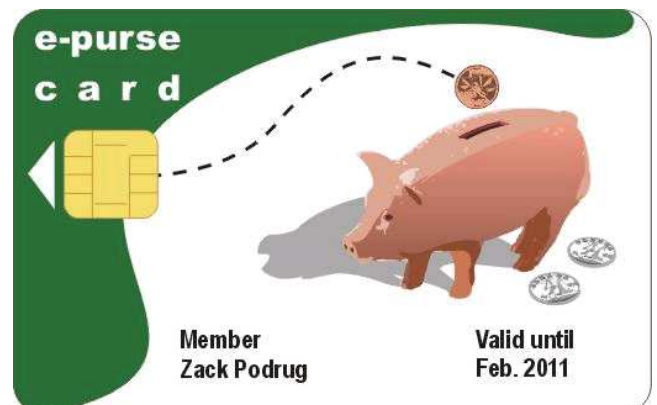
Permet de régler des petites sommes (fraction d'euros à quelques dizaines)

Mode de paiement sûr, pratique, rapide, anonyme (?)

Mais en cas de perte, l'argent est perdu

Exemple :

- SIBS (Portugal)
- Proton (Belgique)
- **Monéo** & Modeus (France)
- GeldKarte (Allemagne)
- CEPS



120

Le porte-monnaie électronique (e-Purse) : Avantages

Pour la banque :

- Absence de fraude
- Contrôle des facilités de crédit
- Coût de transaction faible
- Diminution des liquidités

Pour le commerçant :

- Garantie de paiement
- Rapidité d'encaissement
- Absence de liquidité

Pour le porteur :

- Paiement rapide
- Rechargement simple
- Protection par PIN (optionnel)
- État des différentes transactions sur demande



121

Applications de sécurisation

• Sécurité physique :

- Accès à des locaux : ouverture et fermeture de porte (suivi de passages)
 - ex : Ministère des finances, Gemplus

• Sécurité logique :

- Contrôle d'accès logique à un serveur
 - Identification, Authentification
- Protection de messagerie électronique
 - Signature électronique et chiffrement des messages
- Renforcement de la sécurité des transactions sensibles
 - Ordres de virements internationaux (chiffrement de code porteur)
- Vote électronique

Applications d'identification

- Identification d'une personne :
 - dans une entreprise (contrôle horaire),
 - sur un réseau informatique (login),
 - sur un réseau téléphonique (carte SIM),
 - ou dans la société (carte d'identité)

123

La carte SIM et le GSM

L'abonné est localisé par la carte (le terminal utilisé devient celui du porteur – personnalisation du mobile)

Facturation directe de l'abonné

Sécurité pour l'accès au réseau téléphonique : physique (carte) + logique (PIN)

Stockage de données personnelles (agenda)

1,22 milliards de cartes SIM livrés en 2005

Bientôt, le paiement ?



124

Applications Actuelles des Cartes

- Différents secteurs porteurs

- Les Télécommunications : Cartes Pré-payées, carte SIM
 - Recherche tj + de services, $\forall$ le coût
- Les Banques : Porte-monnaie Électronique, Carte Bancaire
 - Recherche de coût de fonctionnement le plus faible
- La Santé : Carte patient, carte professionnels de santé
 - Recherche la sécurité d ' informations sensibles

- Sous l 'impulsion du GSM,
la carte doit offrir de + en + de services

125

Futures Applications

- La Carte « bookmark » ou Portail

- Toutes les informations propres à l 'utilisateur sont stockées dans la carte
- configuration des serveurs mails et news,
- liste des signets
- Liste des mots de passe

- La Carte à Mémoire Etendue

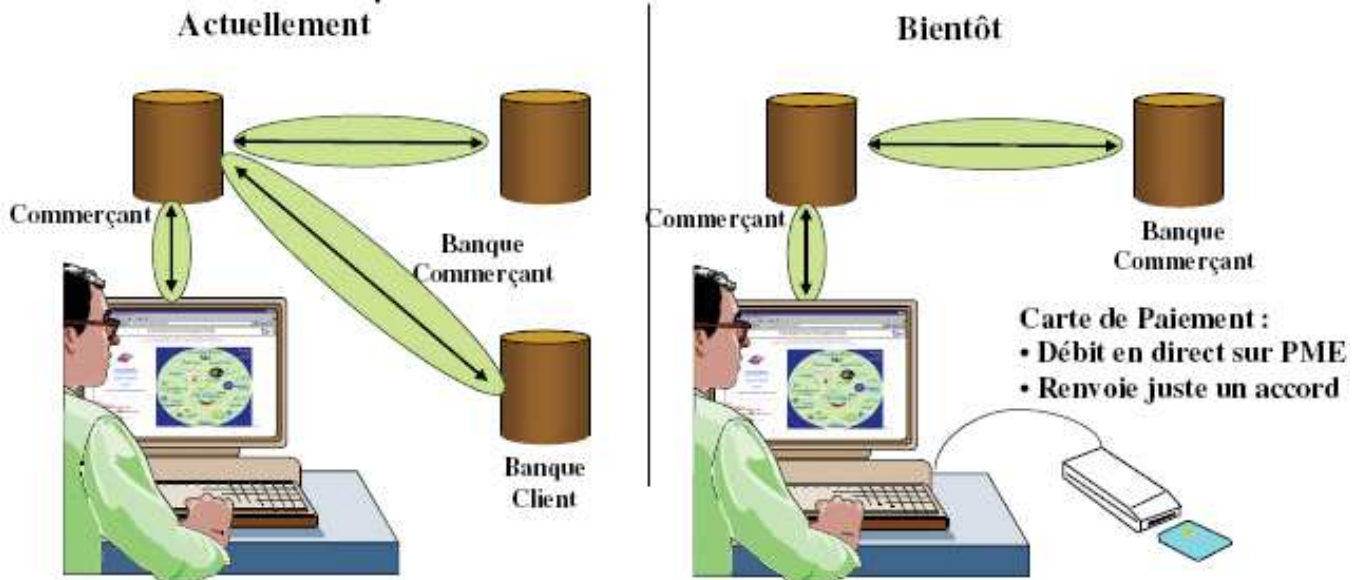
- La carte contient des liens vers des informations stockées sur des serveurs anonymes
- Permet de laisser les informations sur le réseau sans problème de sécurité
- L 'information peut être cryptée, et la carte est utilisée pour décrypter

- La Carte pour personnaliser un terminal générique

- CESURE
- Déploiement des composants de l'application terminal

Futures Applications

- La carte et les paiements distribués



127

Quelques acteurs

Fondeur: Atmel, Infineon, NXP, Samsung, ST Microelectronics,

Encateur: Gemalto, Giesecke&Devrient, Oberthur Card Systems, Orga

Bancaire: Carte Bancaire, Carte Bleue, EMV, MasterCard, Monéo, VISA

Téléphonie: Bouygues Télécom, France Télécom, Free, Orange, SFR, Vodafone

TV: Canal+, CanalSat, TPS

Santé: Sécurité sociale

Evaluation/Certification: Silicomp, Thales, CEA Leti, SERMA Technologies, SGDN -- ANSSI

Laboratoire: LaBRI, XLIM, INRIA, CNRS, CNAM, ...

128

Les Acteurs (2006)

- Bull CP8 (racheté par Schlumberger)
- Schlumberger
- Axalto (filiale mise sur le marché par Schlumberger) (#2)
- Gemplus (#1)
- Gemalto (Mariage Gemplus+Axalto) 7 Décembre 2005
 - 1,8 G€ de CA, 11000 employés, ~50 pays, 22 usines
 - 55% du marché mondial des cartes SIM avec plus de 600 millions de cartes livrées sur un total estimé de 1,22 milliards de SIM pour 2005. La nouvelle entité sera aussi un poids lourd sur le marché des services financiers et sur celui des lecteurs. Elle ne devrait toutefois contrôler "que" 16% du marché des cartes de sécurité
- Oberthur Card Systems
- Delarue
- Giesecke & Devrient (#3)
- Sagem Orga (#4)
 - 1,800 employés
- ...

La carte à puce

—
Sa sécurité

La sécurité physique

techniques d'impression sophistiquées

différentes couches d'impression
hologramme
Embossage

procédés d'encollage

matériaux plastiques

design du micro-contact

131

La sécurité hardware

Le micromodule est monolithique => difficile de distinguer ses différents composants

un numéro de série unique

l'utilisation de mémoire de type PROM

blindage physique du composant (grille, grille active, ...)

des détecteurs de conditions anormales (tension, fréquence, température, lumière, ...)

brouillage des informations dans le composant (mémoire, bus, ...)

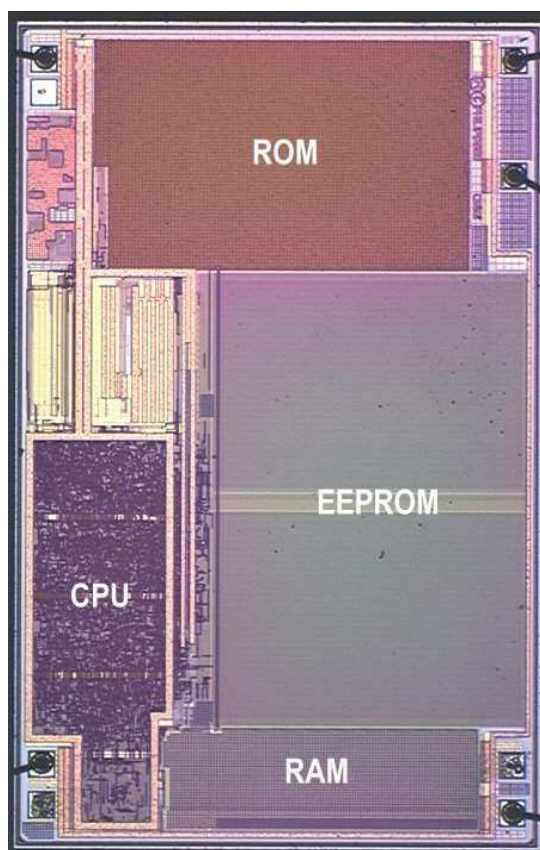
co-processeurs cryptographiques

pompe de charge pour le lissage de la consommation

Dual logic, circuit asynchrone

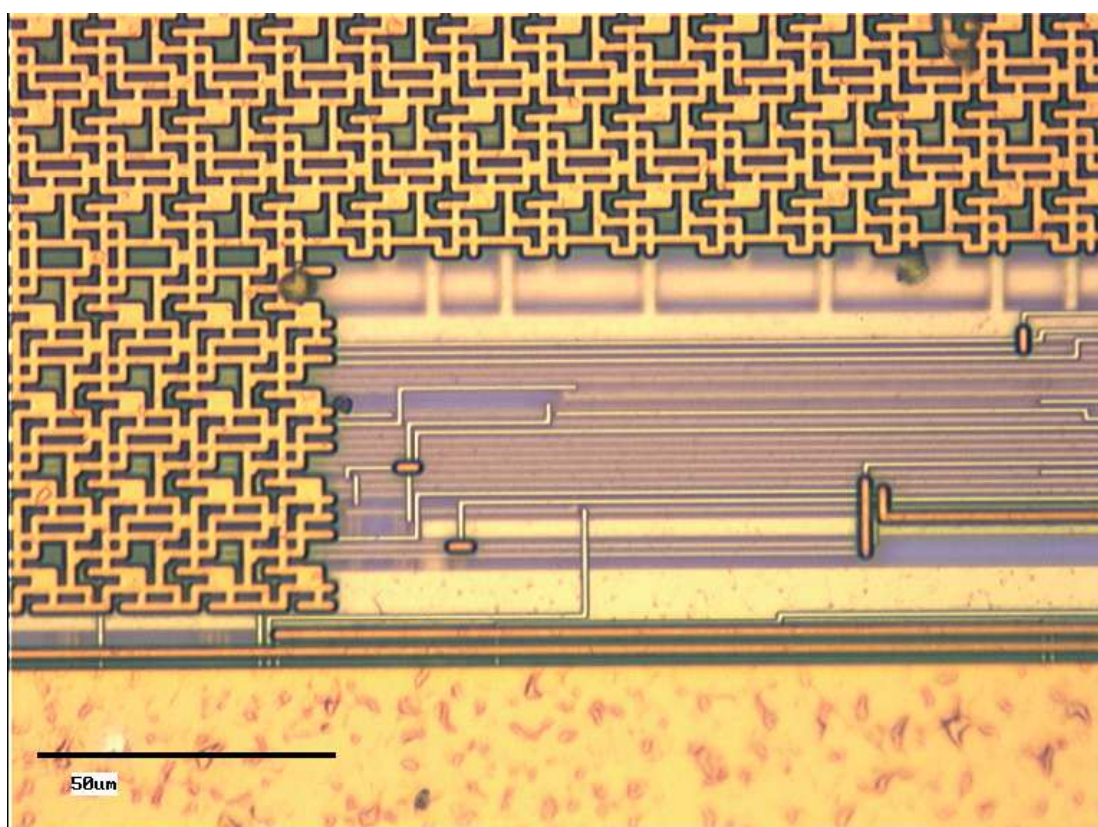
132

Une puce



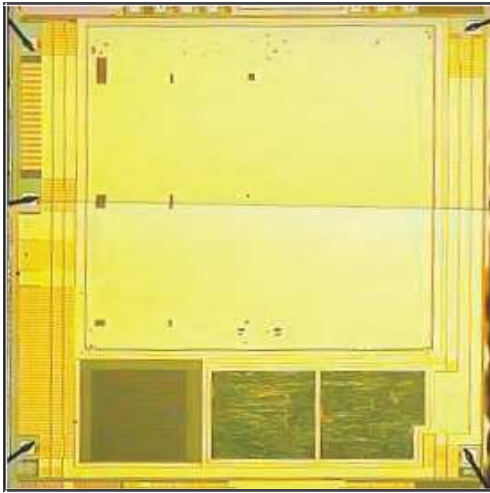
133

La grille de protection

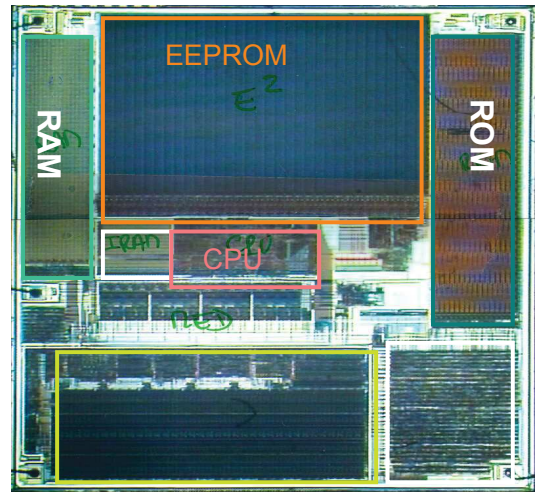


134

LESSON LEARNED: SHIELD IT!

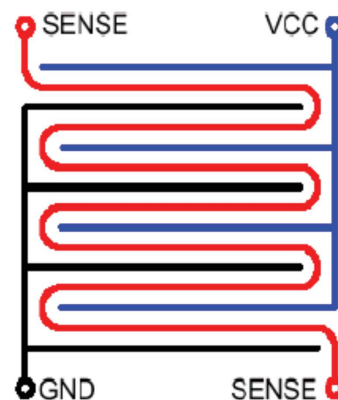
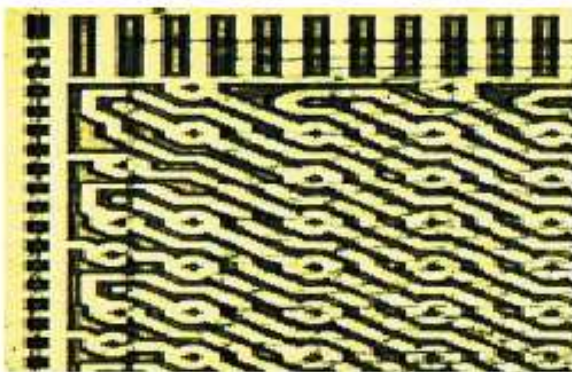


- Shield
- Scrambled glue logic
- No Buses visible

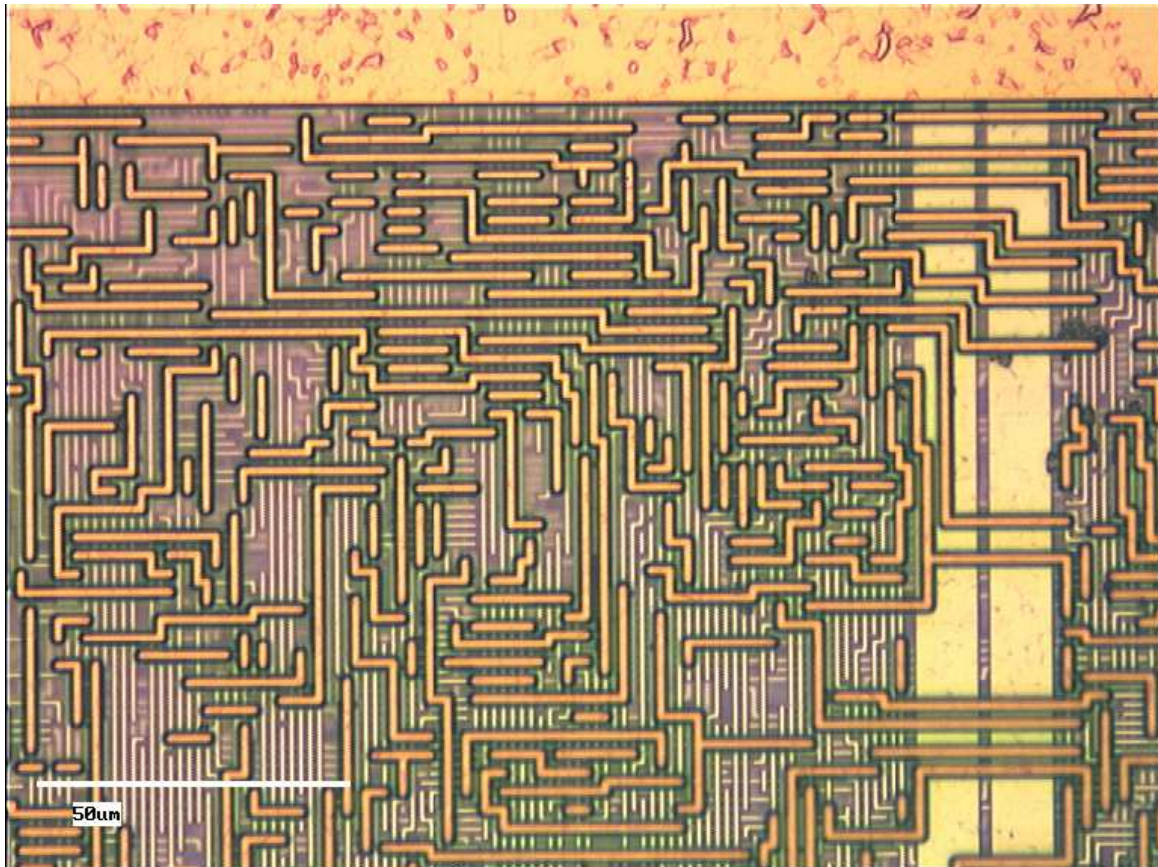


- Blocks easily identified
- No shield
- No glue logic
- Buses clearly visible

Grille active

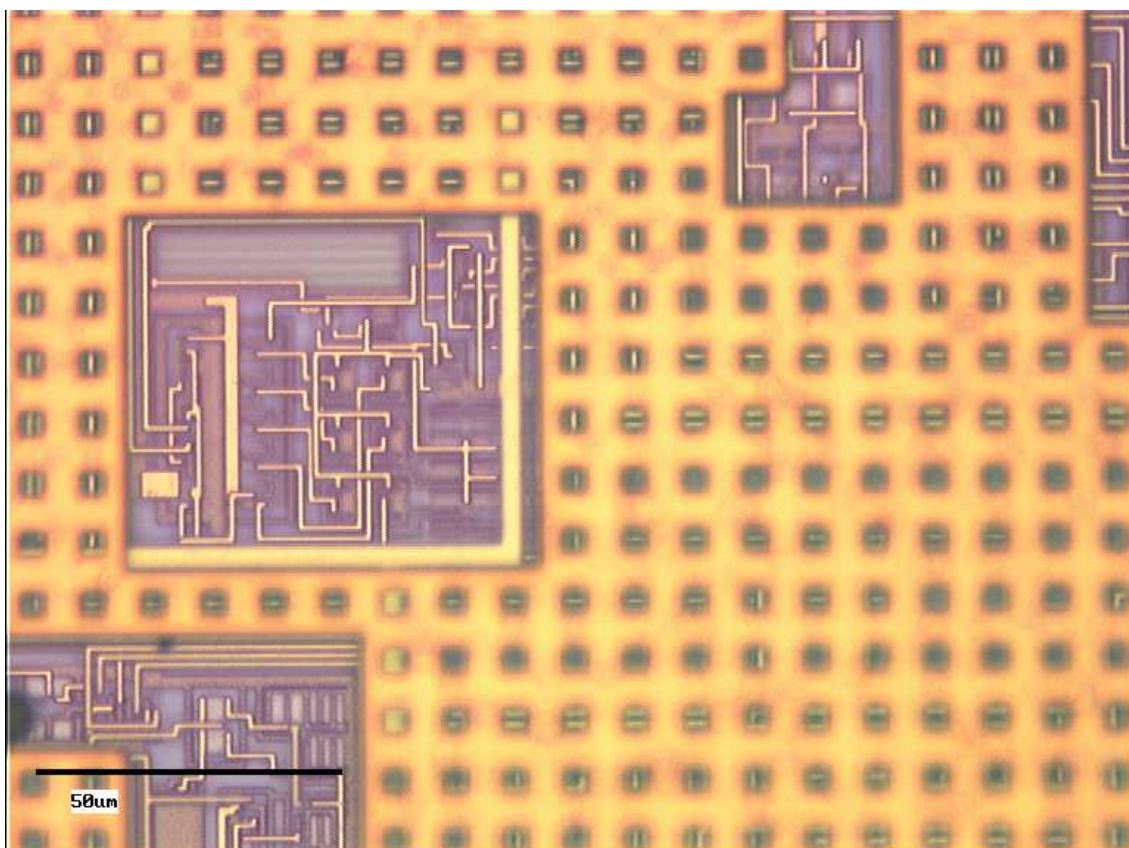


Le CPU



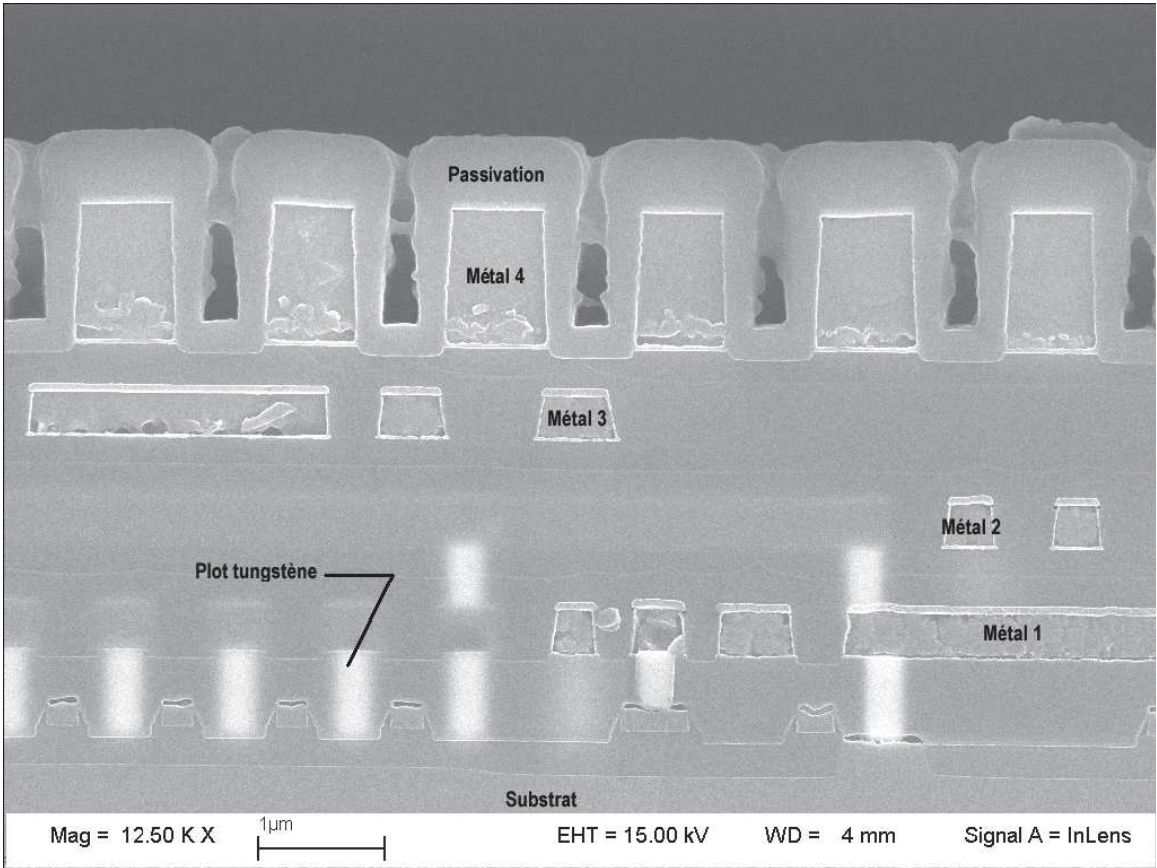
137

La partie analogique



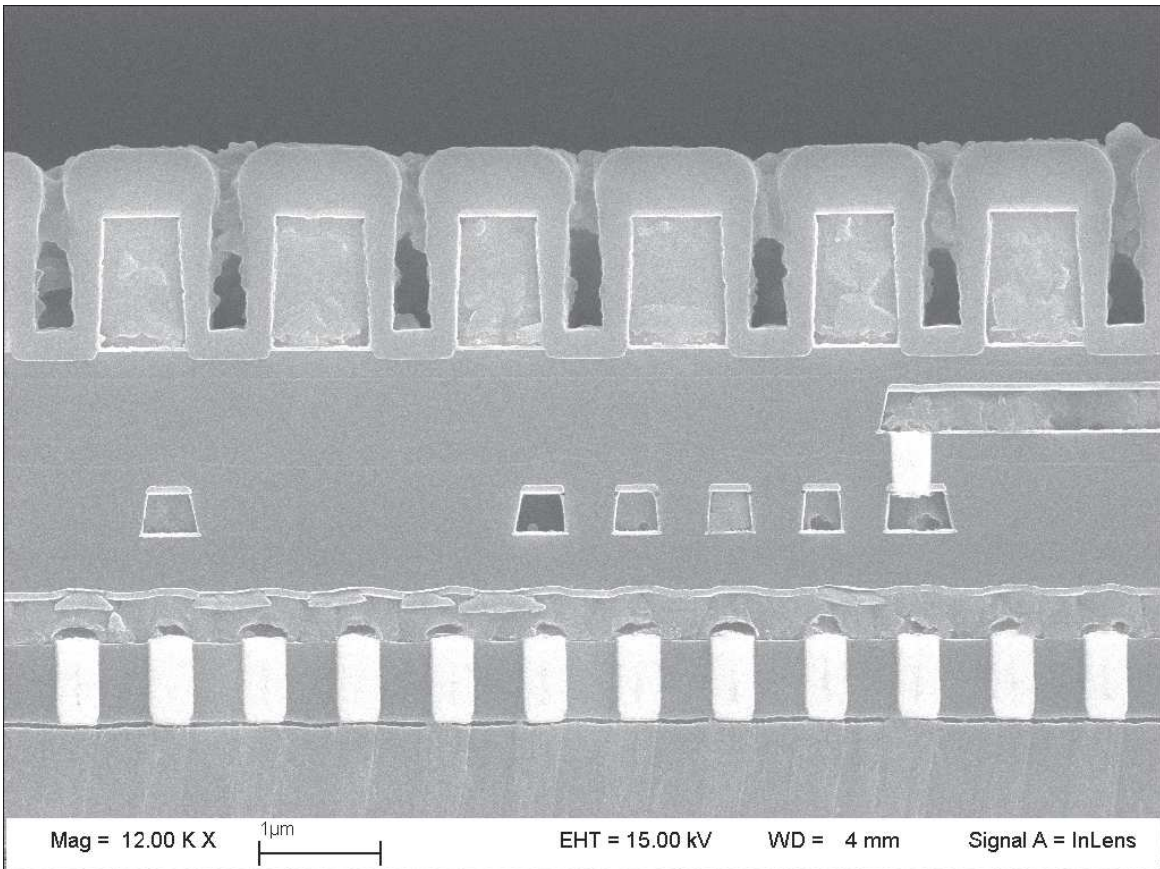
138

La coupe d'une puce



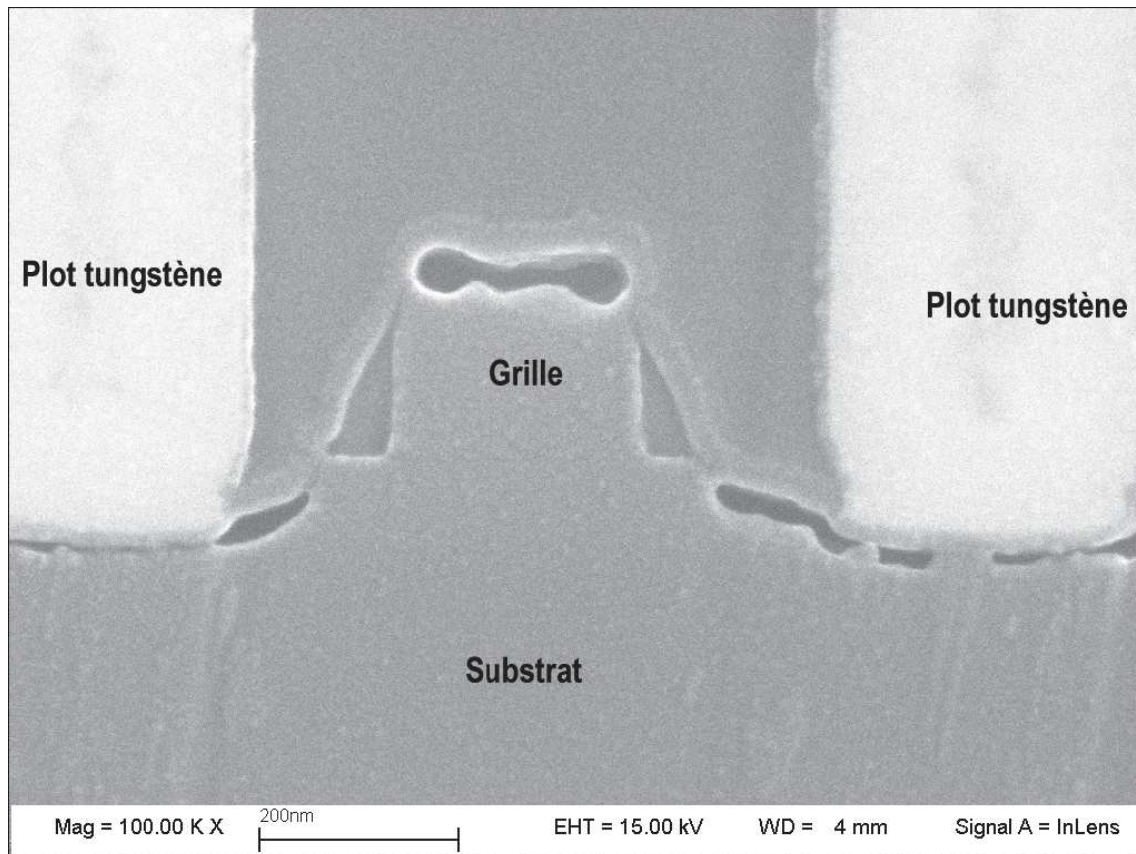
139

La coupe d'une puce



140

La coupe d'un transistor



141

La sécurité software

contrôles d'accès aux données

maintien de l'intégrité des données

entrées/sorties sécurisées

migration du code

142

La sécurité de l'env. de production

physiquement sécurisé

régulièrement audité

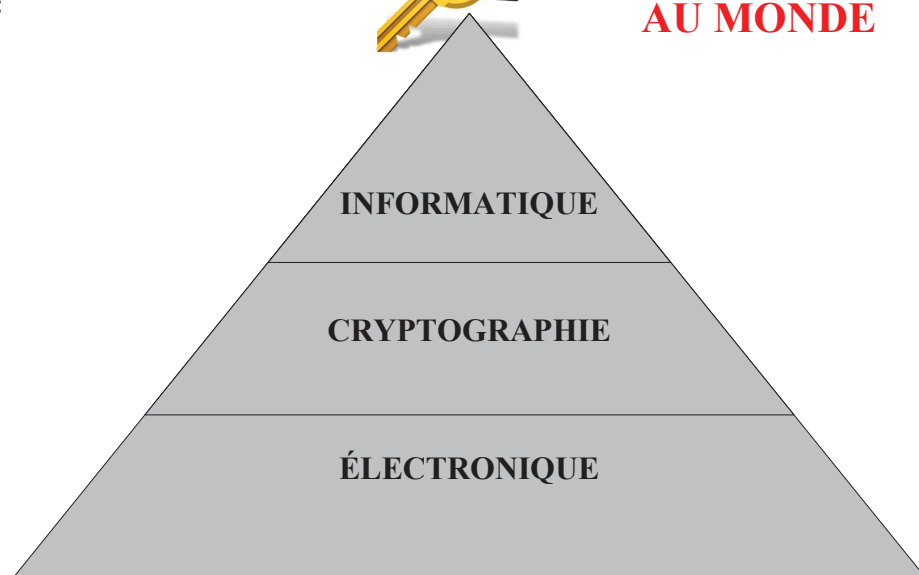
143

La sécurité

Pour résumer :



**LE PÉRIPHÉRIQUE
LE PLUS SÉCURISÉ
AU MONDE**



*La haut niveau de sécurité est assuré
par la triple alliance de l'électronique,
de l'informatique et de la cryptographie*

144

Les méthodes formelles et le test

Conception formelle de certains partie de l'OS (voire de l'OS complet)

Preuve formelle sur des aspects sécuritaire

Génération de jeux de tests associés aux spécifications

...

145

L'évaluation sécuritaire et la certification

La certification

Objectifs :

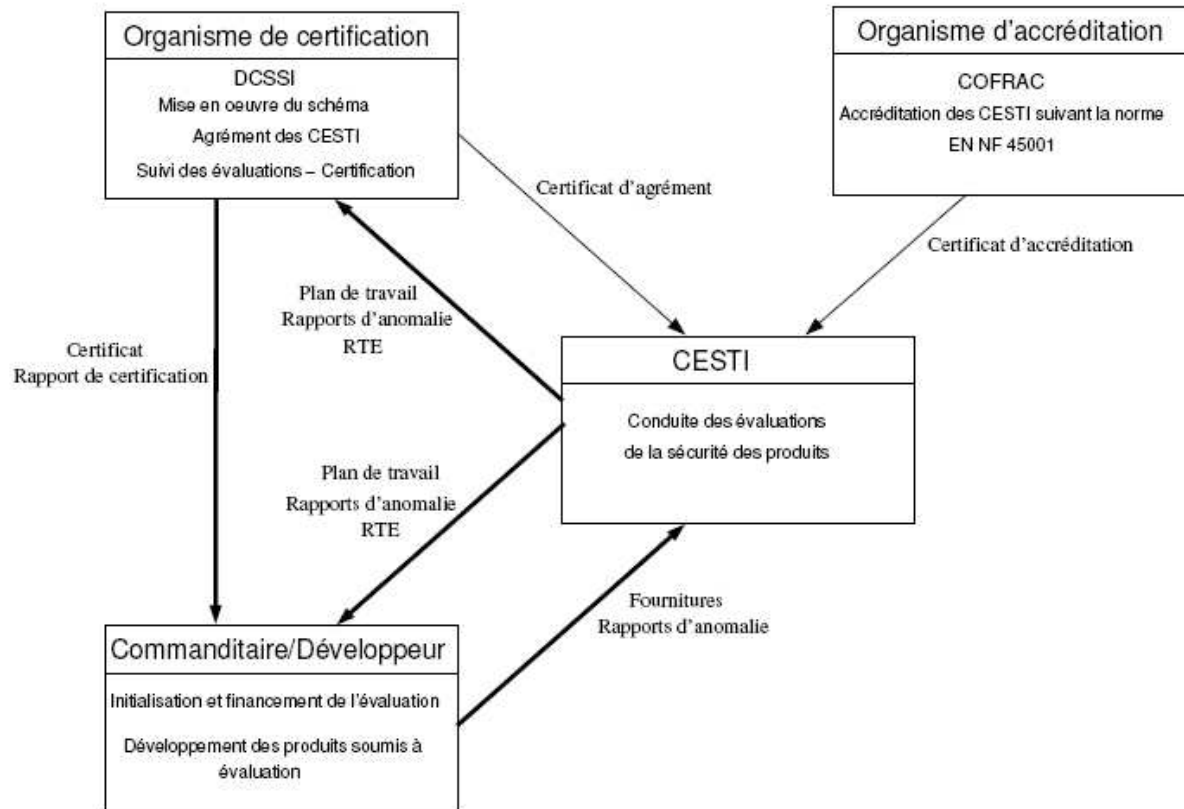
- Garantir le niveau de sécurité d'un produit TI (confidentialité, intégrité, disponibilité).
- Bénéficier des accords de reconnaissance mutuelle.

Avantages pour :

- les utilisateurs : comparer sur une base objectif ;
- les industriels : prouver leur compétence et d'étendre leurs marchés ;
- les autorités d'homologation : s'assurer que les objectifs de sécurité sont satisfaits et que leur pays ne court pas de risque majeur à utiliser des produits TI !

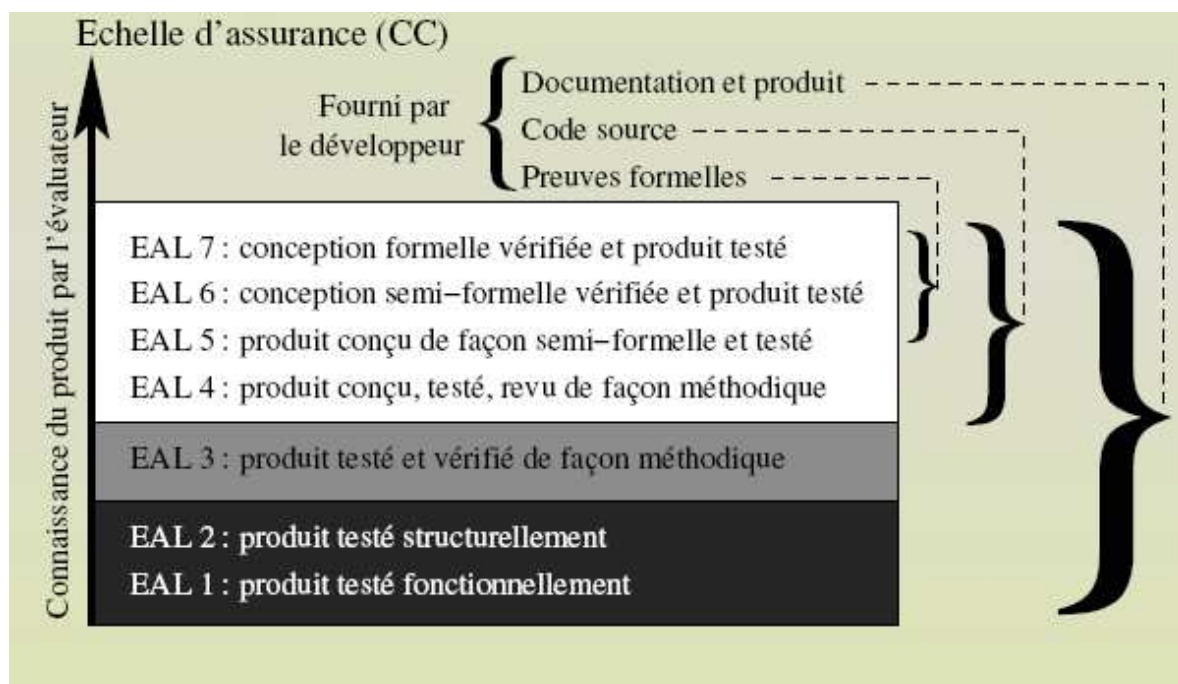
146

Le schéma Français



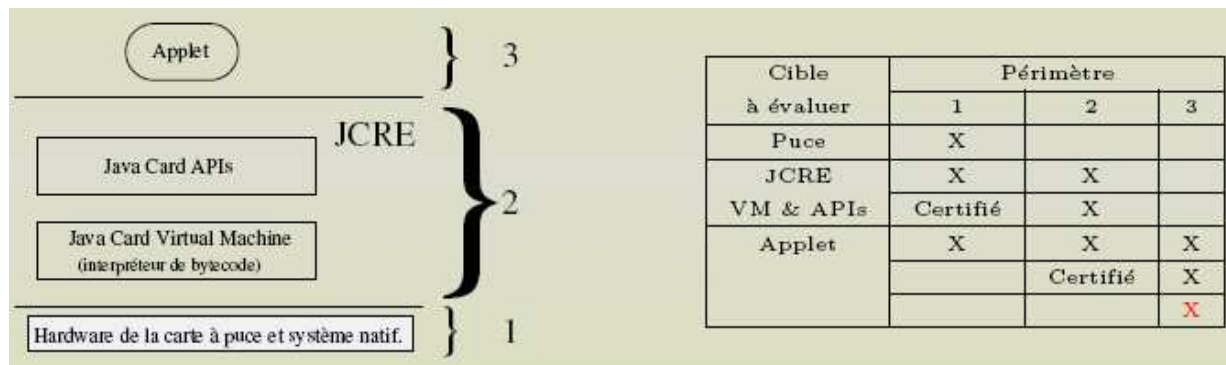
7

L'échelle d'assurance



Le périmètre d'évaluation

Défini dans la cible de sécurité dans la description de la cible d'évaluation.



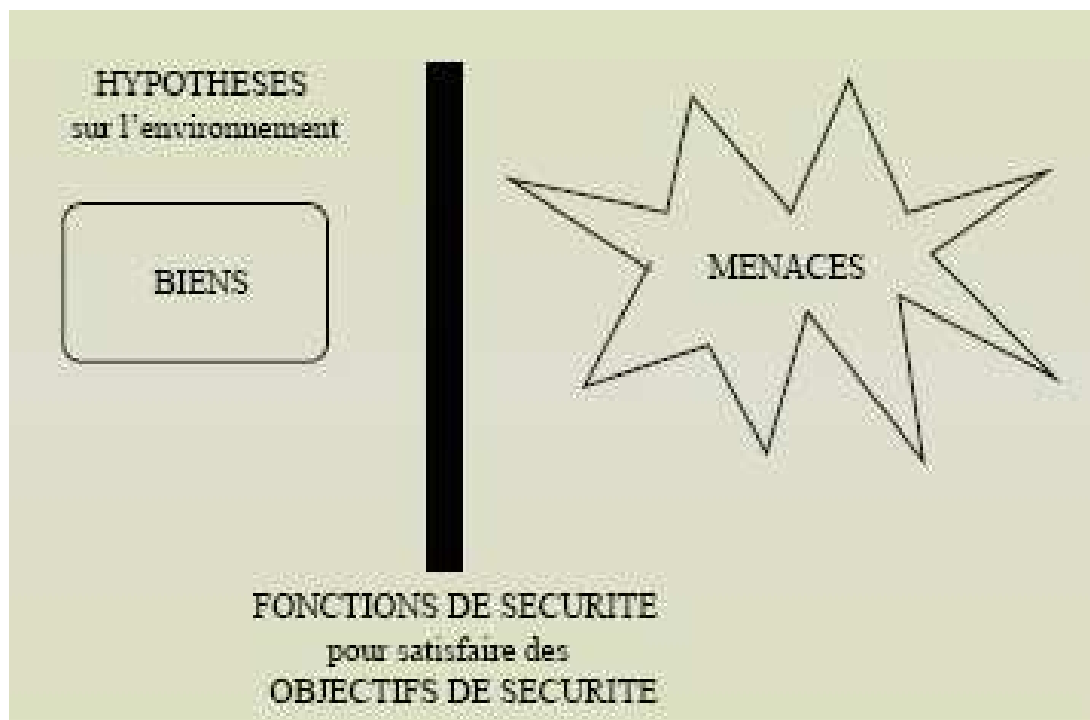
Nouveaux problèmes apportés par les applets Java Card :

- pas de définition précise du périmètre d'évaluation ;
- pas de méthodologie d'évaluation d'une applet ;
- la multi-application.

149

Les bases des critères communs

Critères Communs (ISO15408)



150



La carte à puce

Les attaques

151



Différentes catégories

Non invasives

Le produit reste fonctionnel

Invasives

Le produit est partiellement endommagé

152

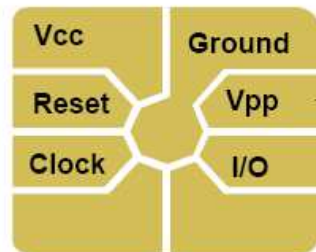
Non invasives

modification des conditions opérationnelles (V_{cc} , F)

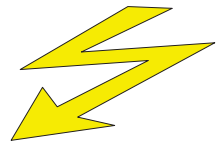
modification de la température

modification des rayonnements lumineux (UV, rayon X, lumière blanche, IR, ...)

injection de fautes (glitches, rayonnements lumineux)



- Attack on VPP
- Using nail polish
- Card not debited...



(injection de fautes sur la JVM : <http://www.cs.princeton.edu/~sudhakar/papers/>)

attaques sur les canaux cachés

Temps

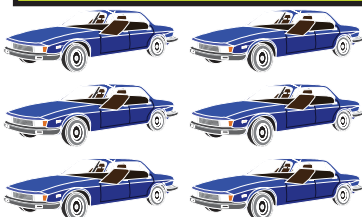
Consommation électrique

Émissions électromagnétiques

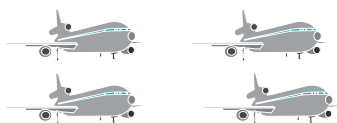
153

Êtes vous prêt ?

Politique : Les jouets cassés ne sont pas payés



voiture = \$3

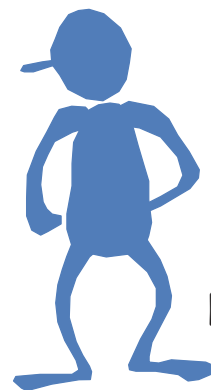


avion = \$5

Jack



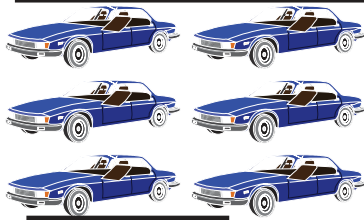
Dino



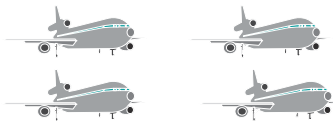
Dino achète des jouets à Jack

Êtes vous prêt ?

Politique : Les jouets cassés ne sont pas payés

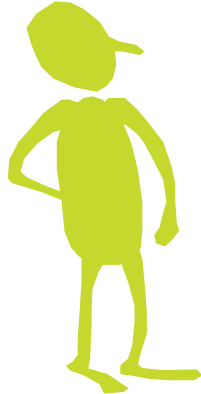


voiture = \$3

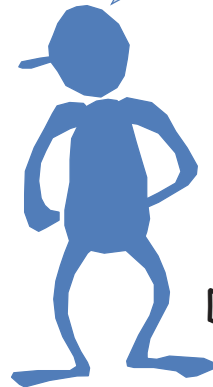


avion = \$5

Jack



Je voudrais acheter
3 avions



Dino

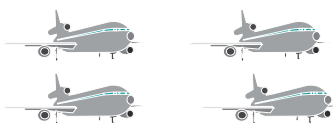
Dino achète des jouets à Jack

Êtes vous prêt ?

Politique : Les jouets cassés ne sont pas payés

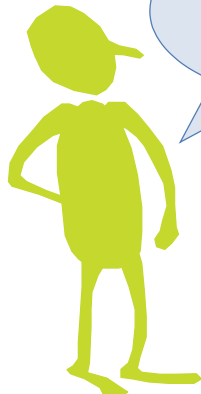


voiture = \$3



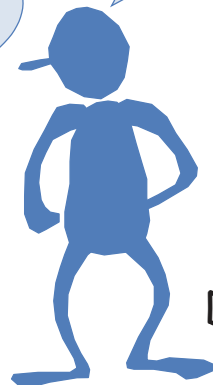
avion = \$5

Jack



Ça fera \$15

Je voudrais acheter
3 avions

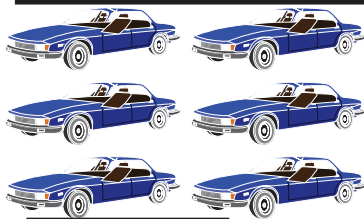


Dino

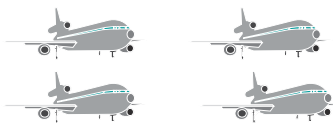
Dino achète des jouets à Jack

Êtes vous prêt ?

Politique : Les jouets cassés ne sont pas payés



voiture = \$3



avion = \$5

Jack

Ça fera \$15

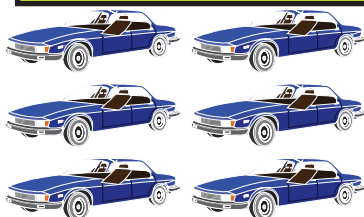
OK, s'il vous plaît
envoyez le par DHL

Dino

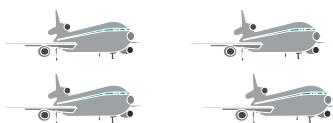
Dino achète des jouets à Jack

Êtes vous prêt ?

Politique : Les jouets cassés ne sont pas payés



voiture = \$3



avion = \$5

Jack

Comment payez vous ?

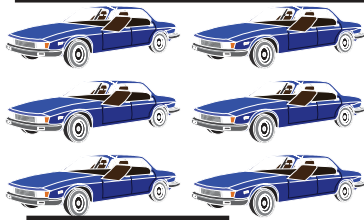
OK, s'il vous plaît
envoyez le par DHL

Dino

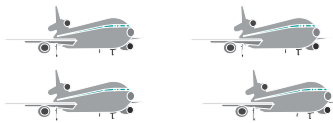
Dino achète des jouets à Jack

Êtes vous prêt ?

Politique : Les jouets cassés ne sont pas payés

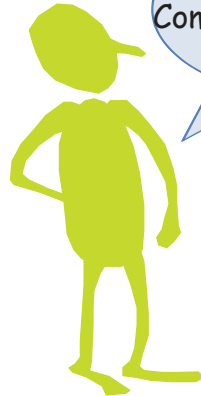


voiture = \$3



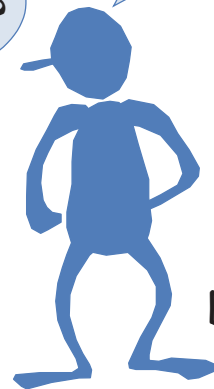
avion = \$5

Jack



Comment payez vous ?

J'enverrai \$15
en mandat postal

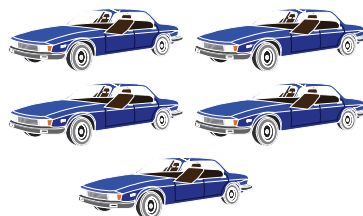
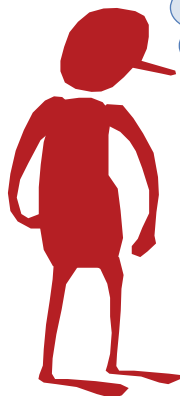


Dino

Dino achète des jouets à Jack

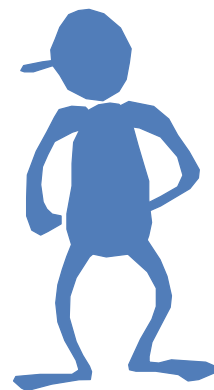
*Le facteur veut savoir ce que Dino
a acheté pour \$15*

Qu'est ce qu'il a
acheté pour \$15 ?

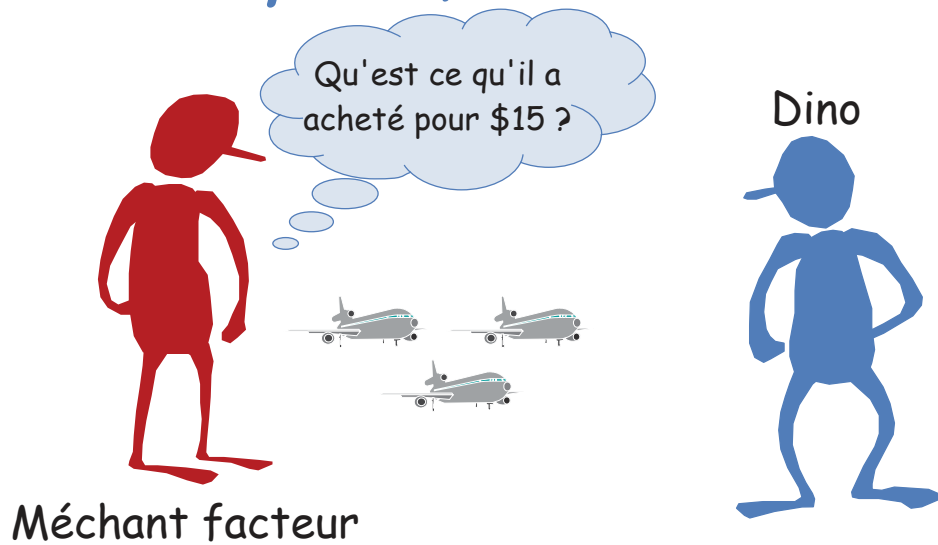


Méchant facteur

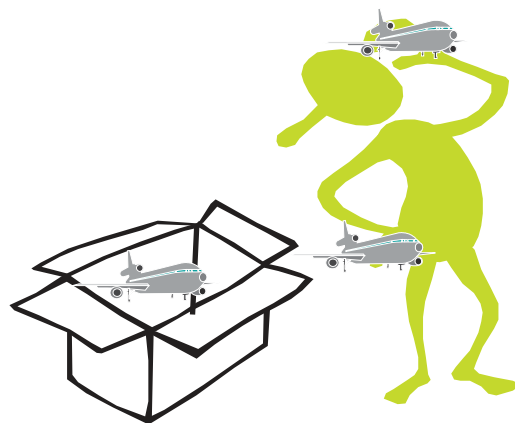
Dino



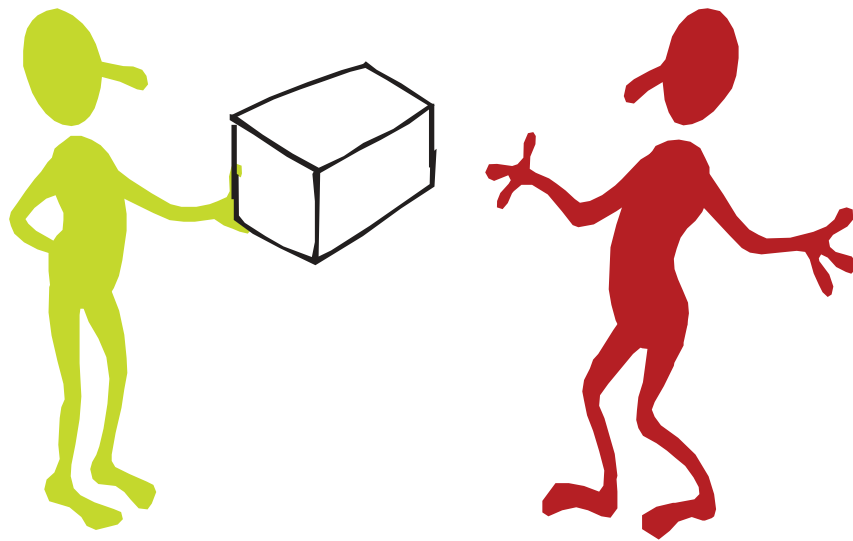
*Le facteur veut savoir ce que Dino
a acheté pour \$15*



*Pendant ce temps, Jack prépare le
colis DHL*



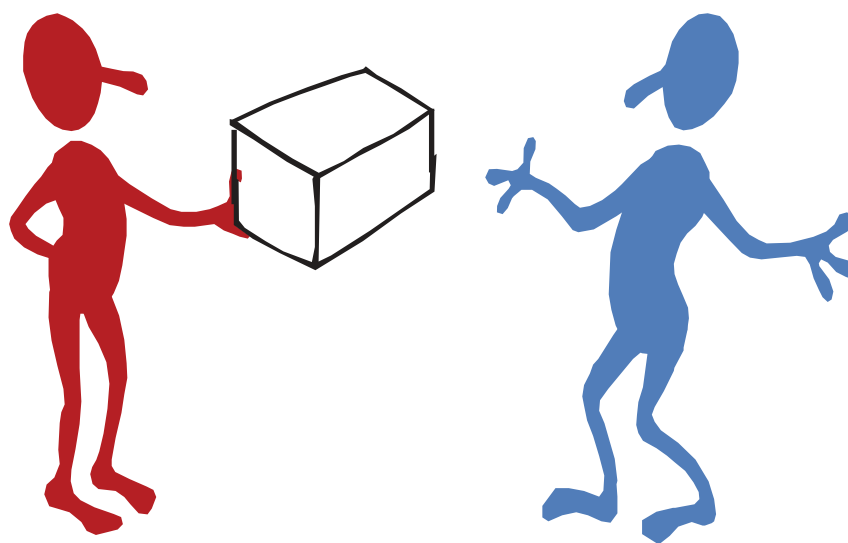
Et le donne au facteur



*Qui tape dedans assez fort pour
casser **un** jouet*



Et le donne à Dino



*Une semaine plus tard, il surveille
le mandat postal de Dino...*



La leçon apprise : Les attaques en faute peuvent aussi permettre d'extraire des secrets de tokens!

Les fautes matérielles peuvent être venir de diverses sources :
Glitches de tension, faisceau lumineux, faisceau laser ...

Utilisable sur la **signature RSA**, le chiffrement DES, ...
Il y a des détecteurs lasers embarqués sur les puces récentes !
(voir thèse d'Alexandre SARAFIANOS)

Attaques software/hardware

Pour forger des pointeurs et ensuite ...



<http://www.cs.princeton.edu/~sudhakar/papers/memerr-slashdot-commentary.html>

167

Les canaux cachés

le temps d'exécution

=> nombre de cycle d'une instruction ou d'un algorithme.

la consommation de courant

=> Les modifications rapides de la tension et de l'intensité du courant au sein du même composant sont à la base des émissions du circuit car ils conduisent des courants RF à l'intérieur et à l'extérieur du chip.

les émissions électromagnétiques

=> Les courants RF entraînent un rayonnement électromagnétique.

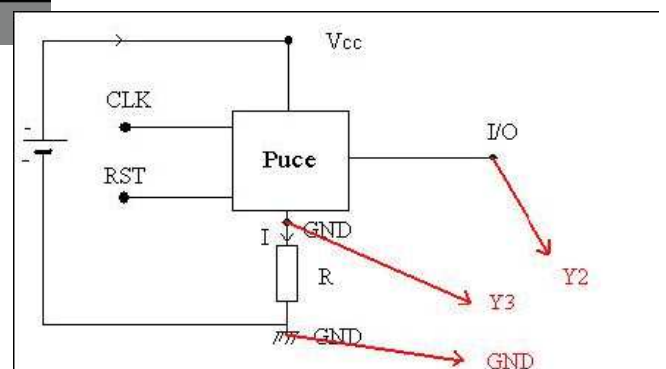
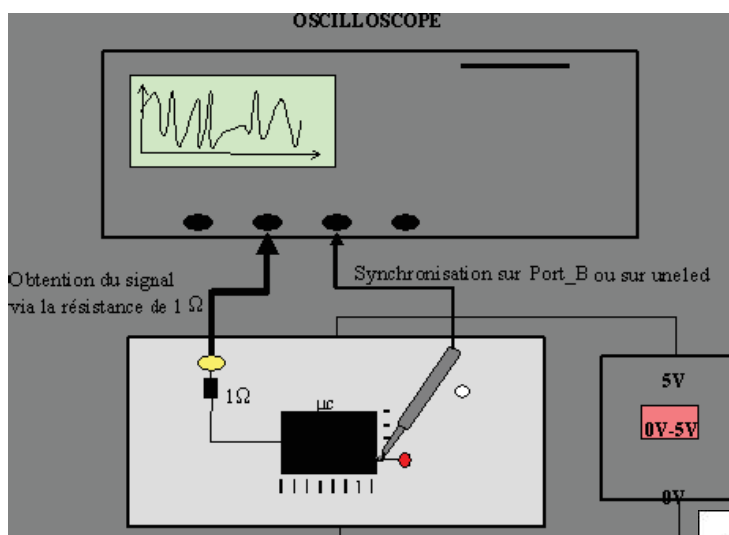
168

Le matériel

- un oscilloscope numérique,
- un lecteur de carte à puce,
- un pc équipé de cartes d'acquisition et de logiciels mathématique pour le traitement des données,
- une sonde CEM si on veut étudier les émissions électromagnétiques.



Le montage pour suivre la consommation en courant



La “timing attack”

Cette attaque consiste à mesurer le temps d’exécution d’un algorithme.

=> Révèle des informations sur les opérations et/ou les opérandes.

Nécessite souvent :

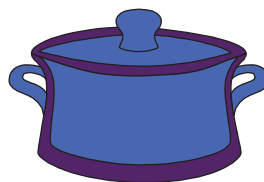
un grand nombre d’exécution à messages choisis,
un traitement statistique des résultats obtenus.

Exemple:

171

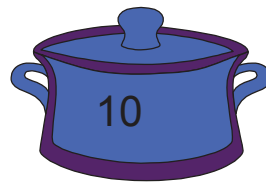
Un jeu

- Mettez 28 dans un des pots et 10 dans l'autre :



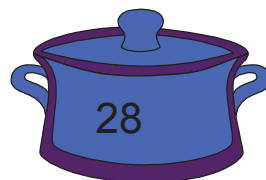
Un jeu

- Mettez 28 dans un des pots et 10 dans l'autre :



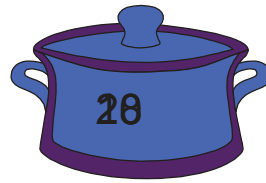
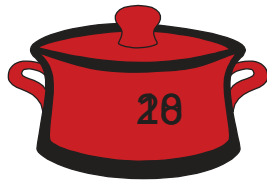
Un jeu

- Mettez 28 dans un des pots et 10 dans l'autre :



Un jeu

- Mettez 28 dans un des pots et 10 dans l'autre :



- Je vous demande de multiplier le contenu du pot bleu par 10 et le contenu du pot rouge par 7, d'additionner les deux résultats et de me dire si la somme est paire ou impaire.
- Est-ce que votre réponse est suffisante pour révéler le contenu de chaque pot ?

Est-ce que ce jeu à un sens ?

Est-ce que ce jeu à un sens ?

- Et bien, normalement non :

$$28 \times 7 + 10 \times 10 = 296 \quad \text{est un nombre pair}$$

et

$$10 \times 7 + 28 \times 10 = 350 \quad \text{est aussi un nombre pair...}$$

- Pourtant, juste en observant le temps pris pour donner la réponse (le calcul mental conduisant à 296 est plus compliqué que celui conduisant à 350), on peut dire quelle valeur était dans quel pot !

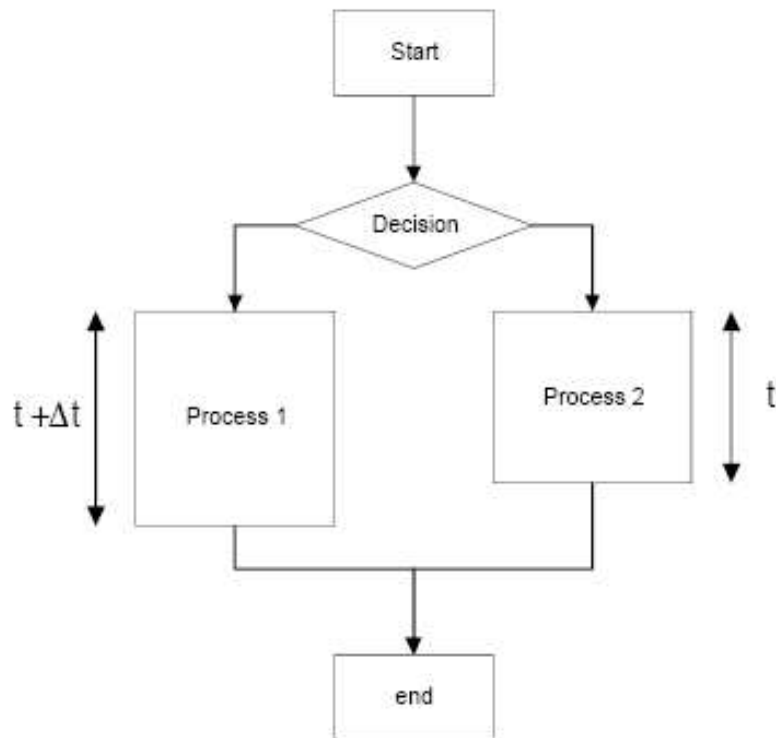
Conclusion

Une observation externe du temps de calcul d'une carte peut conduire à la fuite d'informations secrètes vers l'extérieur (par exemple des clés, des PINs, etc.).

Les timing attacks sont apparues au début des années 1990.

La leçon apprise : **Les logiciels actuels pour cartes s'exécute en temps constant.**

La “timing attack”



179

La “timing attack” sur le PIN (1/2)

```

for ( i = 0 ; i <= 7; i++)
    if ( pinCarte [ i ] != pinPresente [ i ] )
        return false ;
return true ;
    
```

 **pinCarte**
1,3,2,4,250,7,9,134

Présentons un PIN quelconque et déroulons le programme :

pinPresente
0,0,0,0,0,0,0,0

cas du premier octet FAUX

```

i = 0
i <= 7 ? OUI
( pinCarte [ i ] != pinPresente [ i ] ) ? OUI
return false
    
```

cas du premier octet VRAI

```

i = 0
i <= 7 ? OUI
( pinCarte [ i ] != pinPresente [ i ] ) ? NON
i++
i <= 7 ? OUI
( pinCarte [ i ] != pinPresente [ i ] ) ? OUI
return false
    
```

The diagram shows a vertical timeline of execution steps. A red bracket labeled Δt spans the steps from the first comparison to the final return statement, indicating the time taken for the first correct digit.

pinPresente
1,0,0,0,0,0,0,0

180

La “timing attack” sur le PIN (2/2)

```
for ( i = 0 ; i <= 7; i++)  
    if ( pinCarte [ i ] != pinPresente [ i ] )  
        return false ;  
return true
```

 pinCarte
1,3,2,4,250,7,9,134

Présenter les n valeurs possibles de pinPresente[0] (256 valeurs) (n,0,0,0,0,0,0,0).

Mesurer la durée d'exécution de la commande T pour les n valeurs.

Calculer T[n0] le maximum des T

– T[n0] = max(T[n]); n = 0; ... ; 255

n0 est la solution pour pinCarte[0]

Itérer sur tous les pinPresente[i]

Nombre d'essais : $8 * 256 = 2048$ (contre 256^8 en force brute)

pinPresente

0,0,0,0,0,0,0,0

1,0,0,0,0,0,0,0

...
255,0,0,0,0,0,0,0

On peut même s'arrêter avant n=255 !

181

Exemples de protection (1/2)

```
alea = random ( 0 , 7 ) ; // alea situe dans l'intervalle [0 ; 7]  
for ( i = 0 ; i <= 7; i++) {  
    octet = ( alea + i ) mod 8 ;  
    if ( pinCarte [ octet ] != pinPresente [ octet ] )  
        return false ;  
}  
return true ;
```

Nombre d'essais en moyenne : $8 * 2048$ (contre 256^8 en force brute)

Possibilité d'attaques de type MasterMind.

182

Exemples de protection (2/2)

```
boolean test = true ;
for ( i = 0 ; i <= 7; i++) {
    if ( pinCarte [ i ] != pinPresente [ i ] )
        test = test && false ;
    else
        test = test && true ;
}
return test ;
```

équivalent à :

```
boolean test = true ;
for ( i = 0 ; i <= 7; i++)
    test = ( pinCarte [ i ] == pinPresente [ i ] ) && test ;
return test ;
```

Et pas test = test && (pinCarte [i] == pinPresente [i]);

Des attaques en temps existent sur d'autres supports (e.g. PC).

Voir par exemple : *Cache-timing attacks on AES* ou *CACHE*

MISSING FOR FUN AND PROFIT

183

Et sinon vous savez codé un vérification de PIN
sécurisé maintenant ?

184

And what about security ?

- Write **securely** a function that checks that an array is equal to another with less than 3 trials.

```
boolean verify (byte[] buffer, short ofs, byte len)
{...}
```

- Need a constant `maxTries` initialized at 3,
- A field which memorize the trial number, says `triesLeft`

```
boolean verify (byte[] buffer, short ofs, byte len)
{
    // No comparison if PIN is blocked
    if (triesLeft <= 0)
        return false ;

    // Main comparison
    for(short i=0; i < len; i++)
        if (buffer[ofs+i] != pin[i])
        {
            triesLeft-- ;
            authenticated[0] = false ;
            return false ;
        }

    // Comparison is successful
    triesLeft = maxTries ;
    authenticated[0] = true ;
    return true ;
}
```

Check len before ;-)

```

boolean verify (byte[] buffer, short ofs, byte len)
{
    // No comparison if PIN is blocked
    if (triesLeft <= 0)
        return false ;

    // First decrements the number of remaining tries
    triesLeft-- ;

    // Main comparison
    boolean equal = true ;
    for(short i=0; i < len; i++)
        equal = (equal && (buffer[ofs+i] != pin[i])) ;// cst time

    if (!equal) {
        // Comparison failed
        authenticated[0] = false ;
        return false ;
    }
    else {
        // Comparison is successful
        triesLeft = maxTries ;
        authenticated[0] = true ;
        return true ;
    }
}

```

```

boolean verify (byte[] buffer, short ofs, byte len)
{
    // No comparison if PIN is blocked
    if (triesLeft <= 0)
        return false ;

    // First decrements the number of remaining tries
    triesLeft-- ;

    // Main comparison
    boolean equal = true ;
    for(short i=0; i < len; i++)
        equal = (equal && (buffer[ofs+i] != pin[i])) ;// cst time

    if (!equal) {
        // Comparison failed
        authenticated[0] = false ;
        return false ;
    }
    else {
        // Comparison is successful
        triesLeft = maxTries ;
        authenticated[0] = true ;
        return true ;
    }
}

```

No!

```

boolean verify (byte[] buffer, short ofs, byte len)
{
    // No comparison if PIN is blocked
    if (triesLeft <= 0)
        return false ;

    // First decrements the number of remaining tries
    triesLeft-- ;

    // Main comparison
    boolean equal = true ;
    for(short i=0; i < len; i++)
        equal = ((buffer[ofs+i] != pin[i]) && equal) ;// cst time

    if (!equal) {
        // Comparison failed
        authenticated[0] = false ;
        return false ;
    }
    else {
        // Comparison is successful
        triesLeft = maxTries ;
        authenticated[0] = true ;
        return true ;
    }
}

```

Yes!

```

public final static short BOOL_TRUE = (short)0x5a5a ;
public final static short BOOL_FALSE = (short)0xa5a5 ;
short equal = BOOL_TRUE ;

...

// Main comparison
for(short i=0; i < len; i++)
    equal = (short)(equal &
        ((buffer[ofs+i] != pin[i])? BOOL_FALSE:  BOOL_TRUE));

if (equal == BOOL_TRUE) {
    // Comparison is successful
    triesLeft = maxTries ;
    authenticated[0] = true ;
    return true ;
}
else {

...

```

```
// First checks the integrity of the variable
if (triesLeft != triesLeftBackup)
    takeCountermeasure() ;

// No comparison if PIN is blocked
if (triesLeft < 0)
    return false ;
```

- Correct ?

```
// First checks the integrity of the variable
if (triesLeft != triesLeftBackup)
    takeCountermeasure() ;

// No comparison if PIN is blocked
if (triesLeft < 0)
    return false ;
```

- It protects only against a writing between the last and the current one. If the attack is during the evaluation...

- RAM is safer than EEPROM

```
// Transfer in a local and check the integrity of the variable
byte t1 = triesLeft ;

if (t1 != triesLeftBackup)
    takeCountermeasure() ;

// No comparison if PIN is blocked
if (t1 < 0)
    return false ;
```

```

boolean verify (byte[] buffer, short ofs, byte len) {
    byte tl = triesLeft ;
    if (tl != (short)(~triesLeftBackup)) takeCountermeasure() ;
    if (tl < 0) return false;
    JCSystem.beginTransaction() ;
        triesLeft = --tl ;
        triesLeftBackup++ ;
    JCSystem.commitTransaction() ;
    if (triesLeft != (short)(~triesLeftBackup)) takeCountermeasure() ;
    short equal = BOOL_TRUE ;
    for(short i=0; i < len; i++)
        equal = (short)(equal &
            ((buffer[ofs+i]!=pin[i])?BOOL_FALSE:BOOL_TRUE));

    if (equal == BOOL_TRUE) {
        JCSystem.beginTransaction() ;
            triesLeft = maxTries ;
            triesLeftBackup = (byte)(~maxTries) ;
        JCSystem.commitTransaction() ;
        // Verifies the new value
        if (triesLeft!=(short)(~triesLeftBackup))
            takeCountermeasure() ;
        authenticated[0] = true ;
        return true ;
    } else {
        authenticated[0] = false ;
        return false ;
    }
}

```

```

boolean verify(byte[] buffer, short ofs, byte len)
{
    // Initializes the step counter
    short stepCounter = INITIAL_COUNTER ;

    // First checks the integrity of the variable
    byte tl = triesLeft ;
    stepCounter++ ;
    if (tl != (short)(~triesLeftBackup)) takeCountermeasure() ;
    stepCounter++ ;
    // No comparison if PIN is blocked
    if (tl < 0) return false ;
    stepCounter++ ;
    // First decrements the number of remaining tries
    JCSystem.beginTransaction() ;
        triesLeft = --tl ;
        stepCounter++ ;
        triesLeftBackup++ ;
    JCSystem.commitTransaction() ;
    stepCounter++ ;
    // Verifies the new value
    if (triesLeft != (short)(~triesLeftBackup)) takeCountermeasure() ;
}

```

```

stepCounter++ ;
short equal = BOOL_TRUE ; // Main comparison
stepCounter++ ;
for(short i=0;i<len;i++)
    equal = (short)(equal &
        ((buffer[ofs+i]!=pin[i])?BOOL_FALSE:BOOL_TRUE));
stepCounter++ ;
if (equal == BOOL_TRUE) { // Comparison is successful
    //Reset the tries to the max
    stepCounter++ ;
    JCSystem.beginTransaction() ;
    triesLeft = maxTries ;
    triesLeftBackup = (byte)(~maxTries);
    JCSystem.commitTransaction() ;
    stepCounter++ ;
    if (triesLeft!=(short)(~triesLeftBackup))
        takeCountermeasure() ;
    stepCounter++ ;
    authenticated[0] = true ;
    if (stepCounter == (short)(INITIAL_VALUE+11) )
        return true ;
} else { // Comparison failed
stepCounter++ ;
authenticated[0] = false ;
if (stepCounter == (short)(INITIAL_VALUE+9))
    return false ;
}
takeCountermeasure() ; // Should have returned at this point
}

```

Other well known rules ...

```

if (pin.isValidated() == true)
{
    // sensitive operations
}
else
{
    ISOException.throwIt((short) (0x6300+pin.getTriesRemaining()));
}

```

Other well known rules ...

```
if (pin.isValidated() == true)
{
    if (pin.isValidated() == false)
    {
        ISOException.throwIt((short) (0x6300+pin.getTriesRemaining()));
    }
    else
    {
        // sensitive operations
    }
}
else
{
    ISOException.throwIt((short) (0x6300+pin.getTriesRemaining()));
}
```

- **Références :**

<http://javacard.vetilles.com/2008/05/15/jc101-12c-defending-against-attacks/>

Ronald De Keulenaer, Jonas Maebe, Koen De Bosschere, Bjorn De Sutter. Link-time smart card code hardening. International Journal of Information Security, 2015

La consommation de courant

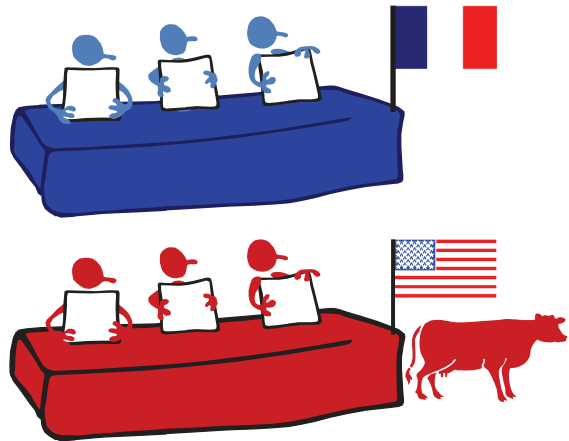
Elle est surtout utilisée dans le domaine de la cryptographie.

Il existe différentes attaques :

- la SPA (Simple Power Analysis)
- la DPA (Differential Power Analysis)
- la HODPA (High Order Differential Power Analysis)

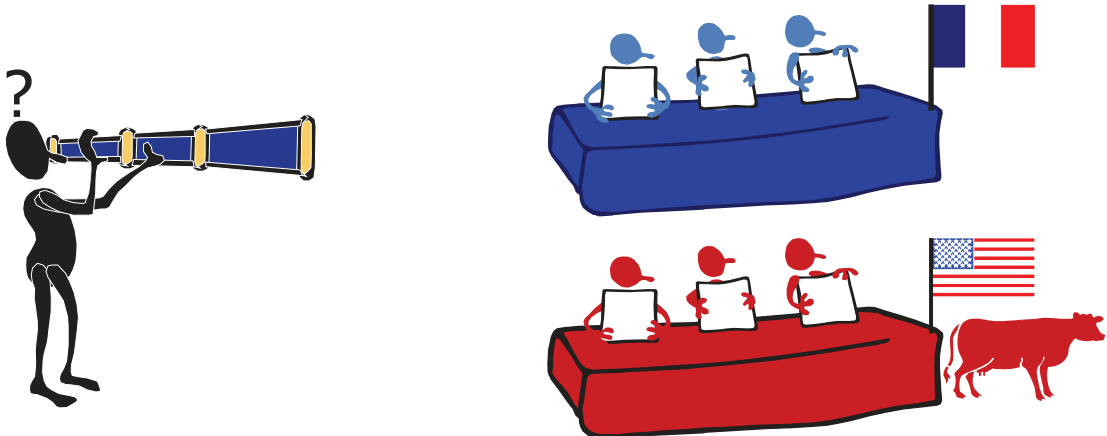
IMPORTATION DU BOEUF ?

- Seattle, 1999.
- Les représentants Français et Américains négocie sous quelles conditions le boeuf pourrait être importé en France.



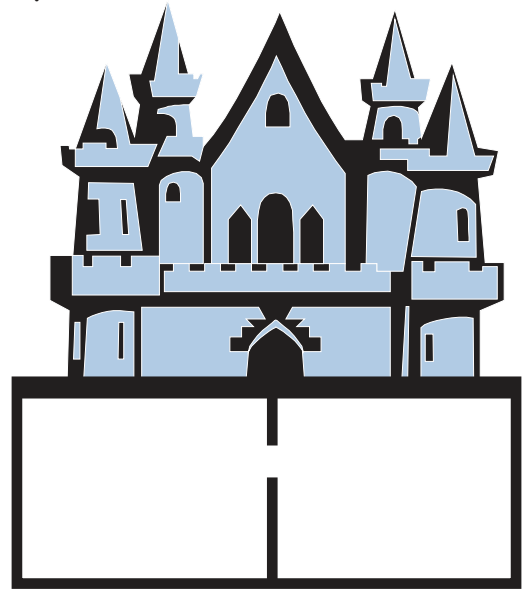
IMPORTATION DU BOEUF ?

- Seattle, 1999.
- Les représentants Français et Américains négocie sous quelles conditions le boeuf pourrait être importé en France.
- «The Sun» envoie un journaliste pour enquêter :



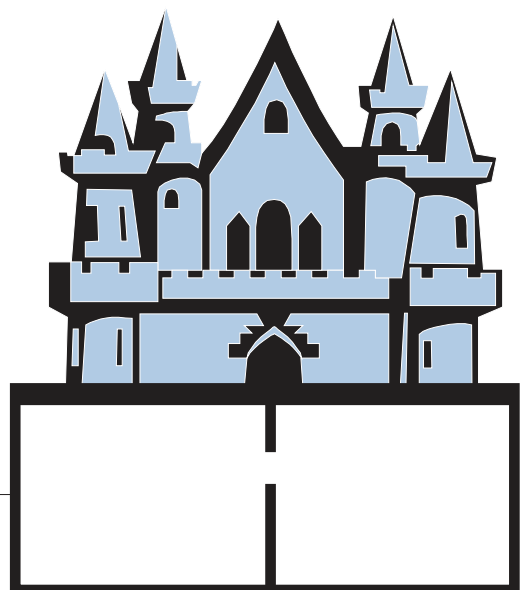
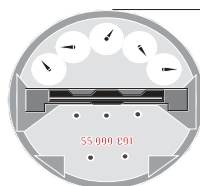
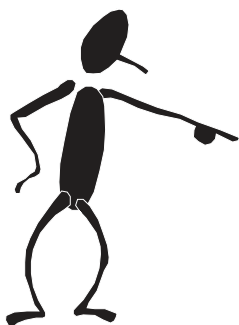
IMPORTATION DU BOEUF ?

- Mais il y a un problème technique : les négociations se déroulent dans un hotel aux vitres opaques



POWER ATTACKS

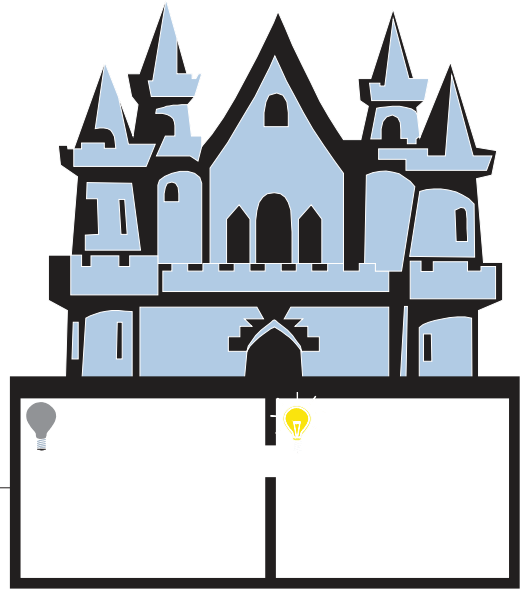
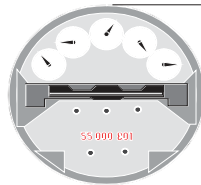
- Idée : Regarder le compteur électrique de l'hotel !



POWER ATTACKS

- Le disque tourne lentement:

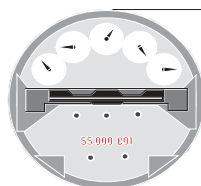
DEAL CONCLUDED



POWER ATTACKS

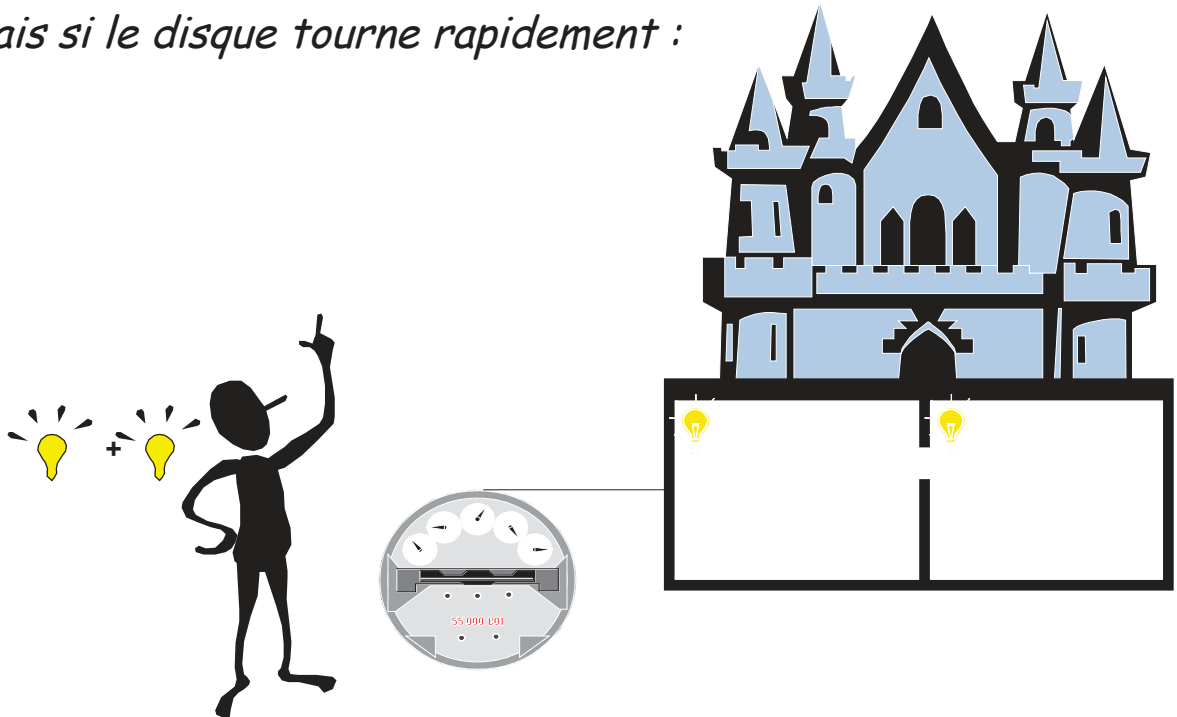
- Le disque tourne lentement:

DEAL CONCLUDED



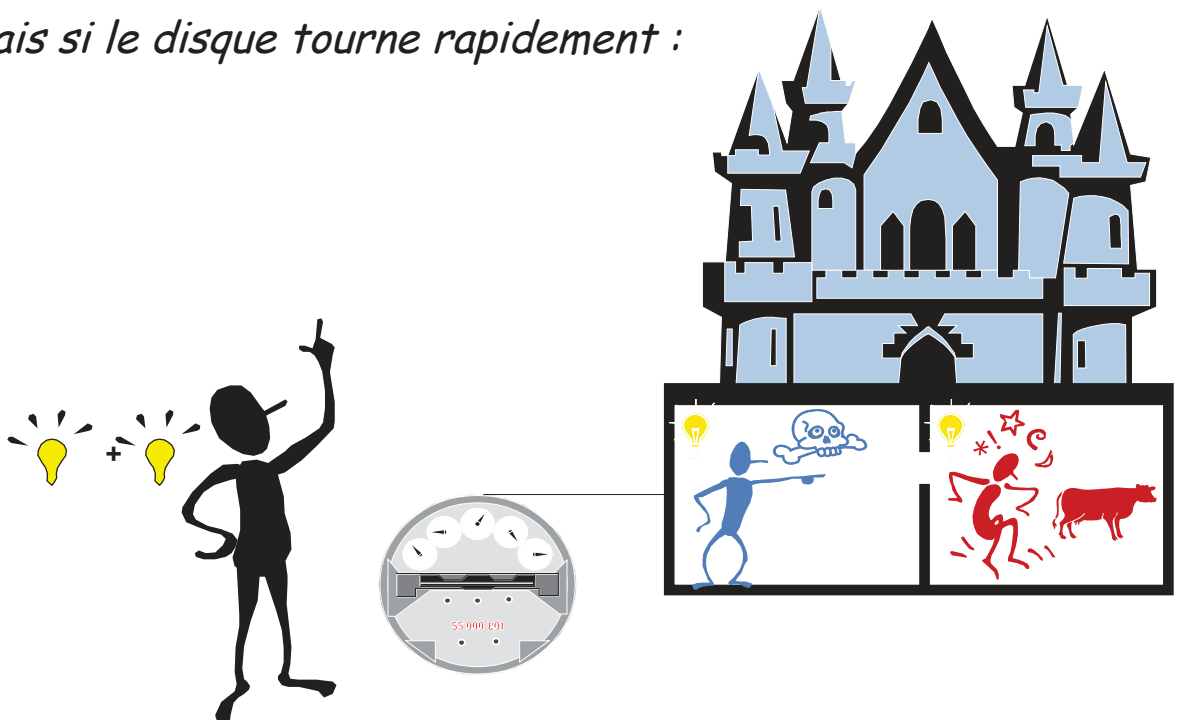
POWER ATTACKS

- Mais si le disque tourne rapidement :



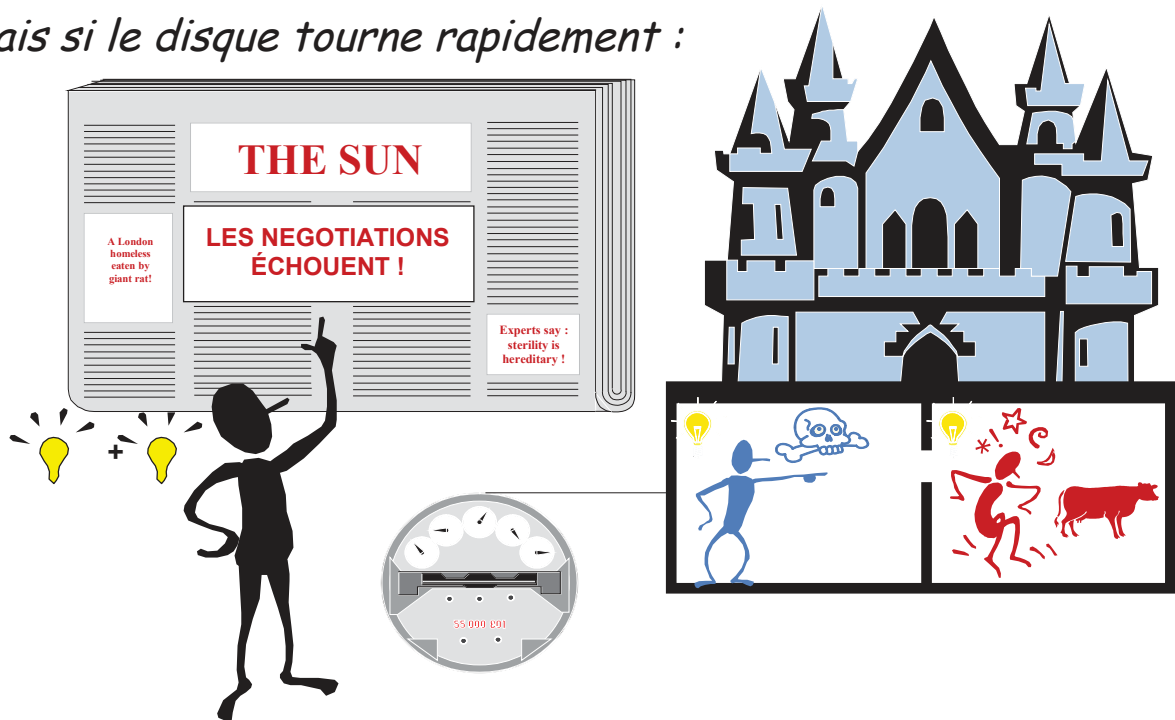
POWER ATTACKS

- Mais si le disque tourne rapidement :



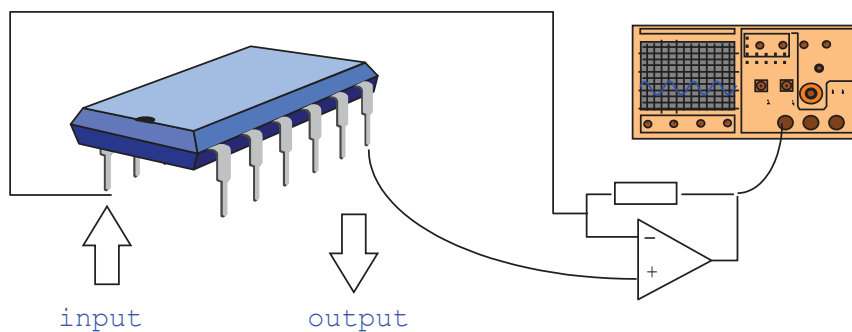
POWER ATTACKS

- Mais si le disque tourne rapidement :



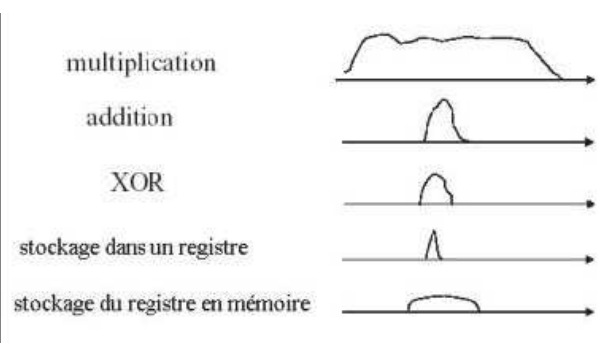
CONCLUSION

The card's current consumption **may reveal secret information.**

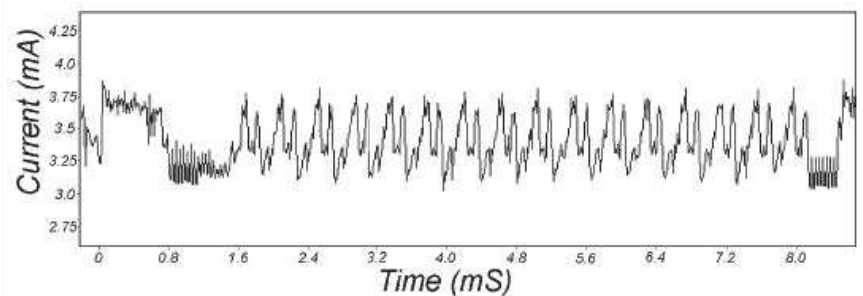


La SPA (Simple Power Analysis)

Principe : Des instructions différentes ont une trace différente.



Consommation en courant d'un DES. On peut voir la permutation initiale, suivie des 16 tours.



209

La SPA sur la signature RSA

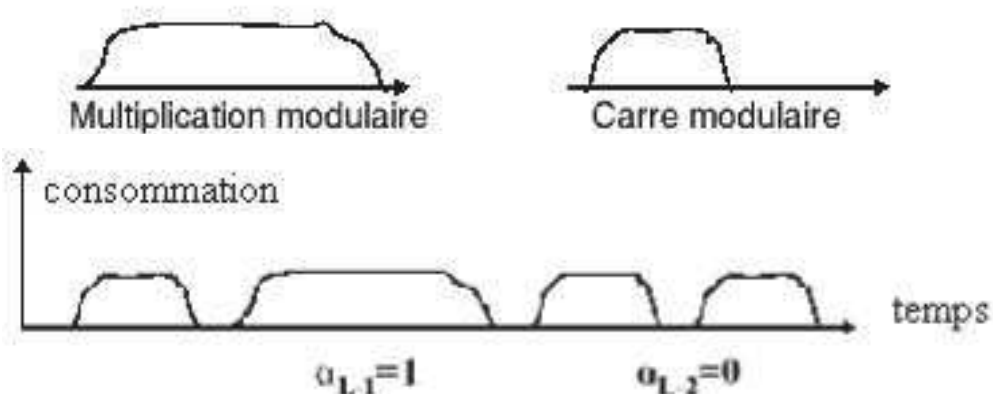
```
s = 1 ;
for ( i = L - 1 ; i >= 0 ; i-- ) {
    s = s*s mod n ;
    if ( a [ i ] == 1 )
        s = s*y mod n ;
}
```

Signature RSA : $y^a \mod n$

y est le message à signer,

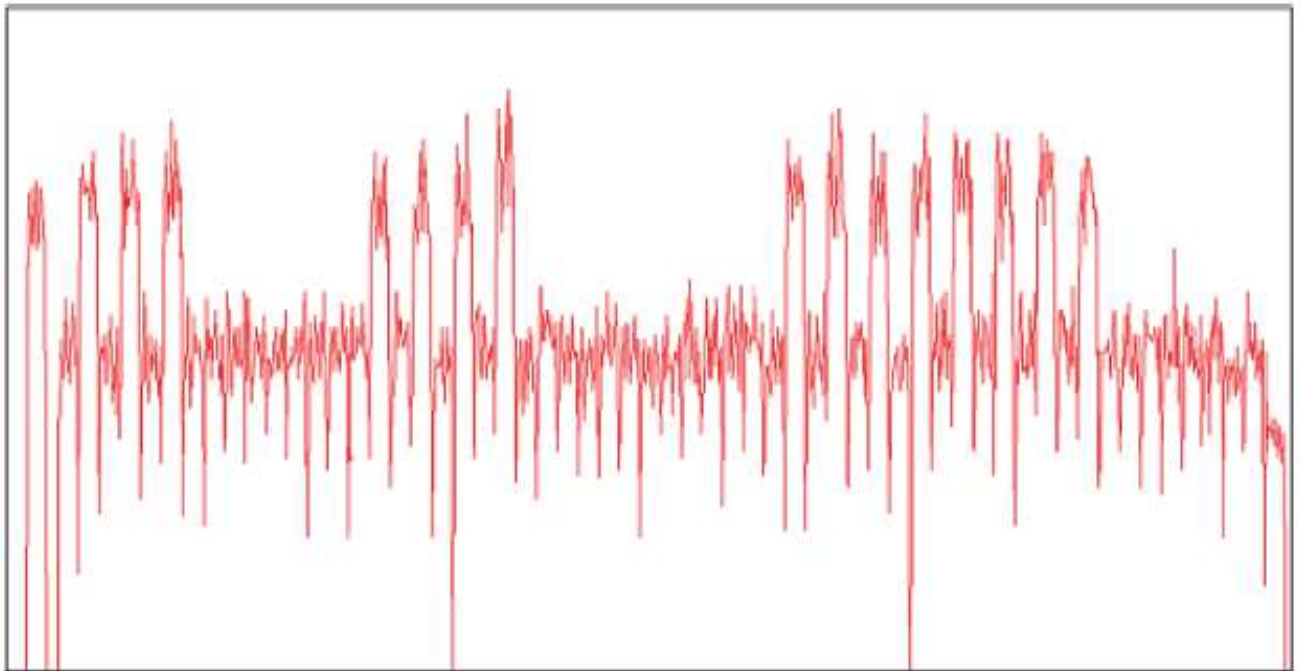
n est public,

a , l'exposant peut être considéré comme la clé secrète



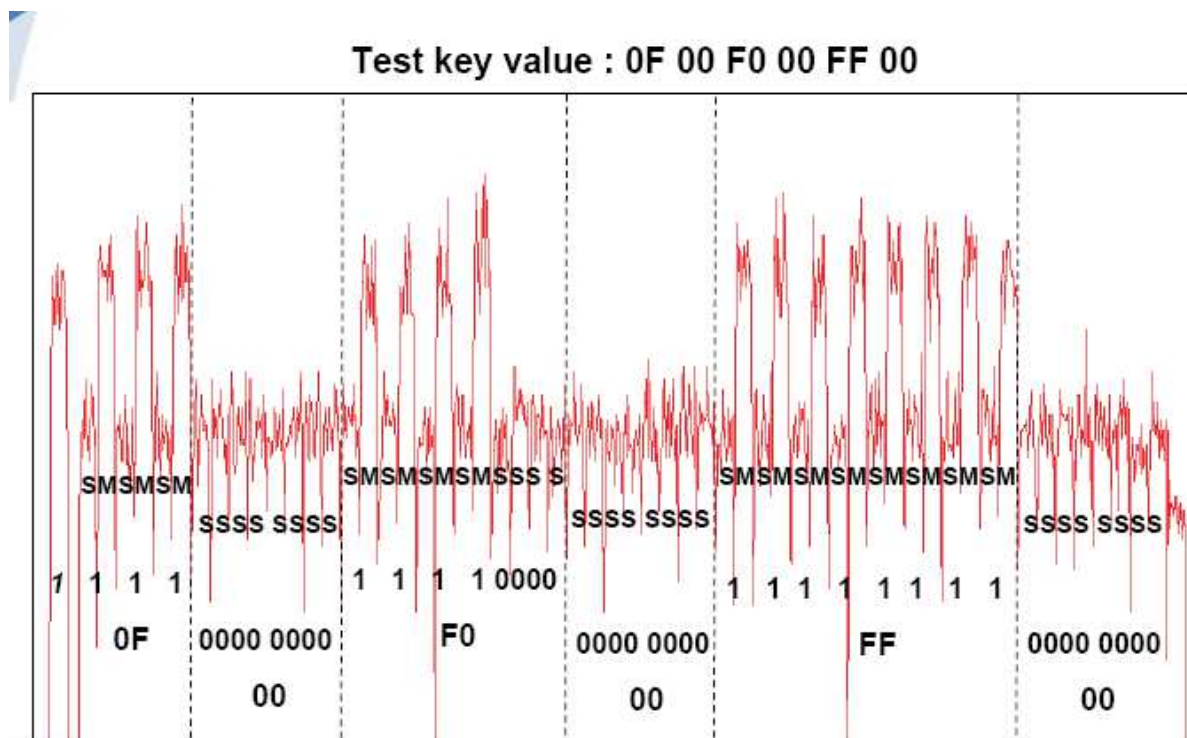
210

Exemple « réel »



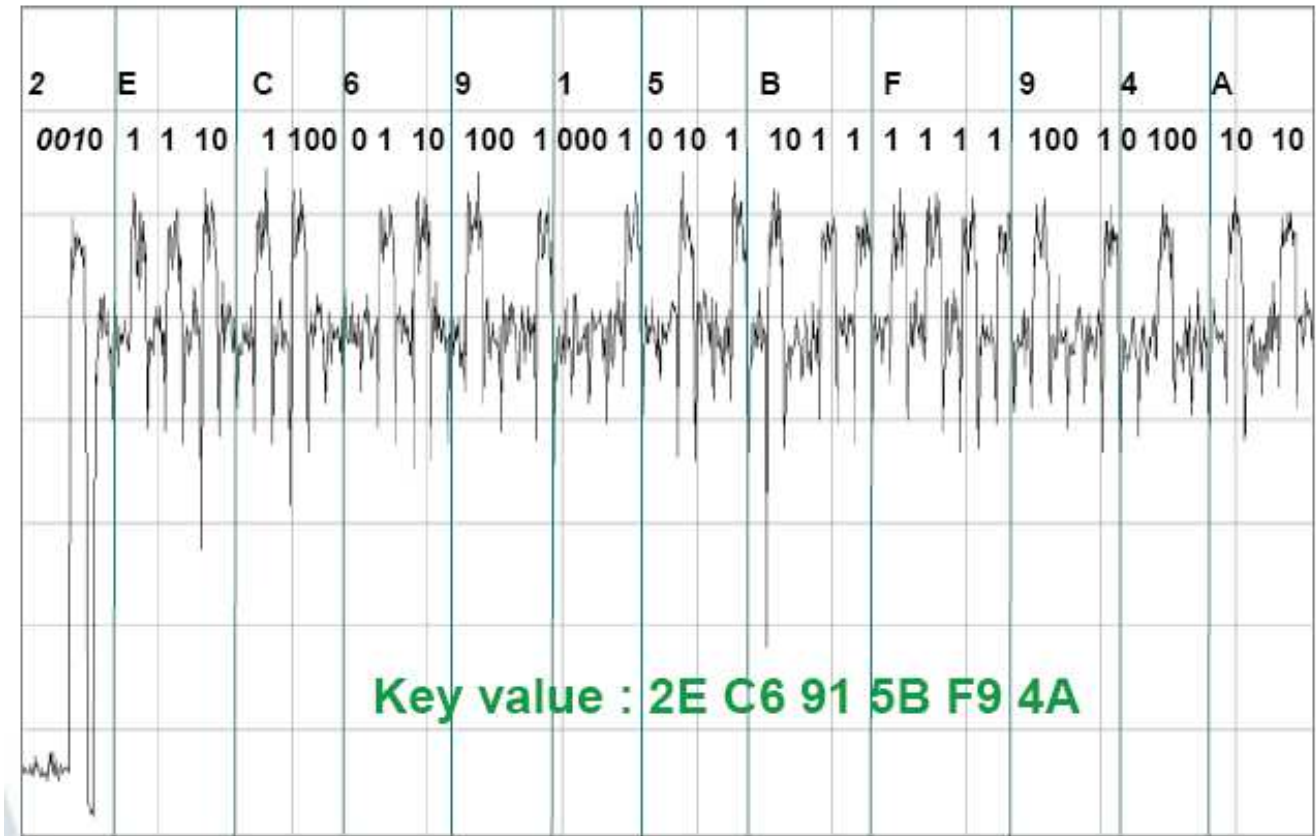
211

Exemple « réel »



212

Exemple « réel »

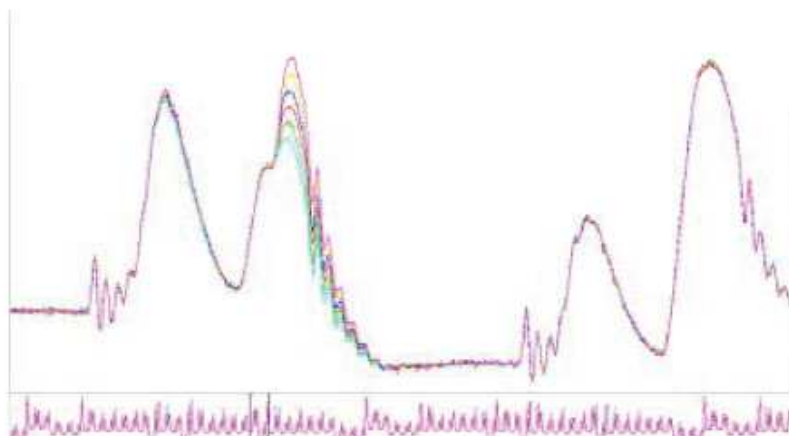


La DPA (Differential Power Analysis)

Principe : La consommation dépend des opérations effectuées (les instructions) mais aussi des opérandes.

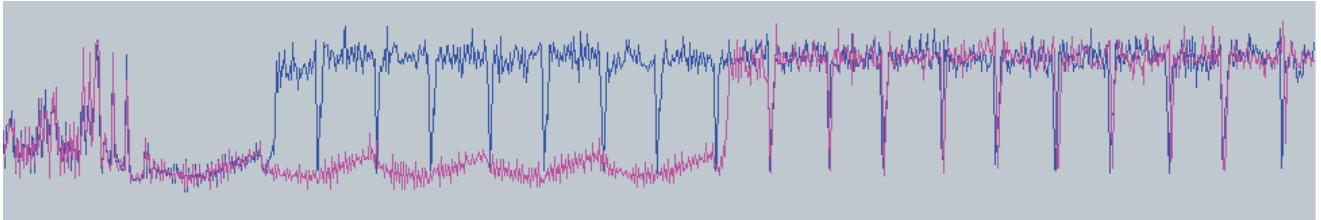
Elle utilise des fonctions statistiques adaptées à l'algorithme visé qui font ressortir des corrélations entre un bit intermédiaire a (ne dépendant que d'un fragment Kr de r bits de la clé et du message d'entrée M) et la consommation de courant.

- Based on SPA
Adding the power of statistics to separate signal from noise



CONCRETE ATTACK ON CONCRETE KEYS

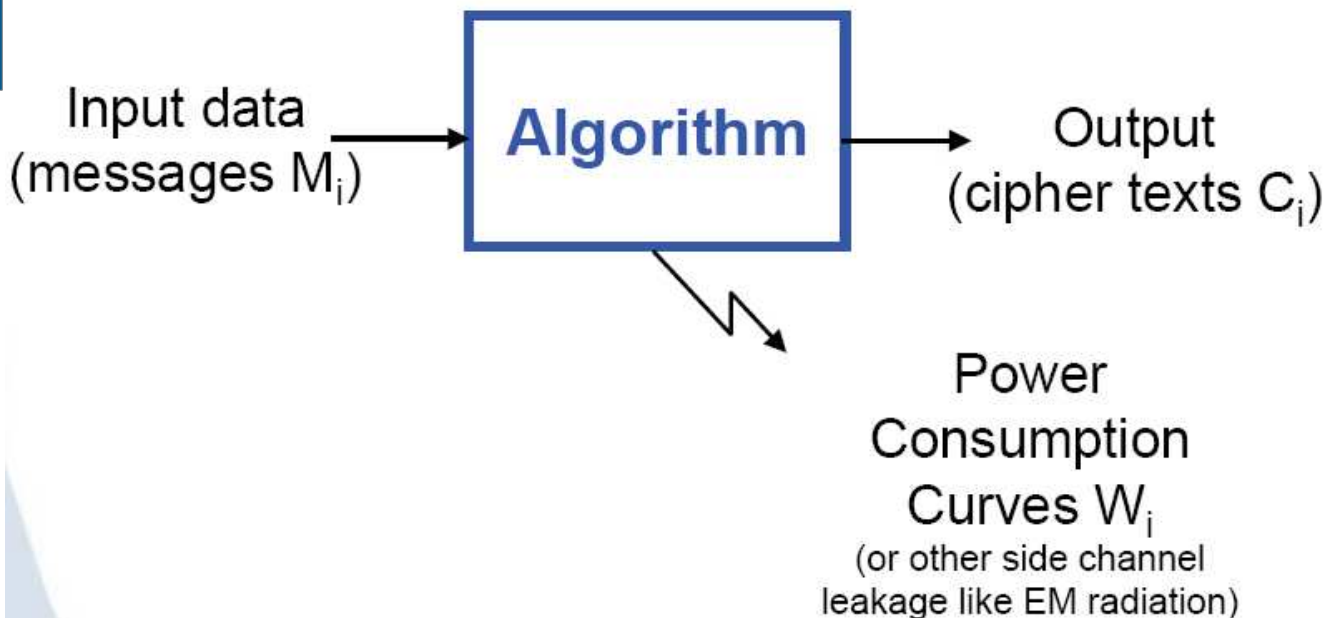
- Comparing:
 - decryption with key = 0000 1111 ...
 - decryption with key = 1111 1111 ...



token designed and manufactured in 1998...

DPA Hypothesis

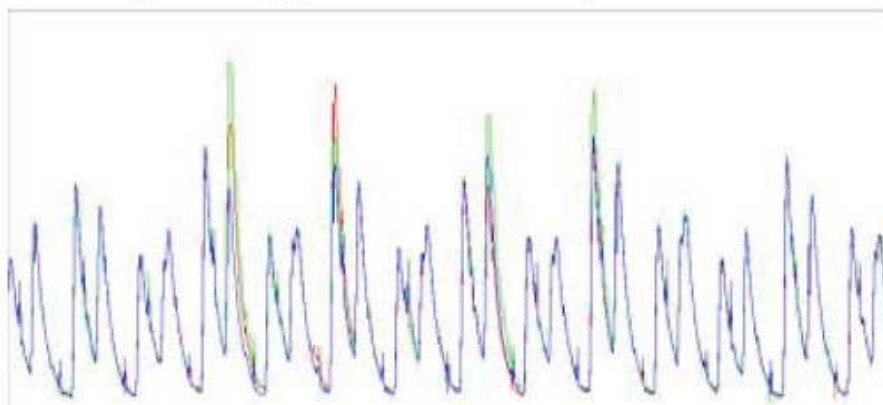
Play the algorithm N times
($100 < N < 100000$)



Acquisition procedure

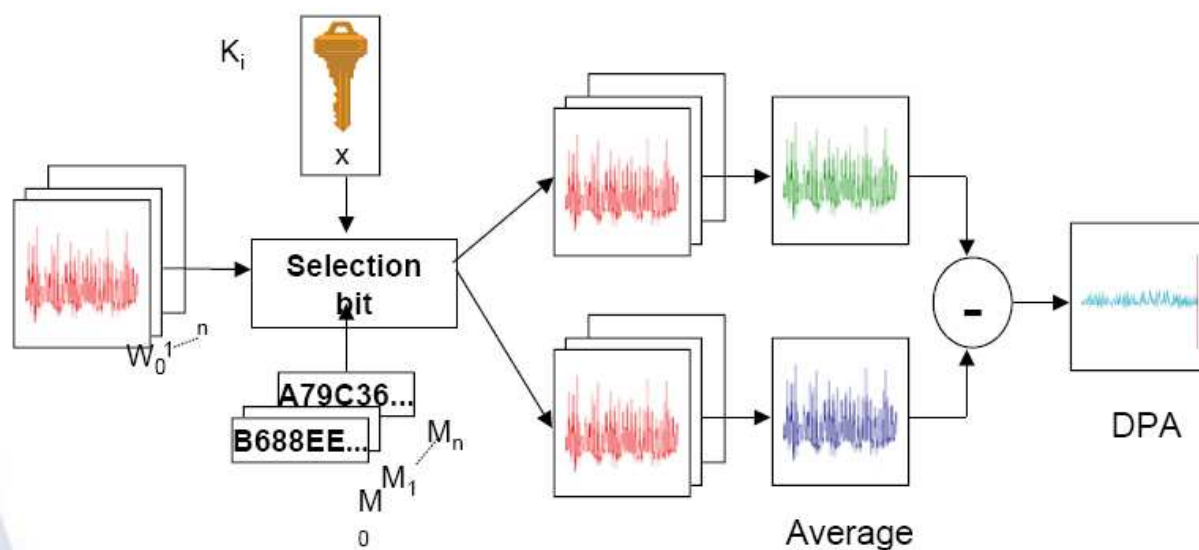
- After data collection, what is available ?
 - N plain and/or cipher random texts

00	B688EE57BB63E03E
01	185D04D77509F36F
02	C031A0392DC881E6 ...
 - N corresponding power consumption waveforms



217

Hypothèse et test



218

La HODPA (High Order DPA)

La HODPA ou DPA d'ordre n est aussi basée sur une étude statistique de la consommation de courant de la carte.

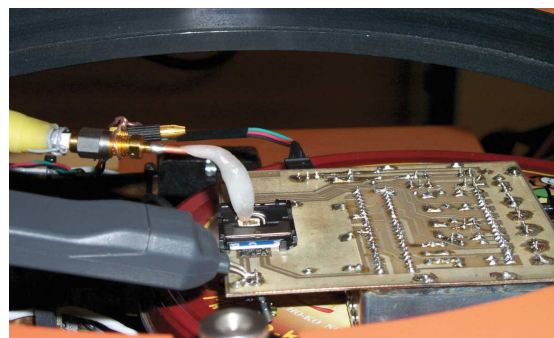
Différence : elle utilise des corrélations entre la consommation de courant et n variables intermédiaires ne dépendant que d'un fragment de la clé et du message d'entrée.

Elle est beaucoup plus difficile à réaliser, mais beaucoup plus puissante.

219

COUNTER-COUNTER MEASURE

- Against *adding noise to the power consumption signal*.
- Capture electromagnetic radiation at various chip locations!
- Equipment:



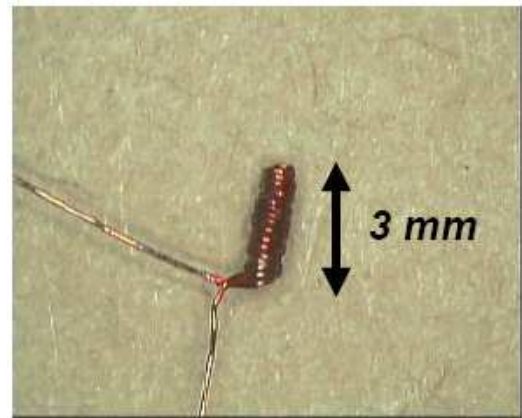
Les émissions EM

Principe : Les courants qui circulent dans la puce induisent des champs électromagnétiques qui sont susceptibles de donner le même type d'information que le courant.

Différence : L'information est plus locale. On peut déplacer la micro-sonde électromagnétique au dessus de la zone qui nous donnera le plus d'informations (exemple : co-processeur cryptographique).

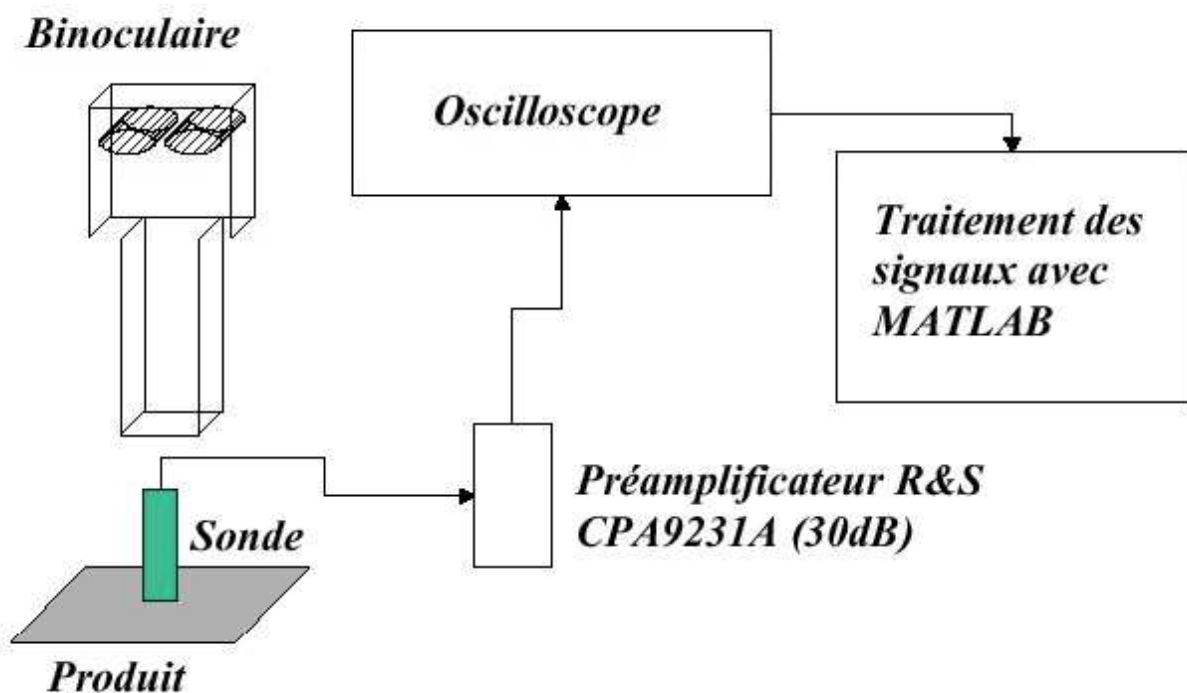
Avantage : Insensible aux contre-mesures physiques tels que l'ajout de bruit en sortie ou le lissage de la consommation globale de courant.

Inconvénient : La reproductibilité des mesures est difficile.



221

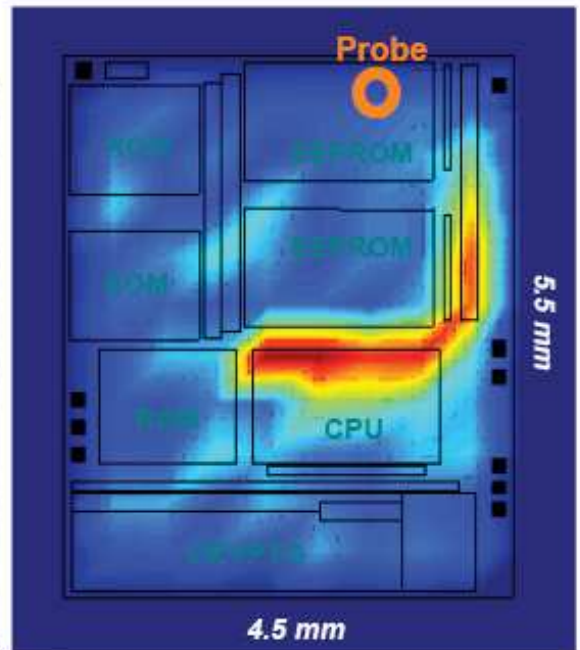
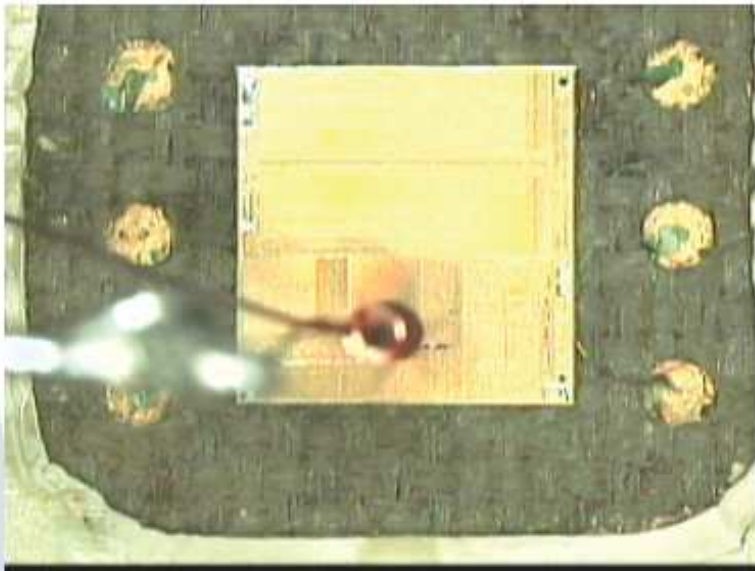
Le dispositif de cartographie EM



222

Le dispositif de cartographie EM

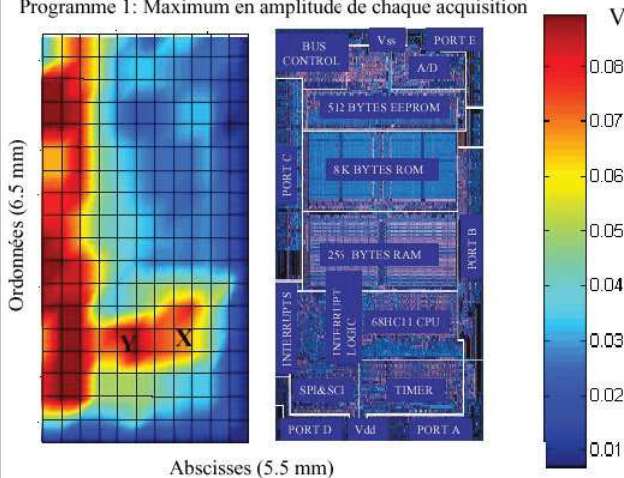
- Horizontal cartography (XY plane)
 - to pinpoint instruction related areas
 - better if automated



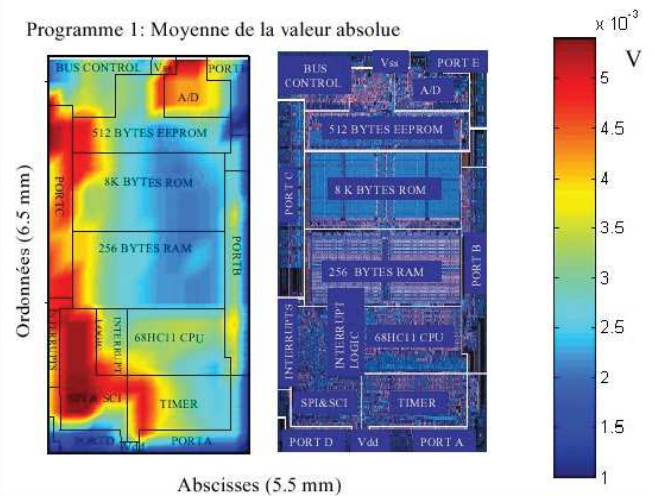
220

Cartographie pour l'algo 1

Programme 1: Maximum en amplitude de chaque acquisition

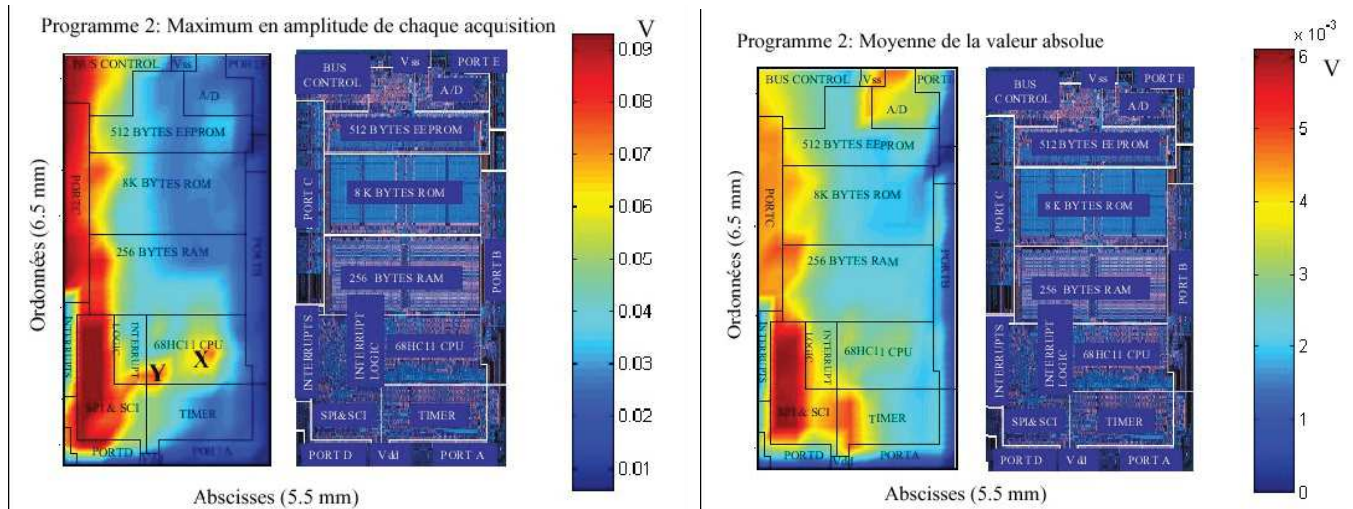


Programme 1: Moyenne de la valeur absolue



224

Cartographie pour l'algo 2

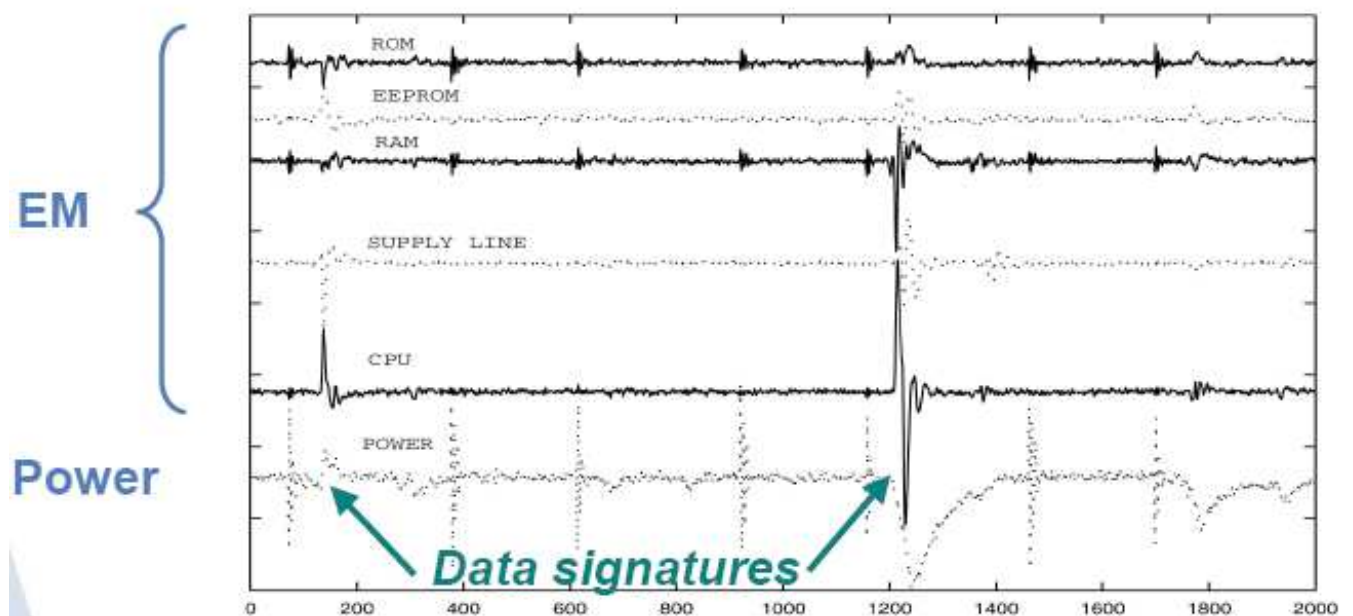


225

Différence EM/Power

- EM signals versus XY probe position

Differential traces between (00h $\oplus$ 00h) and (FFh $\oplus$ 00h) picked up at different locations



226

Attaques EM

la SEMA (Simple EM Analysis)

la DEMA (Differential EM Analysis)

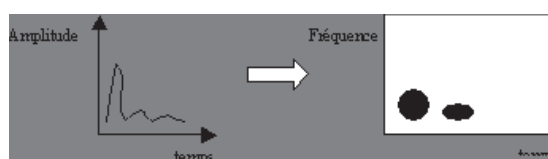
=> nécessite moins d'acquisitions

227

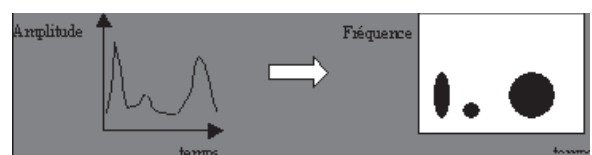
La rétro-conception logicielle

méthode du « dictionnaire » :

Une instruction :



Une autre instruction :



- La séquence des deux :



Utilisation possible de la consommation en courant ou des émissions EM.

Statut : recherches en cours.

228

Invasives

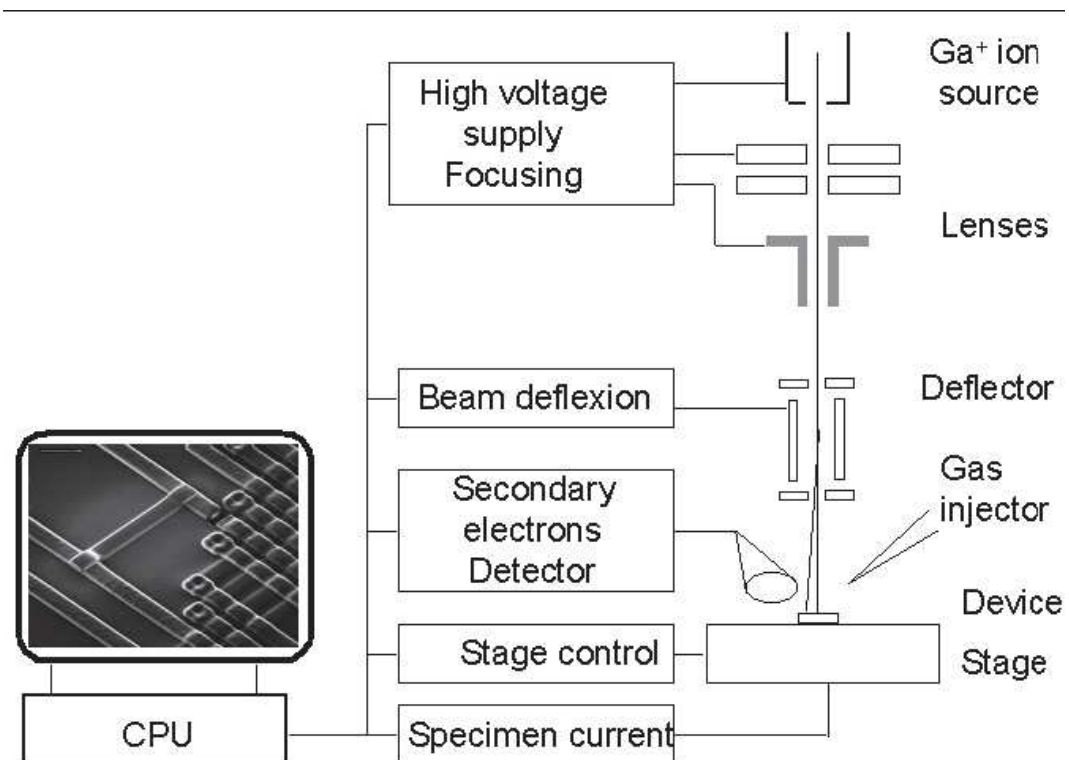
micro-probing

modification de circuit (FIB)

réetro-conception du circuit

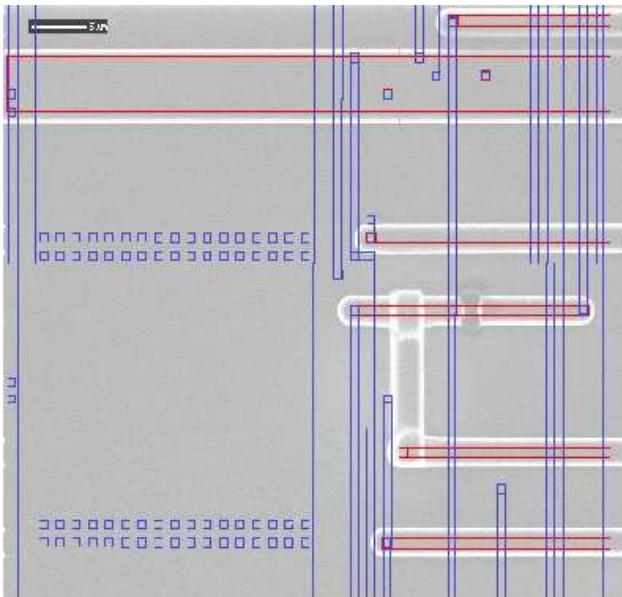
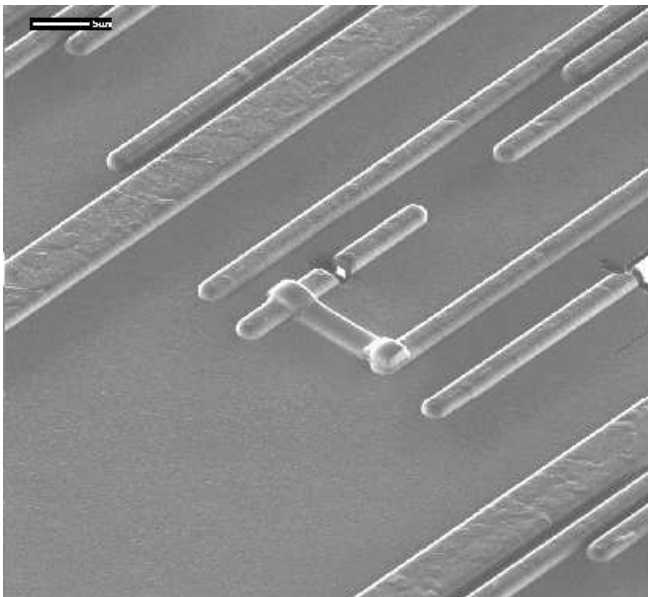
229

Le FIB (Focused Ion Beam)

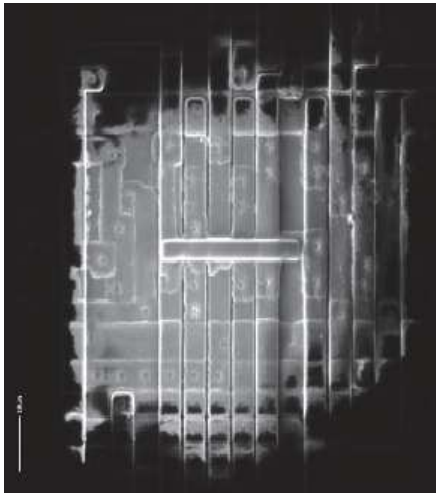
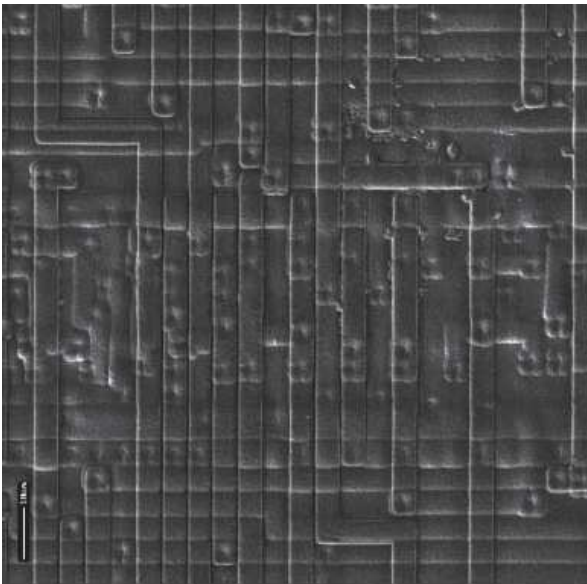


230

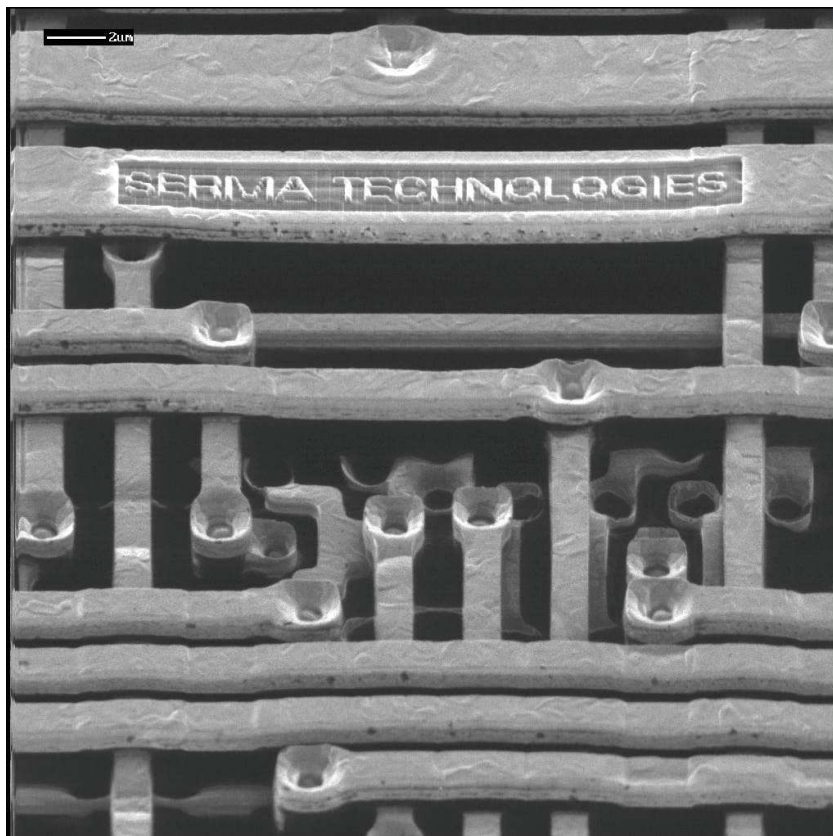
Modifications de circuits (1/2)



Chip Re-Wiring / Addition of a Track

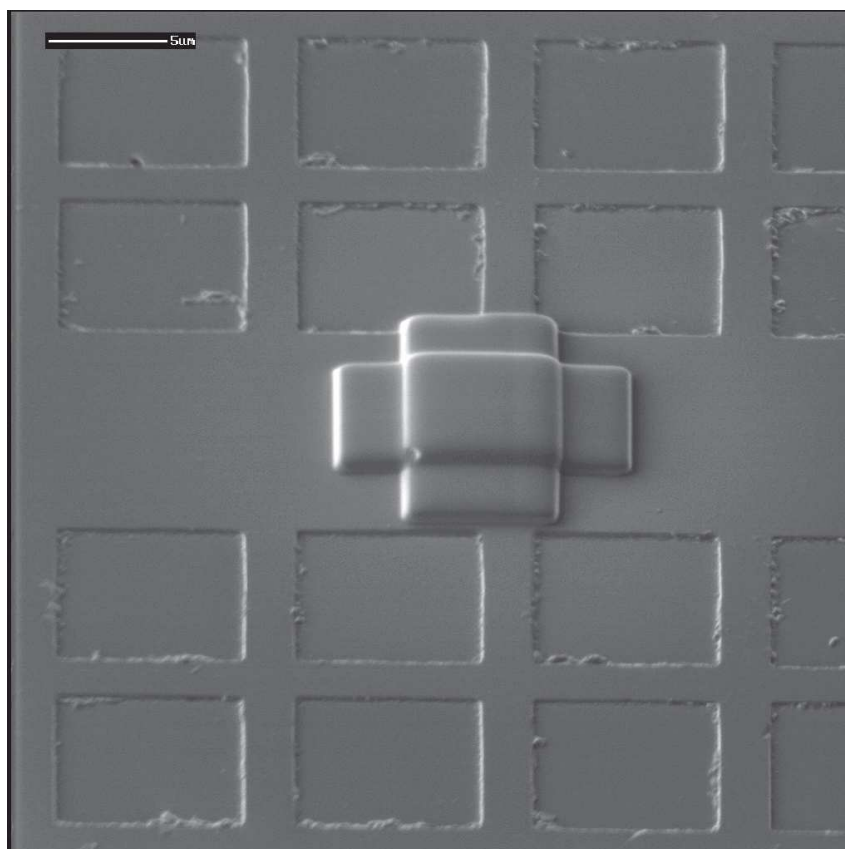
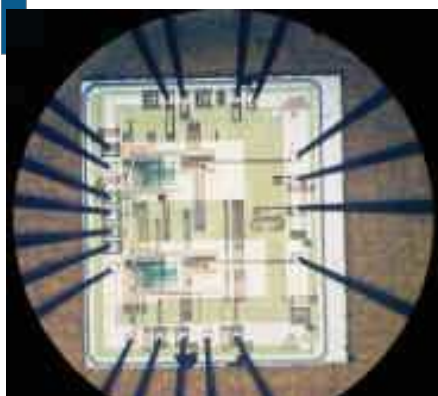


Modifications de circuits (2/2)



233

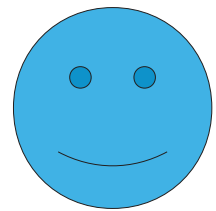
Micro-probing



234

Très cher !

- If you have more money or if you are a student.



235

Côtation d'une attaque

Niveau d'expertise requise

Expert, ..., homme de la rue

Connaissance du produit

Information publique, diffusion restreinte, sensible, critique

Accès au produit

Nombre d'échantillon nécessaire : <10 , <100, <1000, non praticable

Temps d'identification

<1 heure, 1 jour, 1 semaine, 1 mois, ...

Temps d'exploitation

<1 heure, 1 jour, 1 semaine, 1 mois, ...

Equipement

Aucun, standard, spécialisé, ...

...

=> La sécurité et le business s'opposent. Il faut donc trouver un juste compromis !

236



Le piratage facile en vidéos

How to Reverse-Engineer a Satellite TV Smart Card.mp4

Mifare Hack.mp4

237

Quelques applications

La carte SIM/USIM

Les réseaux cellulaires

Il y a eu le GSM/1

Dans les années 80, plusieurs réseaux cellulaires ont vu le jour en Europe

Les systèmes sont incompatibles d'un pays à un autre

Conséquences :

- équipements mobiles limités aux frontières du pays
- marché limité

Création du « Groupe Spécial Mobile » pour :

- Améliorer la qualité de la transmission
- support international : roaming
- rajout de nouvelles fonctionnalités
- offrir des terminaux et des services à coûts accessibles

Il y a eu le GSM/2

➤ Normalisation 1982 : Baptisé « Groupe Spécial Mobile »

➤ Depuis 1989, l'ETSI (European Telecommunications Standard Institute) édite les spécifications du GSM et de l'UMTS (*Universal Mobile Telecommunications System*, réseau de 3^{ème} génération).

Siège de l'ETSI à Sophia Antipolis.

➤ 1991 : devenu une norme internationale nommée « Global System for Mobile communications »

En Europe, le standard GSM utilise les bandes de fréquences 900 MHz et 1800 MHz.
Aux États-Unis, la bande de fréquence utilisée est la bande 1900 MHz.

Tri-bande : les téléphones portables pouvant fonctionner en Europe et aux États-Unis

Bi-bande : les téléphones fonctionnant uniquement en Europe.

➤ La norme GSM autorise un débit maximal de 9,6 kbps

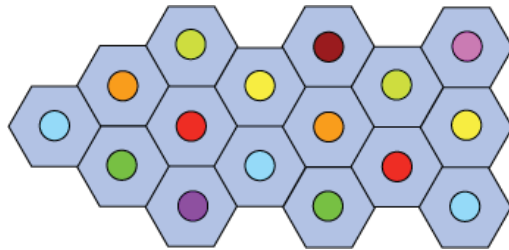
=> transmission de la voix, des données numériques de faible volume, des messages textes (**SMS**, pour *Short Message Service*) ou des messages multimédias (**MMS**, pour *Multimedia Message Service*).

Notion de réseau cellulaire

Un réseau de téléphonie mobile est basé sur la notion de **cellules**,

Une cellule : est une zone circulaire qui couvre une zone géographique.

Une cellule : centaine de mètres (zone urbaine), une trentaine de kms (zone rurale).



Chaque cellule dispose d'un émetteur-récepteur central appelé « **station de base** » (en anglais *Base Transceiver Station*, **BTS**).

Plus le rayon d'une cellule est petit, plus la bande passante disponible est élevée.

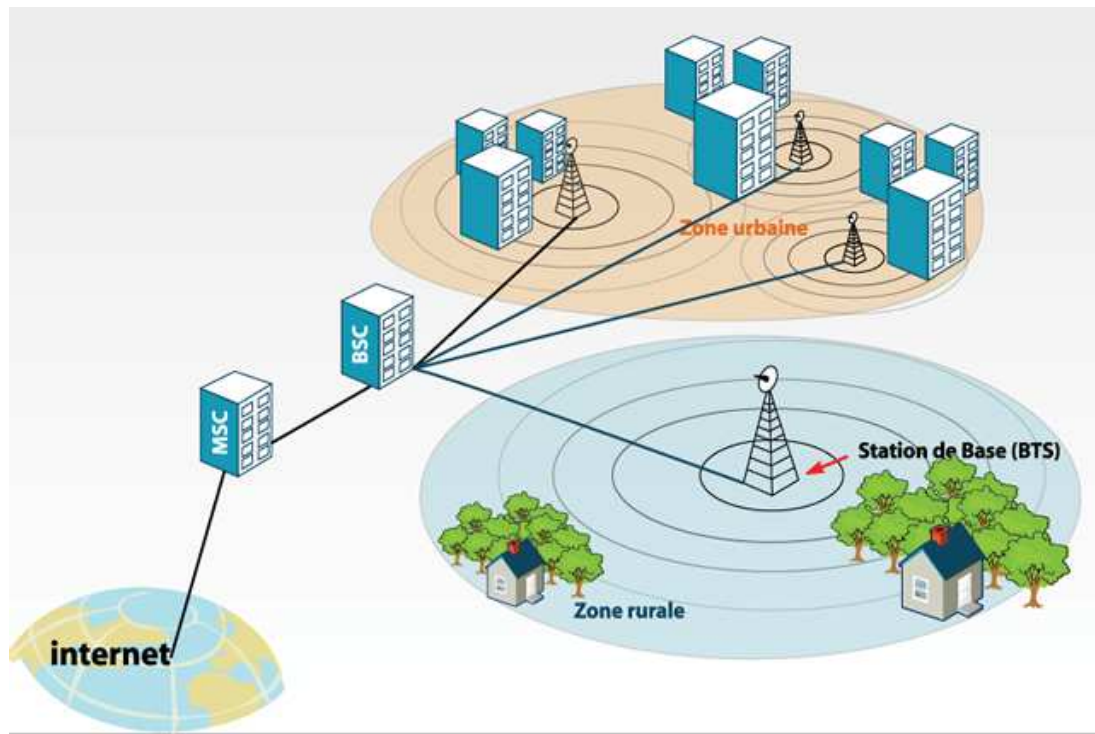
Chaque cellule est entourée de 6 cellules voisines.

Les cellules adjacentes **ne peuvent pas** utiliser la même fréquence.

Éléments du réseau cellulaire GSM

- **Un contrôleur de stations** (**BSC**, *Base Station Controller*)
qui relie toutes les stations de base, chargé de gérer la répartition des ressources.
- **Sous-système radio** (en anglais **BSS** pour *Base Station Subsystem*) =
contrôleur de stations + les stations de base.
- **Centre de commutation du service mobile** (**MSC**, *Mobile Switching Center*),
géré par l'opérateur téléphonique, relie les contrôleurs de stations
au réseau téléphonique public et à internet.
- **Sous-système réseau** (**NSS**, *Network Station Subsystem*) auquel appartient le MSC,
chargé de gérer les identités des utilisateurs, leur localisation et l'établissement
de la communication avec les autres abonnés.

Architecture du réseau GSM



Bases de données manipulées

- **Le registre des abonnés locaux (HLR, Home Location Register)**: base de données contenant des informations (position géographique, informations administratives, etc.) sur les abonnés inscrits dans la zone du commutateur (MSC).
- **Le registre des abonnés visiteurs (VLR, Visitor Location Register)**: base de données contenant des informations sur les autres utilisateurs que les abonnés locaux. Le VLR rapatrie les données sur un nouvel utilisateur à partir du HLR correspondant à sa zone d'abonnement. Les données sont conservées pendant tout le temps de sa présence dans la zone et sont supprimées lorsqu'il la quitte ou après une longue période d'inactivité (terminal éteint).
- **Le registre des terminaux (EIR, Equipment Identity Register)** : base de données répertoriant les terminaux mobiles.
- **Le centre d'authentification (AuC, Authentication Center)** : élément chargé de vérifier l'identité des utilisateurs.

Mobilité

- Le réseau cellulaire supporte la mobilité grâce à la gestion du *handover*, c-à-d le passage d'une cellule à une autre.
- Les réseaux GSM supportent aussi la notion d'**itinérance** (*roaming*), c-à-d le passage du réseau d'un opérateur à un autre.

Les stations mobiles

Station mobile



➤ **Station mobile** : terminal de l'utilisateur

➤ **Station mobile** composée de :

- Une carte **SIM** (*Subscriber Identity Module*), pour identifier l'utilisateur de façon unique.
- Un équipement mobile identifié par un numéro d'identification unique de 15 chiffres appelé **IMEI** (*International Mobile Equipment Identity*).

➤ Chaque carte SIM possède un numéro d'identification unique (et secret) : **IMSI** (*International Mobile Subscriber Identity*), qui peut être protégé à l'aide d'une clé de 4 chiffres appelés *code PIN*.

➤ La communication entre une station mobile et la station de base se fait par l'intermédiaire d'un lien radio, généralement appelé **interface air**.

Carte SIM

➤ **Notion introduite en 1988**

➤ **Plus de 5 milliards de cartes SIM fabriquées en 2015**

➤ **Rôle fonctionnel dans le réseau :**

- Contient les détails concernant l'abonnement de l'utilisateur de téléphone mobile
- Détient les secrets nécessaires pour prouver l'authenticité du mobile et pour chiffrer les échanges
- Chargement de nouveaux services

Carte SIM : Mobilité

➤ Détails d'abonnement mémorisés sur la carte :

- Identité unique de l'abonné (IMSI)
- Numéro de téléphone de l'abonné (MSISDN)
- Code de service (opérateur)
- etc.

Carte SIM : Services sécuritaires

➤ La carte SIM stocke des informations sensibles :

- Codes secrets :
 - Authentification de l'utilisateur : Code PIN (Personal Identification Code)
 - Authentification de l'opérateur : Code PUK (Personal Unlock Code)
- Clés secrètes :
 - Pour l'authentification de la carte SIM par le réseau : Ki
 - Pour la communication chiffrée : Kc

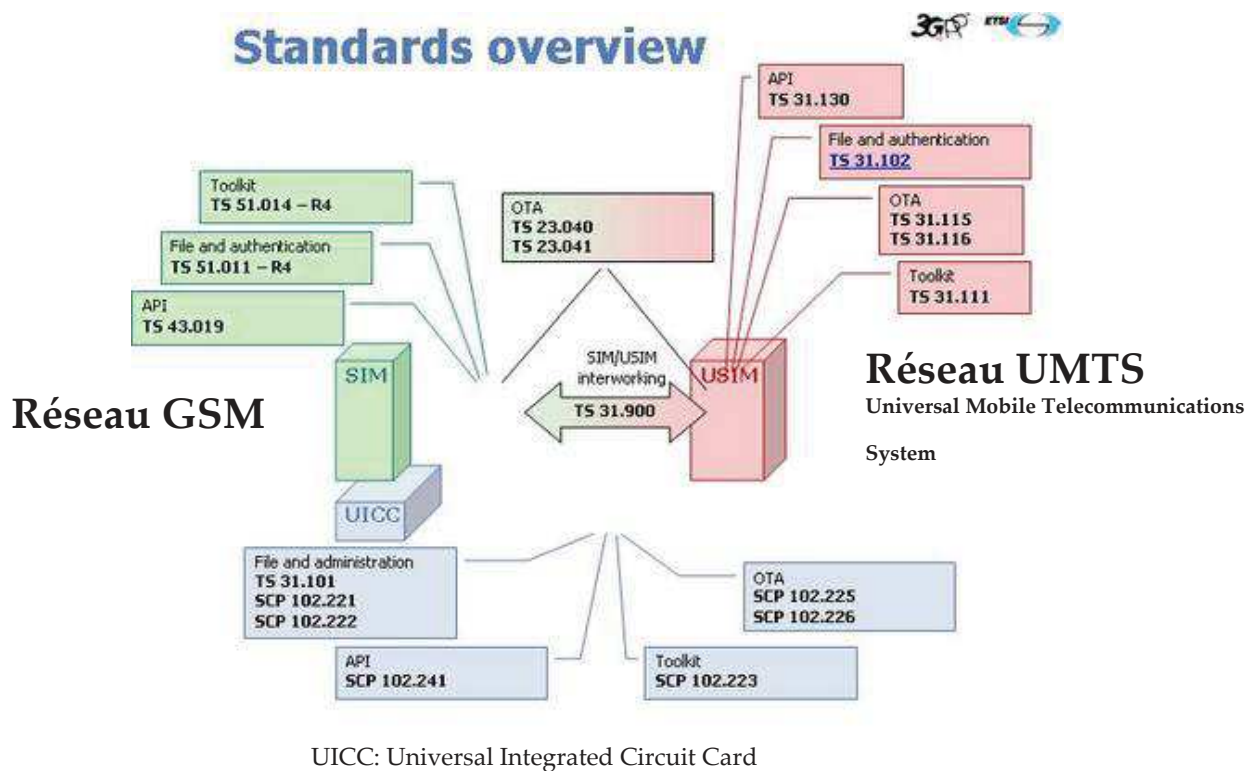
Carte SIM : Services téléchargeables

➤ La carte SIM est un environnement d'exécution pour les applications de confiance

- Capables d'interagir avec le mobile
 - Affichage d'infos sur l'écran du mobile
 - Récupérer les infos de l'utilisateur
 - etc.
- Capables d'interagir avec le réseau :
 - envoyer et recevoir des messages (SMS, GPRS, etc.)
 - géolocalisation
- Capables d'interagir avec le système fichiers de la carte SIM
 - écrire/lire des fichiers de la SIM

La normalisation et la sécurité

Les standards



Les standards ETSI

SIM

Gestion des Fichiers et Authentification : 3 GPP TS 51.011 (ETSI GSM 11.11)

SIM Toolkit Applet Management : 3 GPP TS 51.014 (ETSI GSM 11.14)

SIM API for Java Card : 3 GPP TS 43.019

USIM

Gestion des Fichiers et Authentification : 3 GPP TS 31.102

USIM Toolkit Applet Management : 3 GPP TS 31.111

USIM API for Java Card : 3 GPP TS 31.130

Méthodes de protection proposées dans GSM 02.09/1

1. La protection de l'identité d'un abonné :

L'abonné possède un identifiant (IMSI : *International Mobile Subscriber Identity*) permettant de retrouver les paramètres d'abonnement dans le HLR (Host Location Register) : base de données des comptes client. Le réseau délivre un TMSI (*Temporary Mobile Subscriber Identity*) une identité temporaire qui change à chaque appel pour interdire la traçabilité des communications.

2. L'authentification d'un abonné :

Une authentification forte est réalisée à l'aide de l'algorithme A3 associé à une clé Ki de 128 bits.

GSM 02.09: "Digital cellular telecommunications system (Phase 2+); Security Aspects".

Méthodes de protection proposées dans GSM 02.09/2

3. La confidentialité des données utilisateur :

Dans un réseau cellulaire radio, l'information est transmise par des ondes électromagnétiques (Over The Air) entre le téléphone mobile et la station de base. Les échanges entre mobile et station de base sont chiffrés à l'aide de l'algorithme A5 qui utilise une clé de chiffrement Kc. Kc est mise à jour à chaque appel (authentification) avec l'algorithme A8 de génération de clés. A3 et A8 sont souvent confondus (nommés A38 ou A3A8).

4. La protection de certaines informations telles que :

IMSI, numéros appelés ou appelants, le numéro de série du téléphone (IMEI : *International Mobile Equipment Identity*).

Infrastructures d'authentification du GSM

Il existe cinq entités :

La carte SIM

Le mobile

VLR (*Visitor Location Register*) : entité associée à plusieurs entités de base

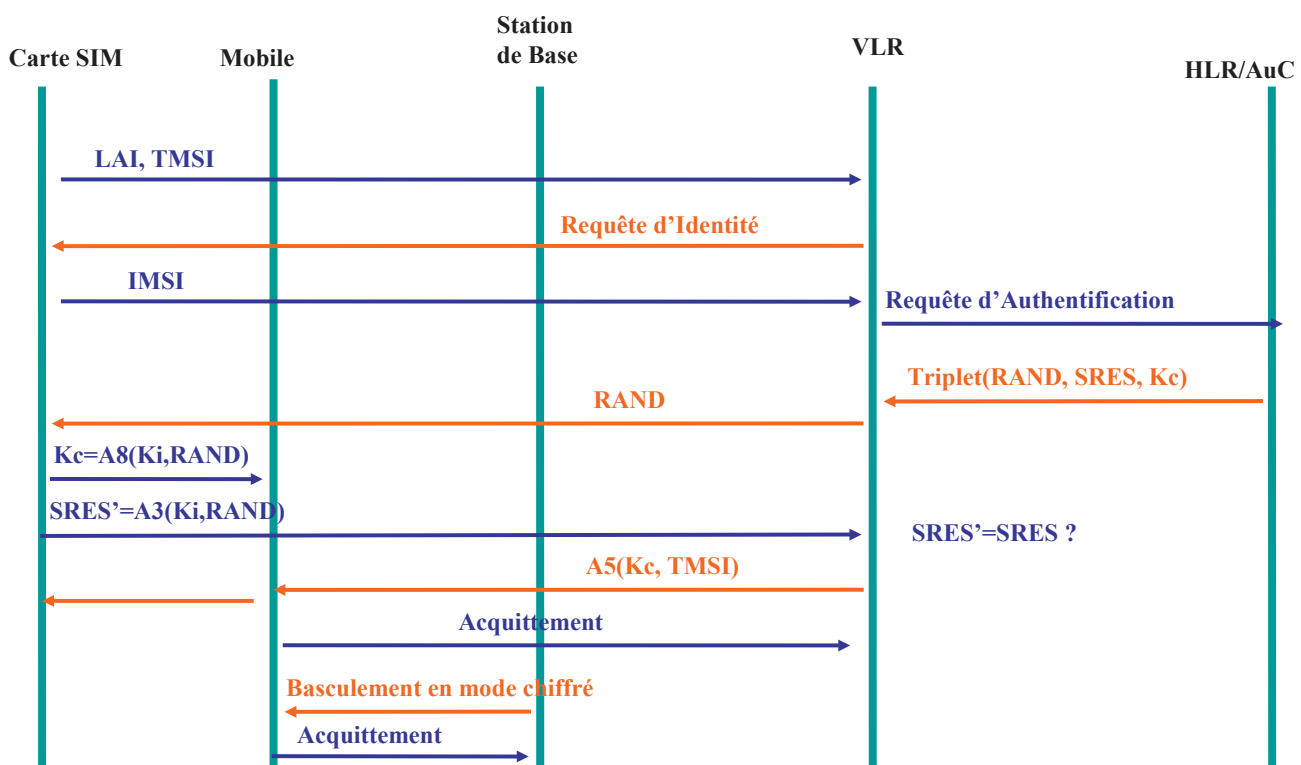
HLR (*Host Location Register*) : base de données clients

Le centre d'authentification (AuC, *Authentication Center*).

La norme 3GPP TS 43.020 identifie une cellule ou un ensemble de cellules à l'aide de l'étiquette LAI (*Location Area Identity*).

3GPP TS 43.020 – Technical Specification Group Services & System Aspects; Security Related Network Functions (Release 5, 2002).

Principes de sécurité d'un réseau GSM/1



RAND : nb aléatoire de 16 octets

SRES (Signed RESponse) : réponse signée $SRES=A3(Ki, RAND)$

Kc : clé de chiffrement des communications, $Kc=A8(Ki, RAND)$.

Principes de sécurité d'un réseau GSM/2

1. L'abonné dispose des valeurs (LAI, TMSI) stockées dans le module SIM, suite à un appel précédent.
2. Le mobile transmet au VLR les valeurs (LAI, TMSI).
3. Si le VLR échoue pour retrouver l'IMSI, il envoie une requête d'identification au mobile
4. Le VLR récupère l'IMSI mémorisé dans la carte SIM
5. Le VLR envoie au HLR/AuC une demande d'authentification
6. AuC produit un triplet GSM (RAND, SRES, Kc)
7. À la réception du triplet, le VLR transmet au mobile RAND
8. La carte SIM calcule $SRES' = A_3(K_i, RAND)$ qui est envoyé au HLR.
9. Le HLR vérifie l'égalité entre SRES et $SRES' \Rightarrow$ authentification de l'abonné en cas de succès.
10. Le VLR choisit un nouveau TMSI, le chiffre avec l'algorithme A5 et la clé Kc et l'envoie au mobile qui le déchiffre.

Les opérations de chiffrement et de déchiffrement appliqués aux signaux radio sont réalisées par le mobile (et non la carte SIM). Au-delà des stations de base, dans le réseau câblé de l'opérateur, il n'y a aucune garantie de confidentialité.

Algorithmes cryptographiques

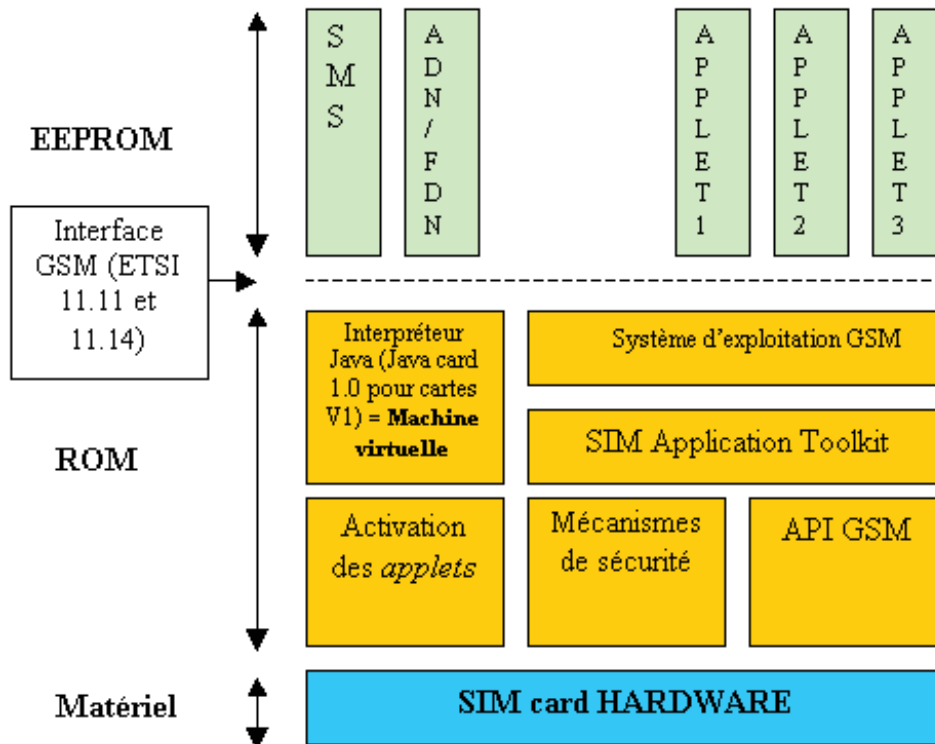
- La carte SIM réalise le calcul A3A8 dans un espace sûr.
- En 1998, Mark Briceno, Ian Goldberg et David Wagner (chercheurs à l'université de Berkeley) ont cassé l'algorithme A3A8.
- Même si GSM ne recommande aucun algorithme, les opérateurs utilisent la procédure secrète COMP128-1.
- Ces chercheurs ont aussi cassé cet algorithme en retrouvant la clé Ki en 219 calculs (environ 500 000 essais). Pour cette raison, les composants qui intègrent COMP128-1 sont munis d'un compteur limitant le nombre d'appels à 100 000.
- Les modules SIM sont aujourd'hui basés sur l'algorithme COMP128-2 dont l'algorithme est pour le moment secret.

Le système de fichiers

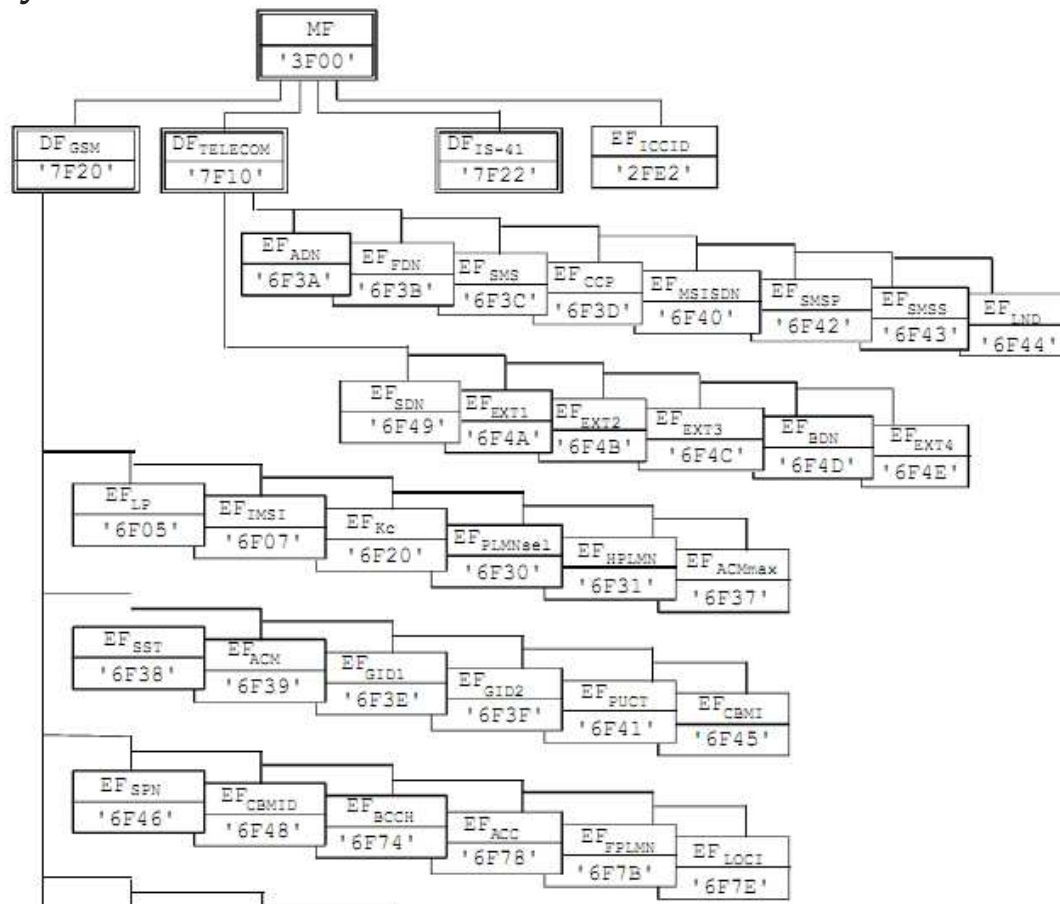
Caractéristiques physiques d'une carte SIM

- Début des années 90:
Une carte SIM : un CPU (8 bits), RAM (128 octets), ROM (7 ko), EEPROM (3 ko).
- Actuellement :
Une carte SIM : un CPU (32 bits), RAM (16 ko), ROM (512 ko), EEPROM/FLASH (512 ko), processeur dédié au calcul cryptographique.
- La ROM (Read Only Memory) contient le système d'exploitation de la carte, les mécanismes de sécurité (algorithmes spécifiques (API GSM)).
- l'EEPROM (Electrically Erasable Programmable Read Only Memory) contient des répertoires définis par la norme GSM (tels que les numéros de téléphones l'abonné...) et des données liées aux applets (service de messages courts et applications spécifiques).
- la RAM (Random Access Memory) permet d'effectuer des calculs ou de charger des instructions et les exécuter.

Structure d'une carte SIM



Système de Fichiers selon la norme 3GPP TS 51.011



La commande SELECT

A0 A4 00 00 02 XX XX (XX XX : FID du fichier/répertoire à sélectionner).

La sélection d'un répertoire entraîne une réponse qui peut inclure des informations telles que :

- la taille mémoire non utilisée
 - le nom du répertoire sélectionné
 - le type du répertoire (MF ou non)
 - présentation du code PIN
 - nombre de sous répertoires
- nécessité éventuelle de présentation du code PIN, avec le nombre d'essais possibles.

Lectures de Fichiers

➤ Lecture de l'IMSI

Le fichier EF_{IMSI} (6F07) du répertoire GSM est de type transparent, il contient l'IMSI.
La sélection du fichier retourne la taille du fichier.

A0 B0 00 00 09 (READ BINARY 9 octets, taille de l'IMSI).

➤ Lecture de TMSI et LAI

Ces paramètres sont lus à partir du fichier EF_{LOCI} (6F7E)

A0 B0 00 00 0B (READ BINARY 11 octets, 4 octets pour TMSI suivis de 5 octets pour LAI, ..)

Algorithme d'authentification

➤ Exécution de l'algorithme d'authentification du GSM

RUN-GSM-ALGORITHM exécute la fonction A3A8 avec comme argument le nombre aléatoire RAND de 16 octets. La commande retourne la signature SRES (4 octets) et la clé Kc (8 octets).

➤ Mise à jour du fichier EF_{Kc}

Le fichier EF_{Kc} est mis à jour par le mobile grâce à la commande UPDATE BINARY . Deux valeurs sont stockées dans le fichier : la clé et un octet de validation (=00 si clé valide et 07 sinon).

Lecture de la tables des Services

Le fichier $EF_{SIM-Service-Table}$ (6F38) contient la liste des services offerts par la SIM.

Chaque service est associé à deux bits (bit1 =1 si service présent, bit2 =1 si service actif).

Exemple :

Service n°1 permet la désactivation du code PIN de l'utilisateur,

Service n°2 signale la présence d'un annuaire de numéros abrégés (fichier EF_{ADN}),

Service n°3 notifie la présence d'un annuaire de numéros non abrégés (fichier EF_{FDN}),

Service n°4 signale la présence du fichier des SMS (fichier EF_{SMS}),

etc.

Les fichiers EF_{ADN} , EF_{FDN} , EF_{SMS} appartiennent au répertoire $DF_{TELECOM}$ (7F10).

Les fichiers Annuaire et SMS

➤ Fichier des SMS :

- noté EF_{SMS} possède 6F3C comme FID,
- un fichier cyclique,
- permet la lecture et l'écriture des SMS dans la SIM.

➤ Fichier de l'annuaire des numéros ADN

- noté EF_{ADN} avec 6F3A comme FID,
- est un annuaire téléphonique.

Cmd: A0 A4 00 00 02 6F 3A (SELECT EF-ADN)

Rép: 9F 0F (la carte souhaite envoyer 0F données)

Cmd : A0 C0 00 00 0F (GET RESPONSE 0F octets)

Rép: 00 00 1B 58 6F 3A 00 11 00 22 01 02 01 1C 90 00.

Taille du fichier : 1B 58 (7 000 octets) et taille de l'enregistrement (1C : 28 octets).
D'où le nb d'enregistrements : 7000/28=250).

Chaque numéro contient une étiquette qui s'obtient en soustrayant 14 de la taille de l'enregistrement (28-14=14). L'étiquette a son bit de poids fort à 0.

Les 14 octets servent à coder le numéro et d'autres infos. Les autres 14 sont pour le nom.

Opérations sur les codes PIN

Le code PIN tient sur 8 octets. Les octets non significatifs sont codés par FF.

VERIFY CHV : présentation de code PIN

A0 20 00 P2 08 **PIN** (P2=01 pour CHV1 : code PIN utilisateur, = 02 pour CHV2).

DISABLE PIN annule l'utilisation du code PIN.

A0 26 00 01 08 **PIN**

ENABLE PIN permet l'utilisation du code PIN

A0 28 00 01 08 **PIN**

CHANGE CHV permet de modifier le code PIN

A0 24 00 01 10 **Ancien_PIN Nouveau_PIN**

UNBLOCK CHV permet de débloquent une carte bloquée après trois essais infructueux du code PIN (CHV1).

A0 2C 00 01 10 **PUK PIN** (**PUK** est un code unique de 8 chiffres associé à la SIM).

JSR-268 Java Smart Card I/O API

- APIs for APDU-based communication with Smart Cards in Java Platform 6.0
- package javax.smartcardio
TerminalFactory, TerminalFactorySpi, CardTerminal
Card, CardChannel
CardException, CardPermission
ATR, CommandAPDU, ResponseAPDU
- Comments
JC-RMI proxy generators based on it

Example

```
// show the list of available terminals
TerminalFactory factory = TerminalFactory.getDefault();
List<CardTerminal> terminals = factory.terminals().list();
System.out.println("Terminals: " + terminals);

// get the first terminal
CardTerminal terminal = terminals.get(0);

// establish a connection with the card
Card card = terminal.connect("T=0");
System.out.println("card: " + card);

CardChannel channel = card.getBasicChannel();
ResponseAPDU r = channel.transmit(new CommandAPDU(c1));
System.out.println("response: " + toString(r.getBytes()));

// disconnect
card.disconnect(false);
```

La carte bancaire

La norme EMV : introduction

Contexte de la norme

- EMVCo manages, maintains and enhances the EMV® Integrated Circuit Card **Specifications** for chip-based payment cards and acceptance devices, including point of sale (POS) terminals and ATMs. EMVCo also establishes and administers **testing** and approval processes to evaluate **compliance** with the EMV Specifications. EMVCo is currently owned by **American Express, JCB, MasterCard and Visa**.
- A primary goal of EMVCo and the EMV Specifications is to help facilitate global **interoperability** and **compatibility** of chip-based payment cards and acceptance devices. This objective extends to new types of payment devices as well, including **contactless** payment and **mobile** payment.

Source : <http://www.emvco.com> (2009)

EMV

Standard des cartes de paiement depuis 1995

Organismes fondateurs (déc. 1993):

- **Europay** International (racheté par Mastercard en 2002) ;
- **MasterCard** International ;
- **Visa** International ;

EUROPAY
International



Rejoint par :

- le japonais **JCB International** (depuis Déc. 2004)
- l'américain **American Express** (depuis Fév. 2009)



Première version des spécifications en 1996.

En France, depuis fin 2006 les cartes bancaires et les terminaux de paiement électroniques (TPE) respectent le standard EMV.

Spécifications EMV

➤ Spécifications disponibles sur <http://www.emvco.com>

➤ Longue spécification

- de l'ordre de 867 pages

Plusieurs parties

➤ Integrated Circuit Card (ICC) Specifications for Payment Systems

➤ Version 4.1, mai 2004

➤ Book 1

- Application Independent ICC to terminal

➤ Book 2

- Security and Key Management

➤ Book 3

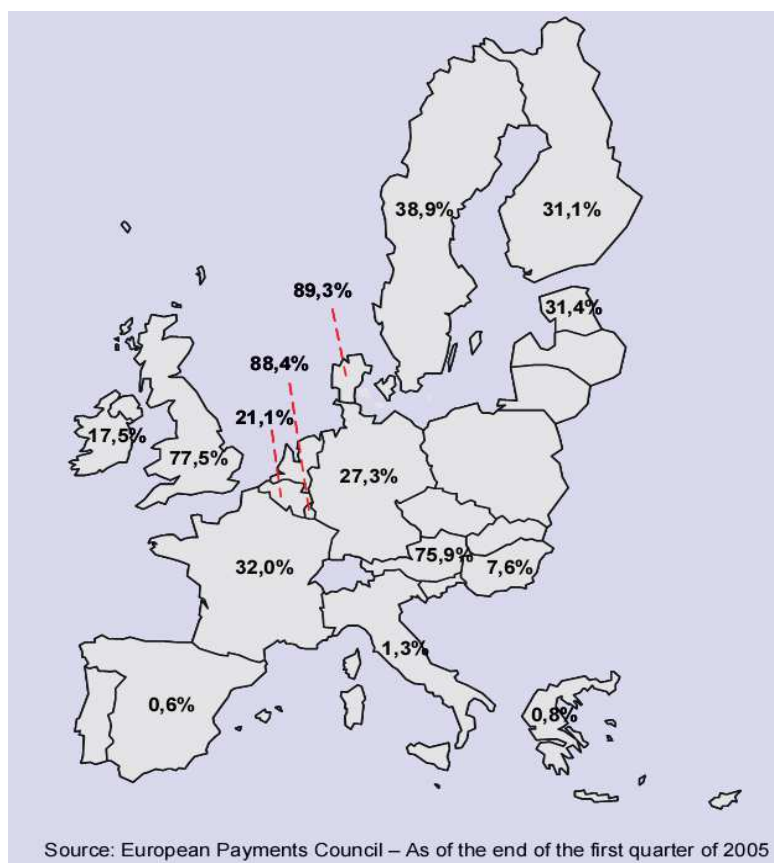
- Application Specification

➤ Book 4

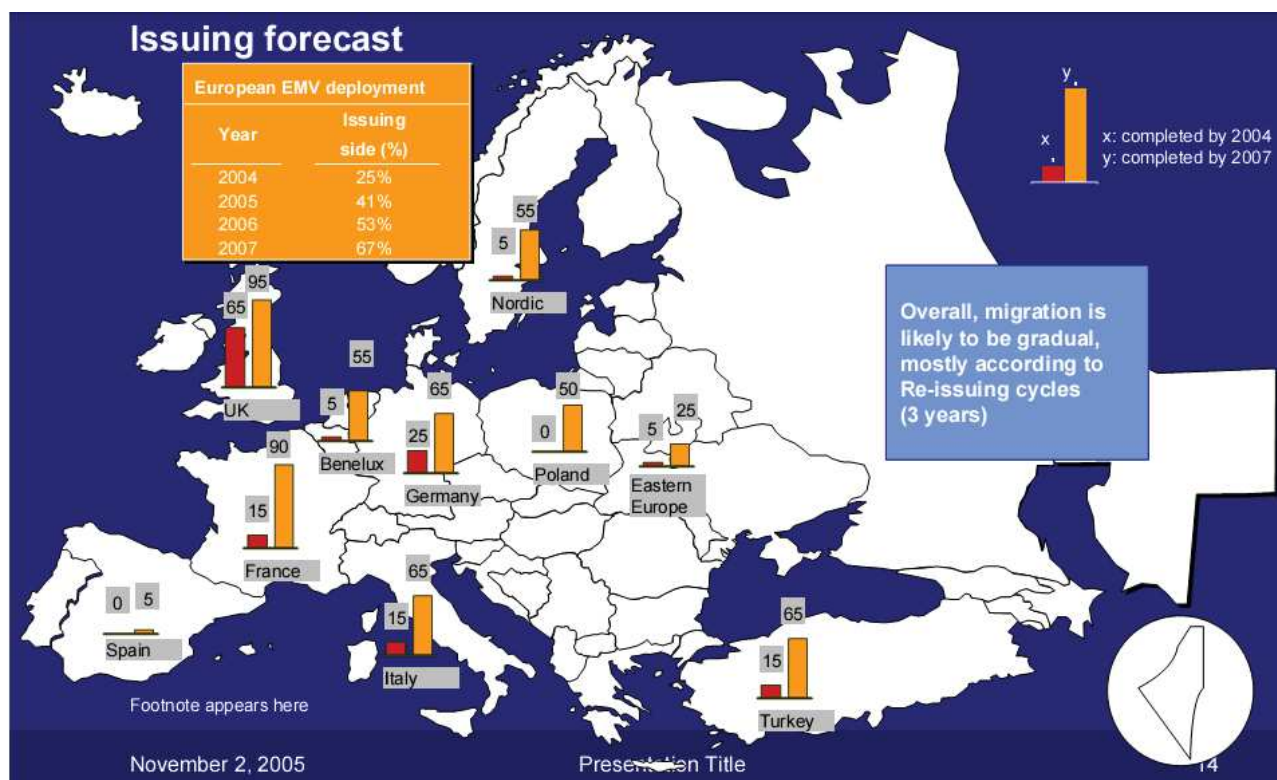
- Cardholder, Attendant, and Acquirer

Déploiement EMV

En 2005, il y avait :
176 millions cartes EMV
sur 550 millions de cartes
circulant en Europe

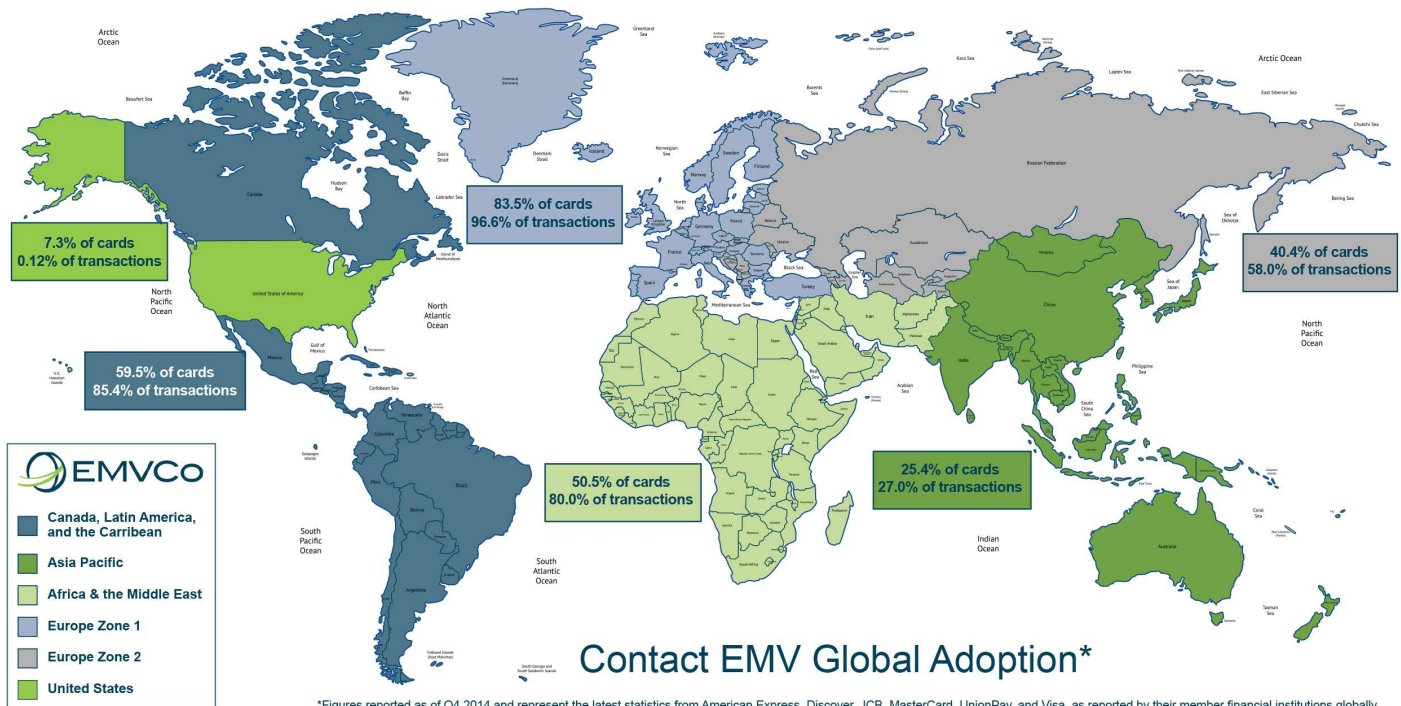


Déploiement de cartes EMV



Source : International Master Card

Déploiement de cartes EMV



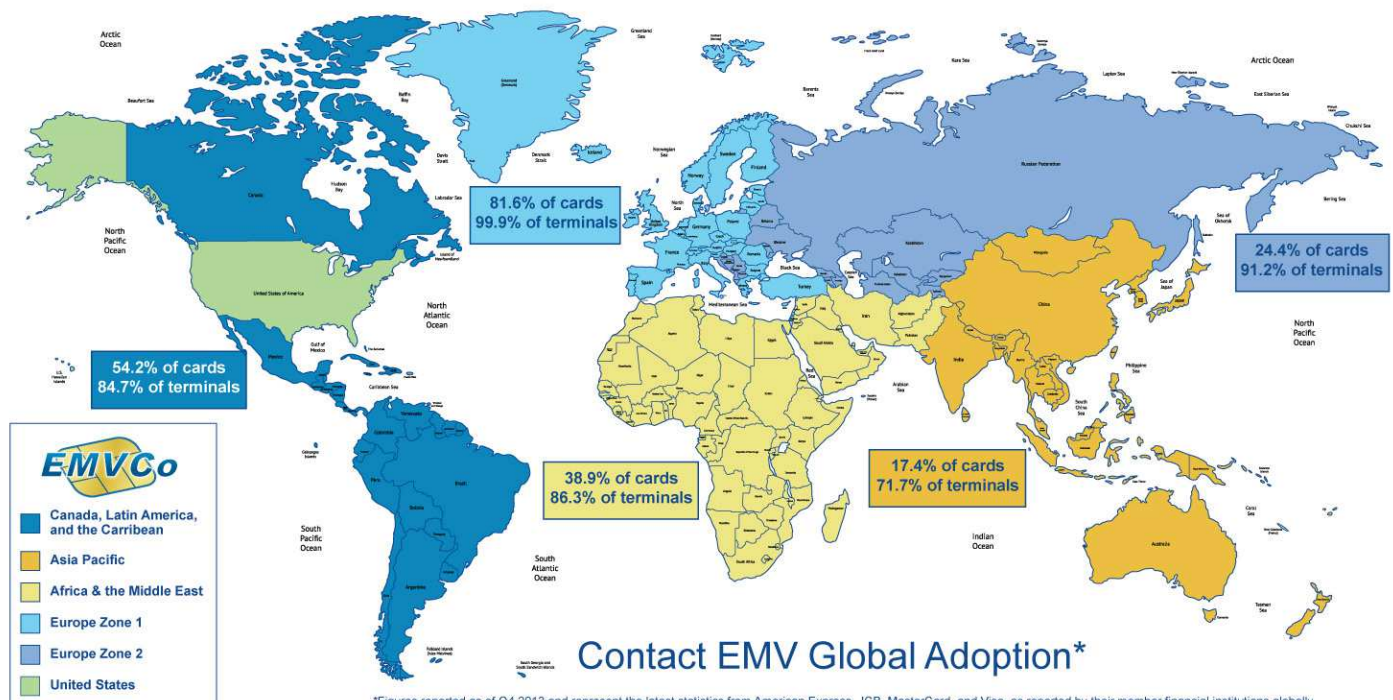
Déploiement de cartes EMV

Worldwide EMV Chip Card Deployment and Adoption*

Region	2013		2014	
	EMV Cards	Adoption Rate	EMV Cards	Adoption Rate
Canada, Latin America, and the Caribbean	471M	54.2%	544M	59.5%
Asia Pacific	942M	17.4%	1,676M	25.4%
Africa & the Middle East	77M	38.9%	116M	50.5%
Europe Zone 1	794M	81.6%	833M	83.5%
Europe Zone 2	84M	24.4%	153M	40.4%
United States	-	-	101M	7.3%

*Figures reported in Q4 2013 and Q4 2014, respectively, and represent the latest statistics from American Express, Discover, JCB, MasterCard, UnionPay, and Visa, as reported by their member institutions globally.

Déploiement de terminaux EMV



Déploiement de l'EMV aujourd'hui

- **Europe :**
 - Zone SEPA (Single Euro Payment Area)
 - Migration vers l'EMV de Janvier 2008 au 31 Décembre 2010.
 - En 2010: 100% des cartes doivent être conformes à l'EMV
 - En 2010, 590 millions de cartes bancaires EMV seront en circulation
- **Dans le monde** (Europe, Asie et Amérique Latine)
 - En 2010, 830 millions de cartes EMV en circulation.
- **USA :** pas de cartes conformes à l'EMV, idem pour les cartes e-ID.

Raisons de la migration

➤Liability shift

les coûts induits par une fraude lecteur ou terminal seront pris en charge par les banques (émetteurs de la carte) ou les commerçants dont les matériels ne seraient pas conformes à EMV

➤Fraude

différente selon les pays et les systèmes de paiement

Spécifications EMV

➤Spécifications EMV:

➤basées sur la norme ISO/IEC 7816

➤doivent être lues conjointement avec la norme ISO

➤ Si des définitions fournies dans EMV sont différentes de la norme ISO alors les définitions de la norme EMV remplacent celles de l'ISO

➤ Ces spécifications doivent être utilisées par :

➤Les fabricants de ICC et de terminaux

➤Les concepteurs de systèmes de paiement

➤Les institutions financières qui implantent des applications financières sur ICC

Book 2 : Security and Key Management

Book 2

- **Static Data Authentication**
- **Dynamic Data Authentication**
- **Cryptage du code PIN hors ligne**
- **Intégrité et confidentialité**
- **Mécanismes de sécurité : cryptage symétrique, asymétrique, signature numérique.**
- **Algorithmes cryptographiques : RSA, DES, SHA-1**

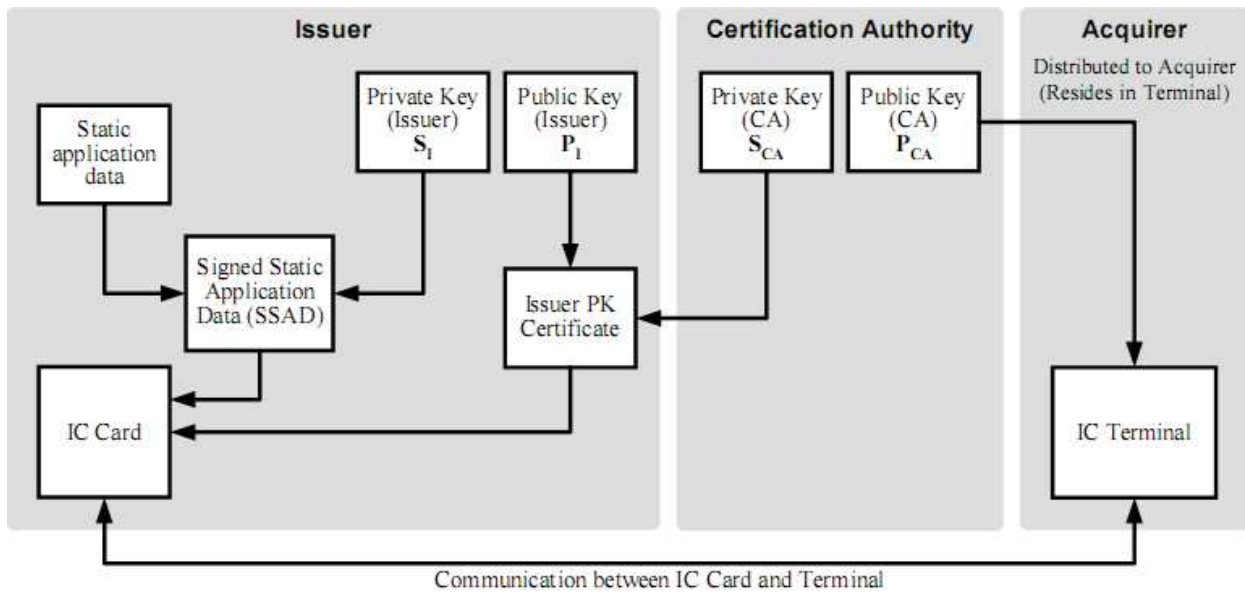
Acteurs du protocole EMV

- **La banque du client : émetteur de la carte**
- **Le client : Carte bancaire**
- **Le TPE (Terminal de Paiement Électronique ou le DAB (Distributeur Automatique de Billets) : marchand**
- **Une autorité de certification : CA**

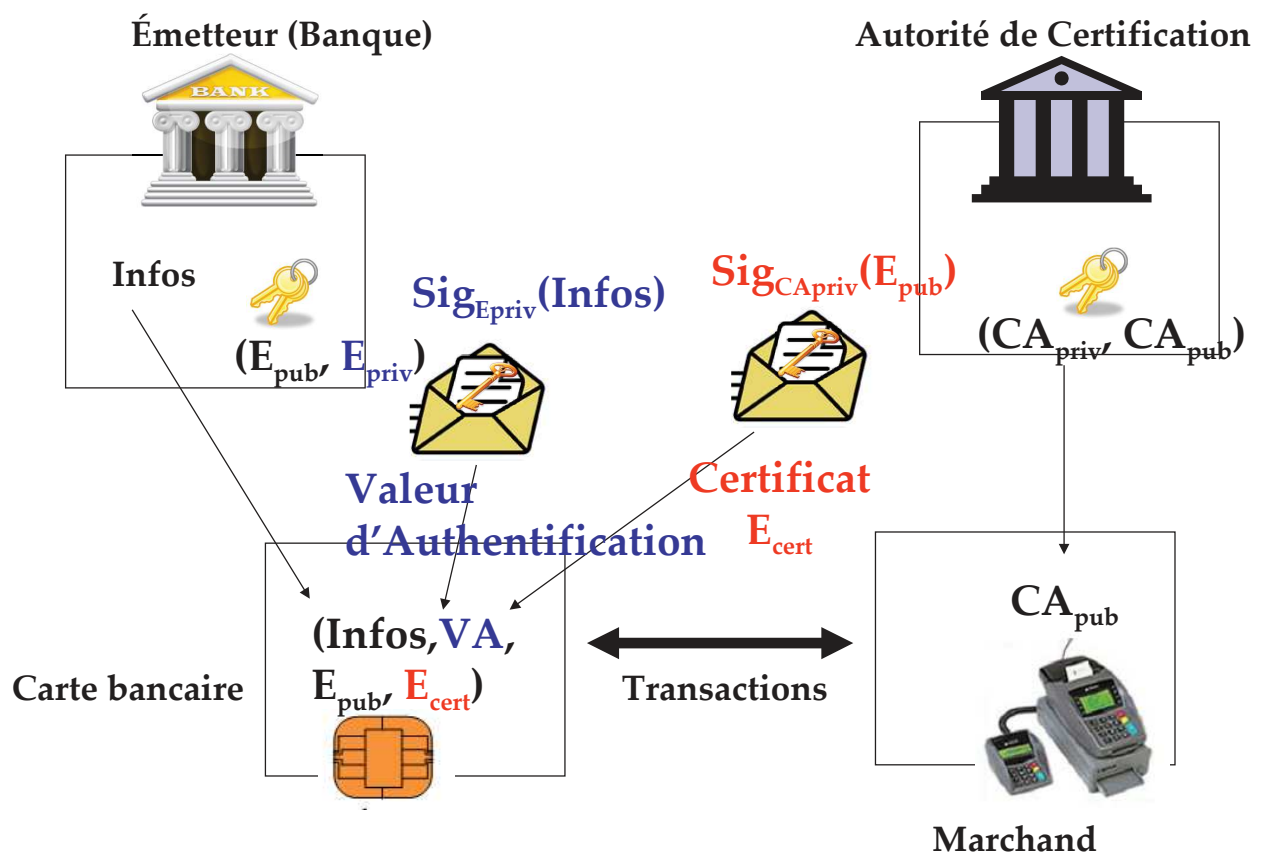
Mécanismes d'authentification

- **Processus SDA : Static Data Authentication**
consiste pour le terminal à vérifier une donnée signée mise dans la carte durant sa personnalisation
- **Processus DDA : Dynamic Data Authentication**
en plus d'une authentification statique, vérifie si la carte possède un secret délivré par l'émetteur de la carte

SDA : Static Data Authentication



Organisation des acteurs dans SDA



SDA : à la personnalisation

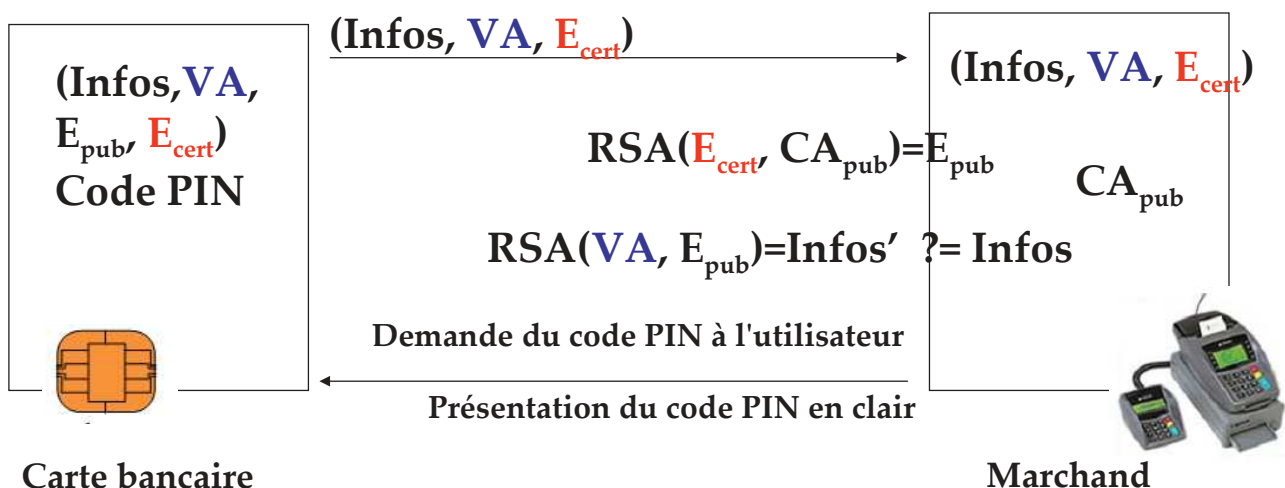
➤ Pendant la phase de personnalisation, la carte reçoit les informations suivantes:

- Le nom du porteur, le numéro de la carte ou encore la date limite de validité de celle-ci (notés *Information*).
- une valeur d'authentification (noté *VA*), signature RSA d'*Informations* générée avec la partie privée de la clé de l'émetteur
($VA = \text{Sig}_{E_{\text{priv}}}(\text{Information})$)
- le certificat de l'émetteur (E_{cert}) contenant sa clé publique signée par une autorité de certification
- le code PIN transmis au porteur de cette carte.

Lors de l'utilisation


$$VA = \text{Sig}_{E_{\text{priv}}}(\text{Infos})$$


$$E_{\text{cert}} = \text{Sig}_{CA_{\text{priv}}}(E_{\text{pub}})$$



SDA : à la l'utilisation

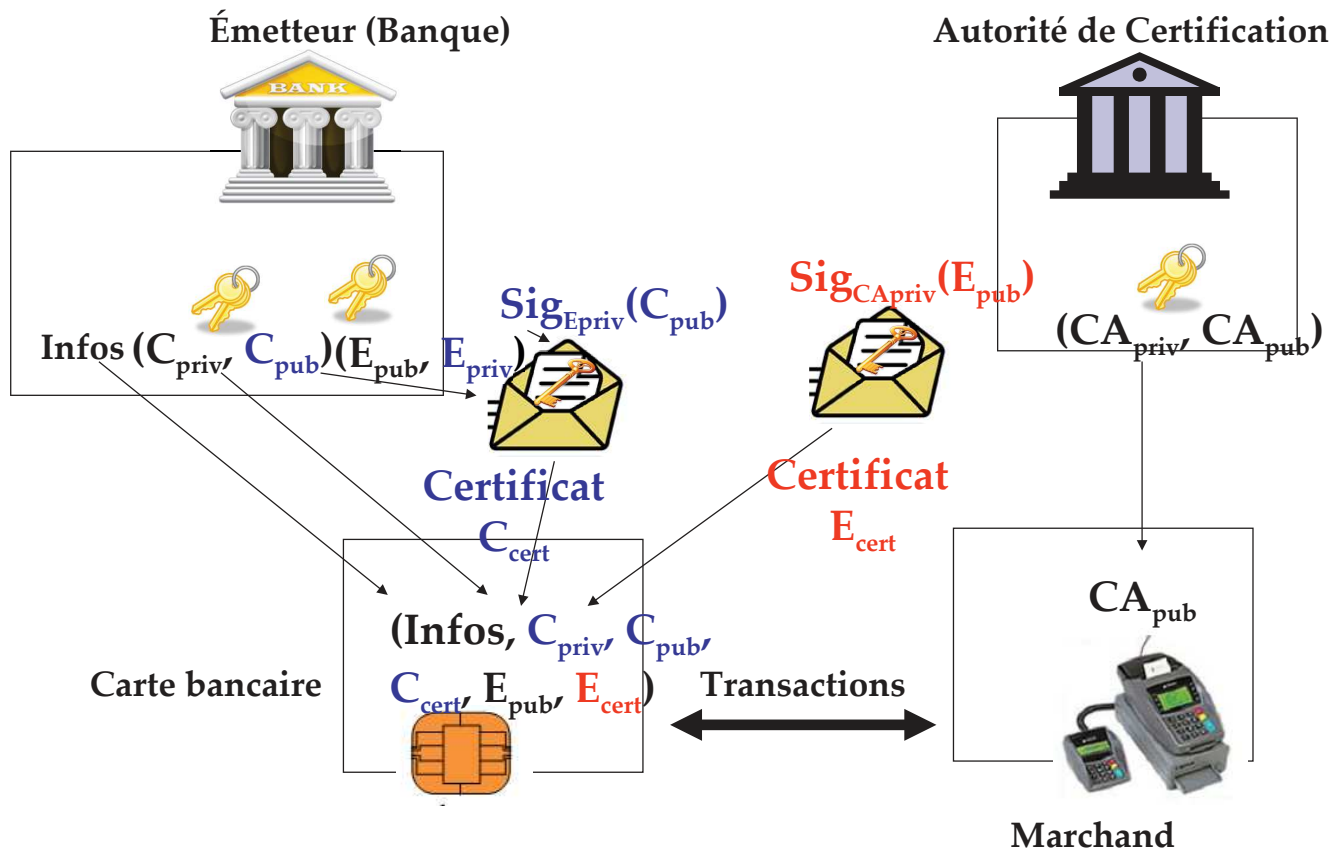
➤ Avant toute transaction :

- la carte fournit au terminal **Informations**, le certificat E_{cert} de la banque émettrice, ainsi que la valeur d'authentification **VA**
- le terminal vérifie E_{cert} avec la clé publique de l'autorité de certification (CA_{pub}) et vérifie **VA** avec la clé publique de la banque émettrice
- le terminal demande à l'utilisateur le code PIN et le transmet (**en clair**) à la carte pour qu'elle le vérifie.

La Yes card est possible

- Lors de transactions offline (par exemple le terminal n'a pas de connexion), il est possible de renvoyer les données statiques copiées d'une autre carte.
- Mais la plupart des terminaux étant en ligne de façon permanente, la fraude a diminuée car les yes cards peuvent être détectées
- Puis la DDA arrive afin de permettre à la carte de signer un aléa produit par le terminal (et donc de prouver qu'elle est une vraie carte)

Dynamic Data Authentication : DDA

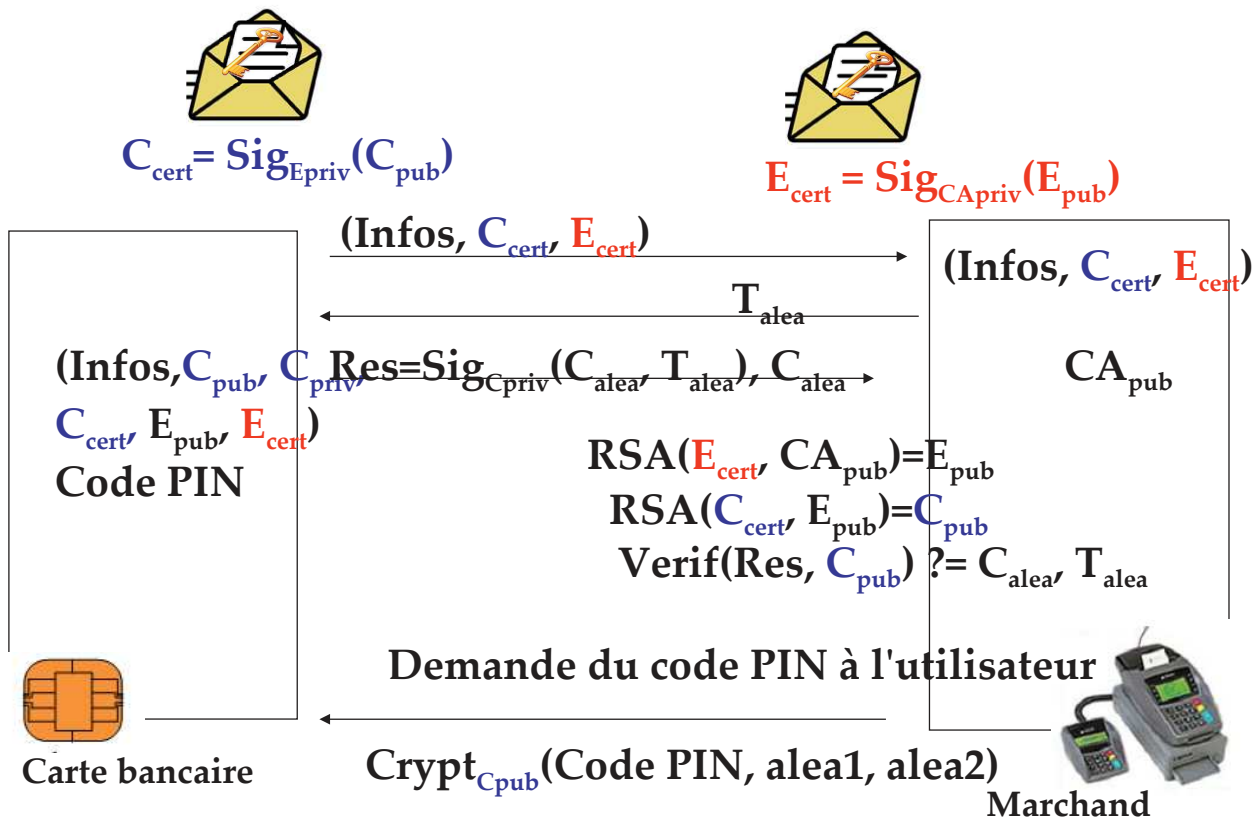


DDA : à la personnalisation

➤ Pendant la phase de personnalisation, la carte reçoit les informations suivantes:

- Le nom du porteur, le numéro de la carte ou encore la date limite de validité de celle-ci (notés **Information**).
- une paire de clés RSA (C_{pub}, C_{priv})
- un certificat (C_{cert}) contenant C_{pub} signée par l'émetteur
- le certificat de l'émetteur (E_{cert}) contenant sa clé publique E_{pub} signée par une autorité de certification
- le code PIN transmis au porteur de cette carte.

Lors de l'utilisation



DDA : à la l'utilisation

➤ Avant toute transaction :

- la carte fournit au terminal **Informations**, le certificat E_{cert} de la banque émettrice, et son certificat C_{cert}
- le terminal génère une valeur aléatoire T_{alea} et l'envoie à la carte
- la carte génère une valeur aléatoire C_{alea} . Puis, elle signe T_{alea} et C_{alea} avec sa clé privée C_{priv} . Elle envoie le résultat de la signature et C_{alea} au terminal.
- le terminal vérifie E_{cert} avec CA_{pub} et vérifie C_{cert} avec E_{pub} . Puis, il vérifie la signature des aléas avec C_{pub} .
- le terminal demande à l'utilisateur le code PIN et le transmet (**chiffré par C_{pub}**) à la carte pour qu'elle le vérifie. Le code PIN est d'abord concaténé avec deux nouvelles valeurs aléatoires fournies par la carte et le terminal, afin d'éviter les attaques par replay.

Mode de gestion des transactions

➤ La transaction est finalisée en ligne ou hors ligne, choix fait par la carte ou le terminal selon une politique de gestion de risques :

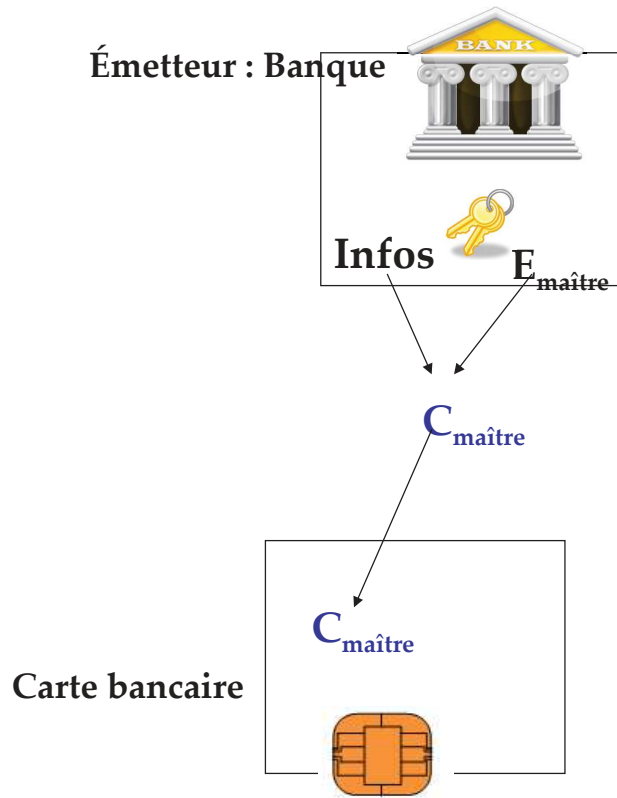
- sélection aléatoire
- validation en ligne pour n validations hors ligne
- en fonction du montant de la transaction
- en fonction du montant cumulé des transactions déjà effectuées hors ligne ou d'un plancher fixé par le marchand.

Fonctionnement en ligne et hors ligne

➤ Une clé secrète (clé 3DES de 112bits) est utilisée :

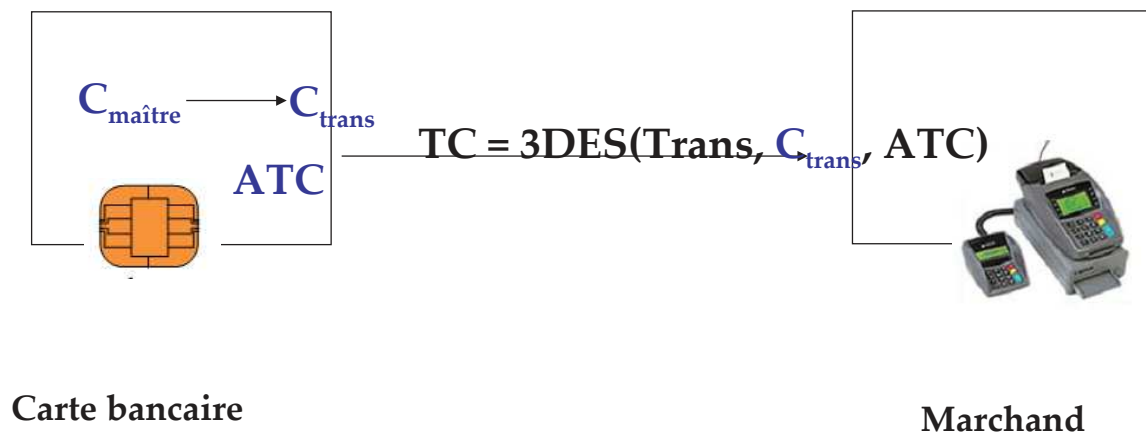
- unique par transaction
- notée C_{trans} calculée à partir de C_{maitre} et d'un compteur de transactions (ATC : Application Transaction Counter)
- C_{maitre} est une clé de la carte générée par la banque émettrice à partir d'une clé maître de la banque E_{maitre} et des informations bancaires
- C_{maitre} est mise dans la carte lors de la personnalisation
- ATC est un compteur sur deux octets géré par la carte et incrémenté à chaque transaction.

Transaction: personnalisation



Transaction: utilisation

Dérivation d'une clé unique par transaction:



Fonctionnement hors ligne

- Le terminal envoie à la carte les détails de la transaction
- La carte produit alors un **certificat de transaction TC** en signant ces données (algorithme DES CBC-MAC) à l'aide de C_{trans} .
- Le terminal ne peut pas vérifier TC mais le garde pour validation ultérieure auprès de sa banque.

Fonctionnement en ligne

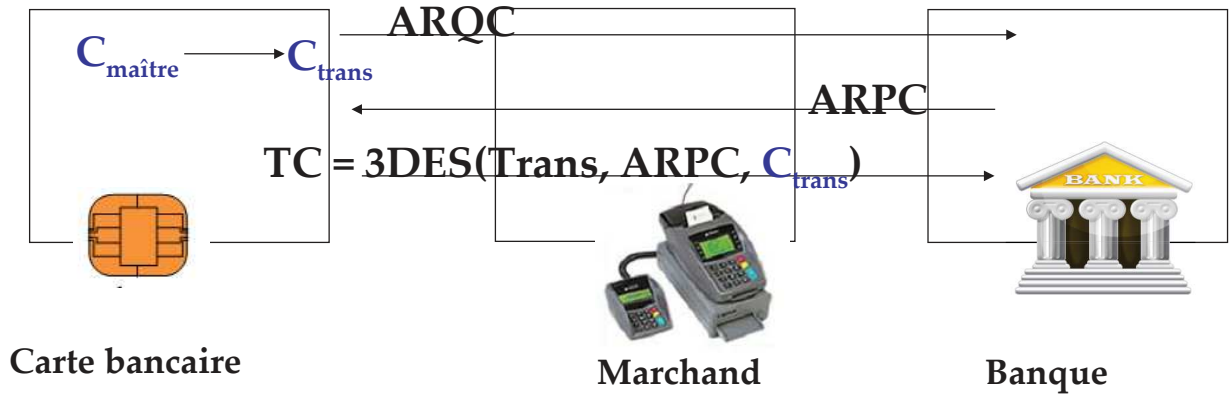
Le terminal envoie à la banque émettrice **le cryptogramme généré par la carte** (ARQC Authorization ReQuest Cryptogram).

La banque le vérifie et génère un cryptogramme réponse (ARPC Authorization ResPonse Cryptogramm) envoyé à la carte via le terminal

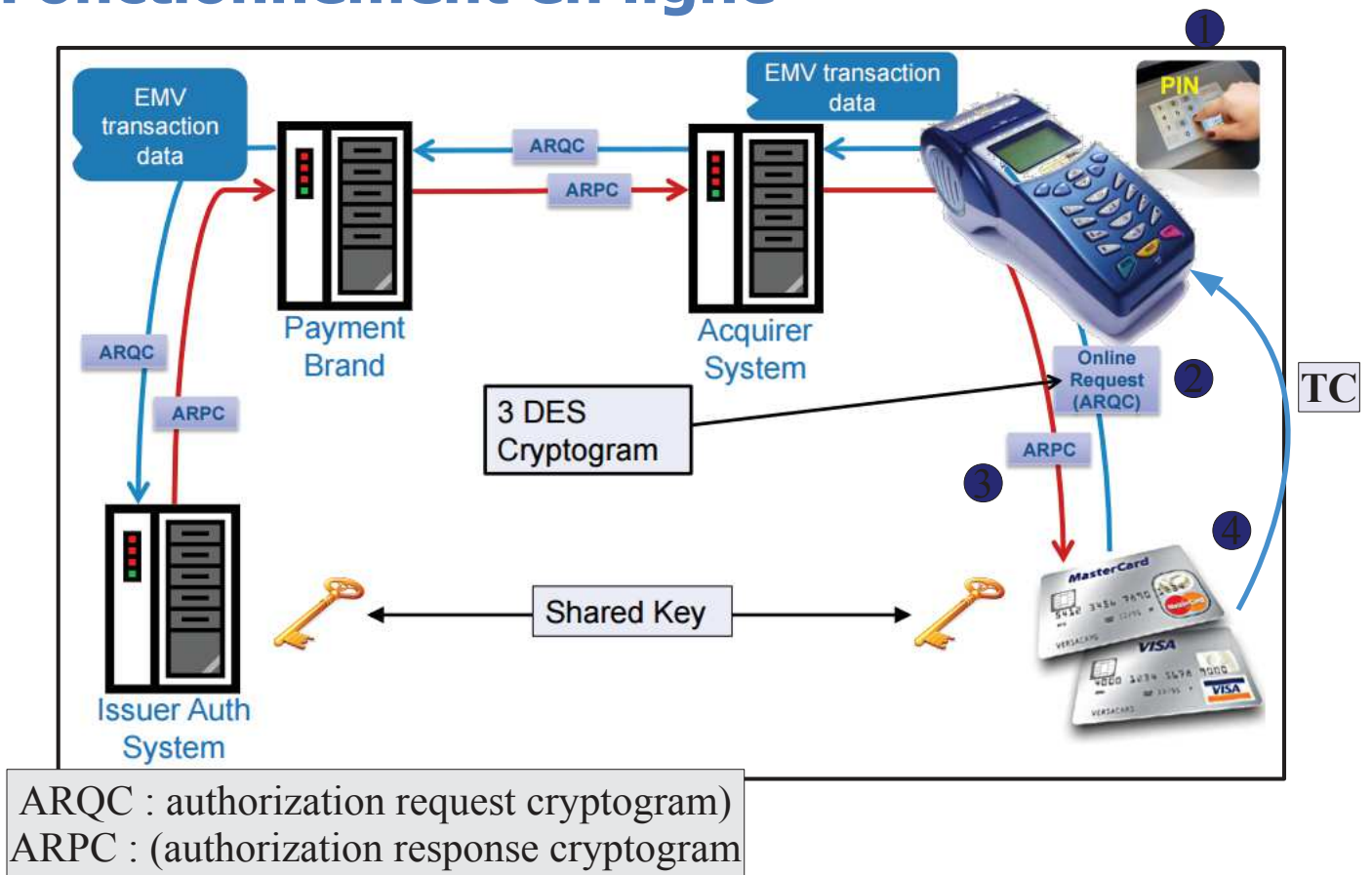
Le terminal redemande alors à la carte de lui générer un **certificat de transaction TC** qui inclut l'autorisation de la banque.

Transaction en ligne

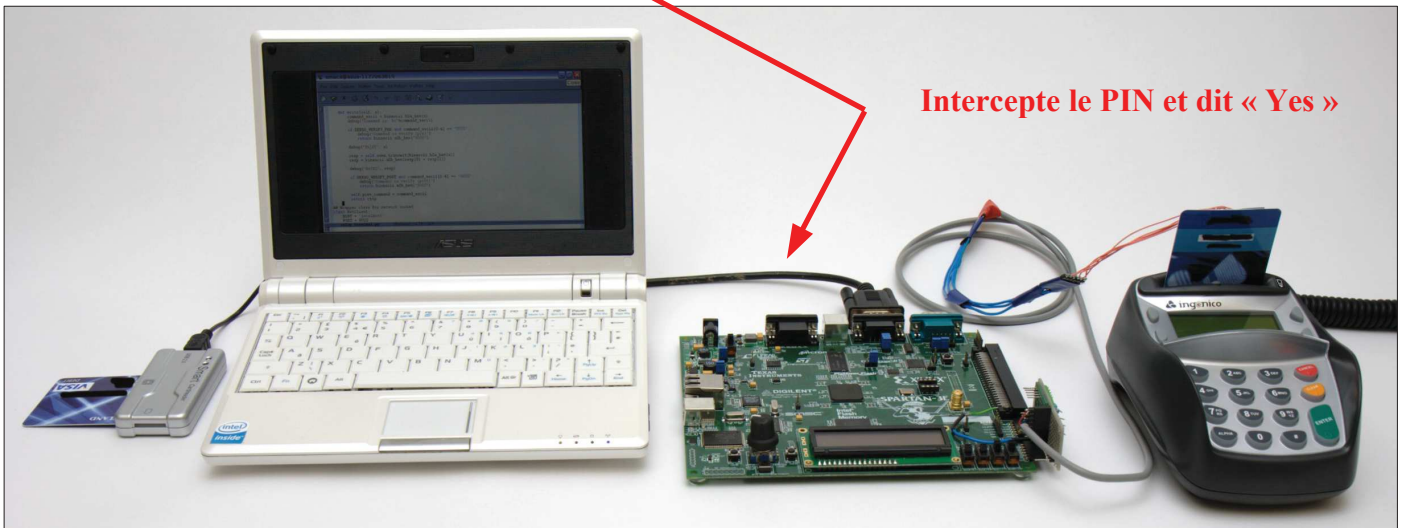
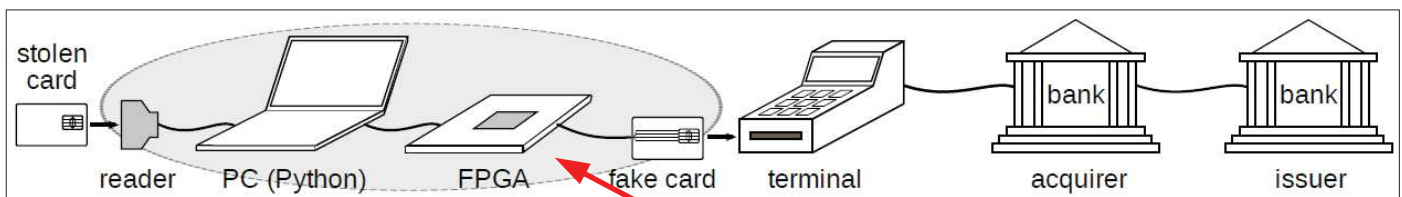
Dérivation d'une clé unique par transaction:



Fonctionnement en ligne

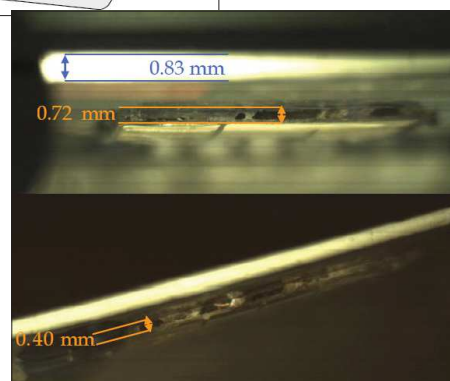
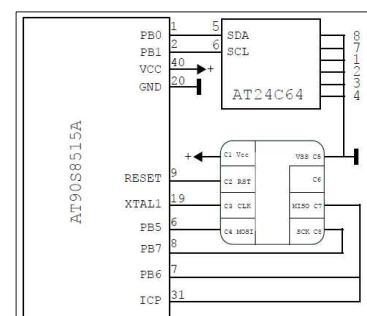
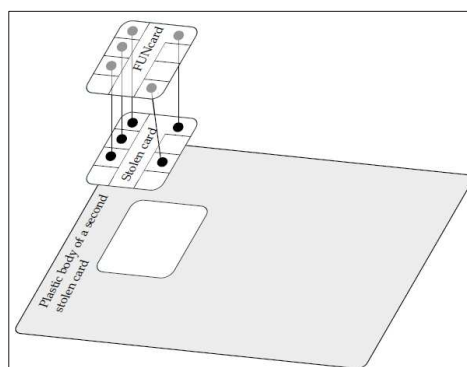


La Yes card est toujours possible



La carte doit supporter que le PIN ne soit pas présenté (terminal ne comportant pas de clavier par exemple – BUSINESS is BUSINESS) et elle donc peut envoyer un ARQC, etc.

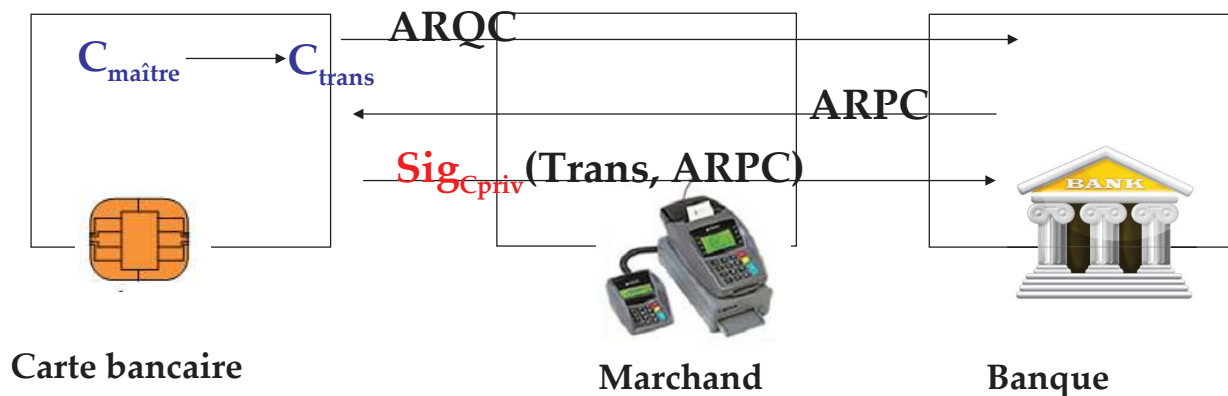
La Yes card est toujours possible



- Voir : <https://eprint.iacr.org/2015/963.pdf>
<http://sec.cs.ucl.ac.uk/users/smurdoch/papers/oakland14chipandskim.pdf>
<https://www.cl.cam.ac.uk/research/security/banking/nopin/oakland10chipbroken.pdf>

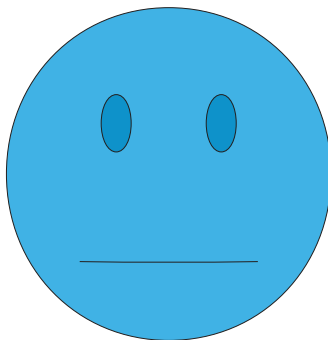
Authentication CDA

- CDA : Combined Data Authentication, variante de DDA
- utilise TC
- inclut TC dans le bloc de données signé par la carte avec sa clé privée



La Yes card est toujours possible

- Sauf si :
Le terminal vérifie la signature puis les champs devant contenir si la carte a oui ou non fait la vérification de PIN.



PSE (Payment System Environment)

➤ Un système de paiement comprend :

- **Un ensemble de fichiers dans l'ICC**
- **Des données au niveau du terminal fournies par le marchand**
- **Un protocole d'application compris par l'ICC et le terminal**
- **Les applications sont identifiées par AIDs de manière unique conformément à l'ISO7816-5**
- **Pour les cartes à contact : 1PAY.SYS.DDF01**
- **Pour les cartes sans contact : 2PAY.SYS.DDF01 (PPSE pour Proximity PSE)**

Sur Internet

➤ L'utilisation des 16 chiffres visibles de la carte n'est pas un protocole de paiement sûr.

➤ Ce numéro à 16 chiffres n'est plus imprimé sur les factures

➤ 9 chiffres sont encore inscrits sur les factures et correspondent à peu près à l'aléa choisi par la banque. => possibilité de reconstituer le numéro entier

➤ En 2001, un cryptogramme a été rajouté et imprimé uniquement sur la carte, appelé CVV (Card Verification Value) chez Visa et CVC (Card Validation Code) chez MasterCard

➤ Ce code est généré par la banque à partir des informations bancaires du client et de données secrètes de la banque

➤ Ce code ne peut être reconstruit car l'algorithme est secret

➤ Depuis 2004, le cryptogramme doit être demandé par tout site marchand en plus du numéro de la carte afin de valider toute transaction à distance auprès de la banque émettrice.

➤ Ce numéro n'est écrit nulle part sauf sur la carte.

Conclusion

- L'authentification DDA n'est pas obligatoire et
- DDA est plus robuste mais plus chère
- En France, la majorité des cartes aujourd'hui supportent DDA

- L'authentification SDA :
 - Code PIN envoyé en clair
 - Clé RSA de 1984 bits mais la donnée d'authentification peut être lue sans présentation de code

- Aujourd'hui, il manque des liens entre l'authentification, la vérification du code PIN et la génération de TC : les requêtes envoyées par le terminal peuvent être interceptées en forgeant des réponses à envoyer

- Nouvelle tendance de paiement : paiement sans contact (ISO 14443) => nouveaux systèmes, nouveaux protocoles et nouvelles failles ...

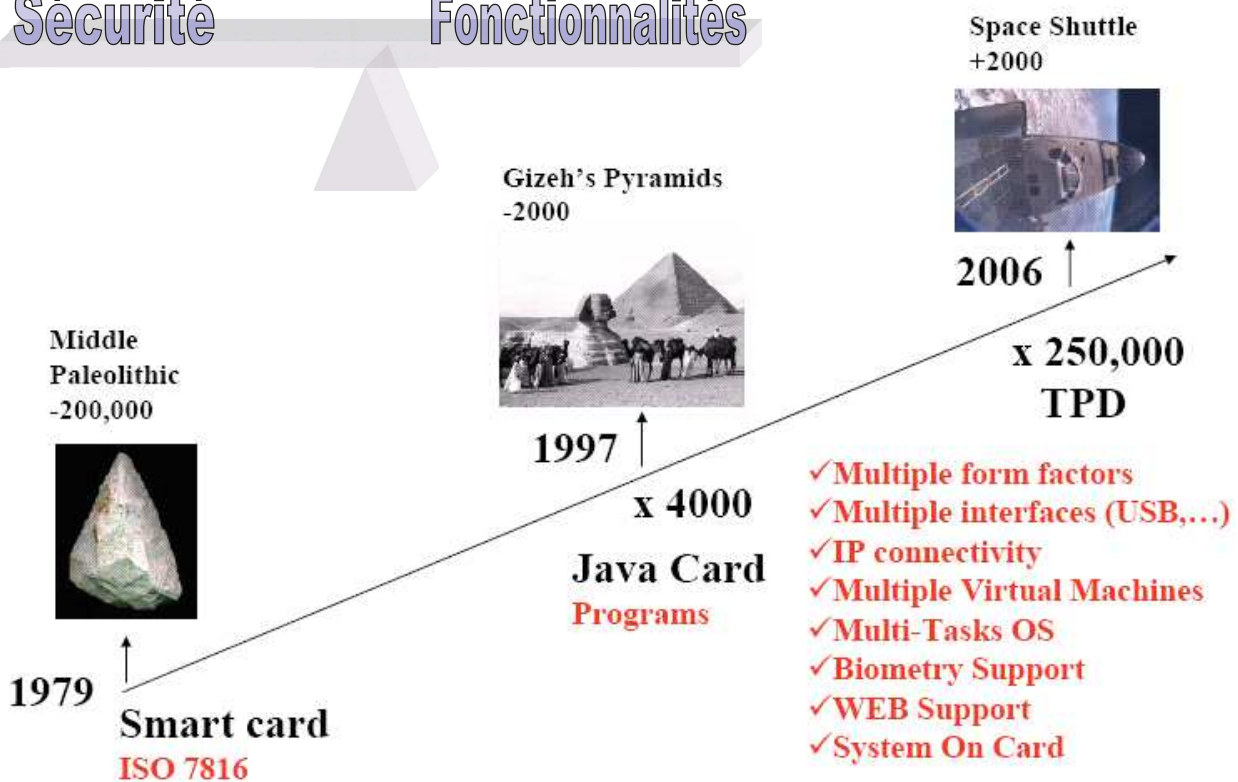


Le futur de la carte à puce

Moore's Law

Sécurité

Fonctionnalités



319

L'évolution fonctionnelle

D'après Bertrand du Castel

- 1997 JavaCard
- 1998 JavaCard+GSM SIM
- 1999 JavaCard+RSA
- 2000 WAP MicroBrowser
- 2001 Biometric (FingerPrint)
- 2002 RMI
- 2003 .NET Card
- 2004 TCP/IP Full Duplex
- 2005 Linux
- 2006 Multi-Channel (STIP)
- 2007 Streaming (full TV satellite)
- 2008 Spontaneous networking (JINI)
- 2009 GRID

320

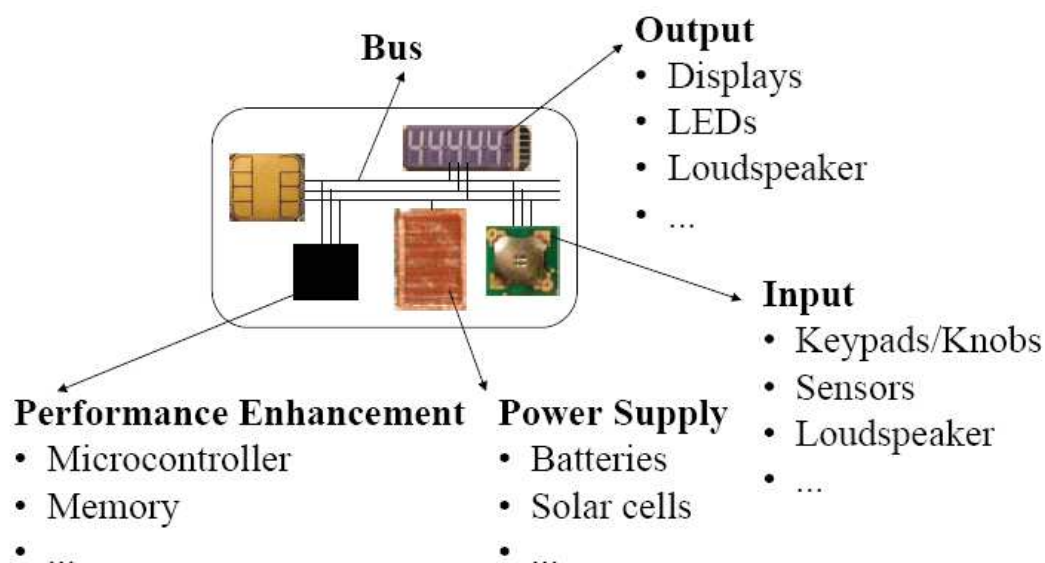
L'évolution matérielle

D'après Bertrand du Castel

- 2004 100Mhz E2+ & FullDuplex 1MB RSA2048 en 50ms
- 2006 200Mhz Bluetooth/UWB 4MB RSA2048 en 10ms
- 2008 400Mhz Bluetooth/UWB 16MB
- 2010 800Mhz Dynamic Display 50MB
 - Microphone
 - Battery (2003 : CEA Grenoble 500µm d'épaisseur)
 - FingerPrint Sensor
 - Pad
 - Rock-n-roll interface (scrolling avec la gravité)

321

About System on Card (SoC)



322

L'évolution matérielle :

Exemple : Java-Powered iButton



- <http://www.ibutton.com>
- Packaging : capsule étanche en métal
 - Peut-être monté sur bague (Java Ring)
 - « Chaque fois qu'un Gonda désirait quelque chose de nouveau, des vêtements, un bagage, des objets, il payait avec sa clef. Il pliait le majeur, enfonçait sa clef dans un emplacement prévu à cet effet, et son compte, à l'ordinateur central était aussitôt diminué de la valeur de la marchandise ou du service demandé. » La Nuit des Temps, René Barjavel
- Communication : 1-Wire
 - Un fil pour les échanges et l'alimentation
 - Débit : 16,6 Kbit/s et 144 Kbit/s
- Horloge temps réel (Secure timestamping)
- Mémoire
 - 64Ko de ROM (OS+JVM)
 - 6Ko à 135Ko de NV-RAM à 100 ns (Non Volatile RAM : 10 ans)
- API JavaCard 2.0
 - Entiers 32 bits
 - javacardx.crypto : Crypto SHA-1, RSA DES, 3DES
- Coté terminal
 - OCF, OneWireContainer
 - PKCS#11, MS CSP, X509, Win2000 log on



Joli concept
mais
concept abandonné

Év

L'évolution matérielle :

Exemple: Cyberflex eGate

- Axalto (ex Schlumberger)
- Brevet WO0016255
- Capacité
 - 32 de FlashRAM
- Format
 - ISO7816-1, GSM11.11
- Interface
 - USB/OTG (de 1.5 Mbits/s à 12 Mbits)
 - TCP/IP+USB pour la couche physique (bulk mode)
 - Full duplex
- Crypto :
 - RSA, DES, AES (très haut débit)
- Programmation
 - JavaCard
 - Mini-Serveur HTTP et SOAP



L'évolution matérielle : Exemple: la clé MicroVault

- Clé USB Micro Vault
 - clé USB à reconnaissance d'empreinte digitale.
 - 8 identifications biométriques différentes sur le MicroVault.
- Fonctions
 - File and Folder Encryption/Decryption
 - Screen Saver Lock
 - ID / Password Auto Login
 - Access to Favourites



25

L'évolution matérielle : Exemple

- Un afficheur, un clavier
 - protection contre les terminaux trafiqués
- Une horloge
 - éviter les attaques temporelles
- Une batterie
 - la mémoire volatile est plus dure à observer !
 - Le processeur peut être réveiller pour rappeler des événements au porteur



Une notion importante : Le time2market

- Temps entre la décision et le lancement d'un produit
 - Phase très longue si ré-écriture d'un masque (peut atteindre 1 an)
 - Peu adapté aux besoins du marché :
 - La téléphonie mobile est très concurrentiel
 - Les coûts de développement sont très important
- Apparition d'un nouveau type de carte : les cartes génériques
 - Disposent d'un véritable OS
 - Peuvent charger des applications au cours du cycle d'utilisation

327

Les cartes à puce du futur

*Des cartes ... encore ... toujours ...
... toujours plus ...
... partout ...*



*Comment résoudre
ce problème ?*



La carte multi-applicative

328

Oyster Card et Barclays

Oyster – Facile et pratique pour voyager pas cher dans Londres.
Chargement de crédits ou carte d'abonnement.
C'est une fonctionnalité totalement séparé du compte associé à la carte de crédit



Cashless – Technologie Visa "wave and pay" pour des petits achats (inférieurs à £10) sans PIN.
Les transactions apparaissent sur le relevé de compte.

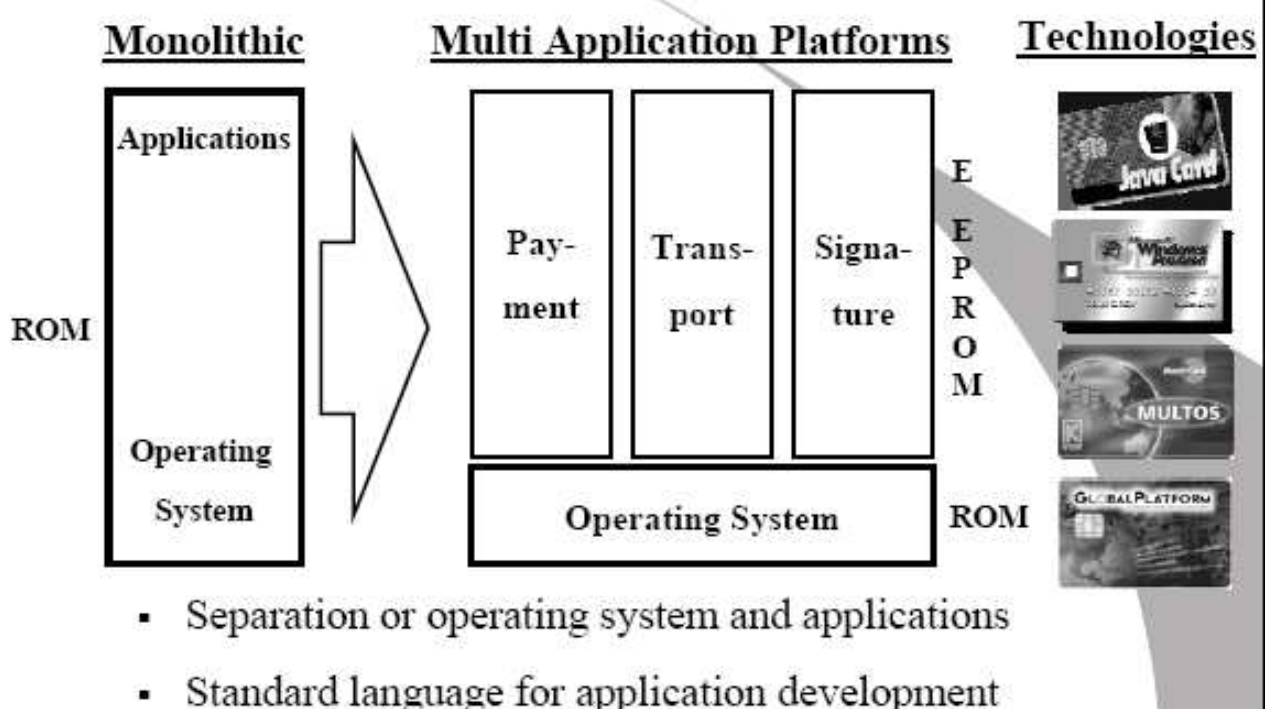


Credit – Application Visa classique avec PIN pour les paiements supérieurs à £10.



329

Evolution

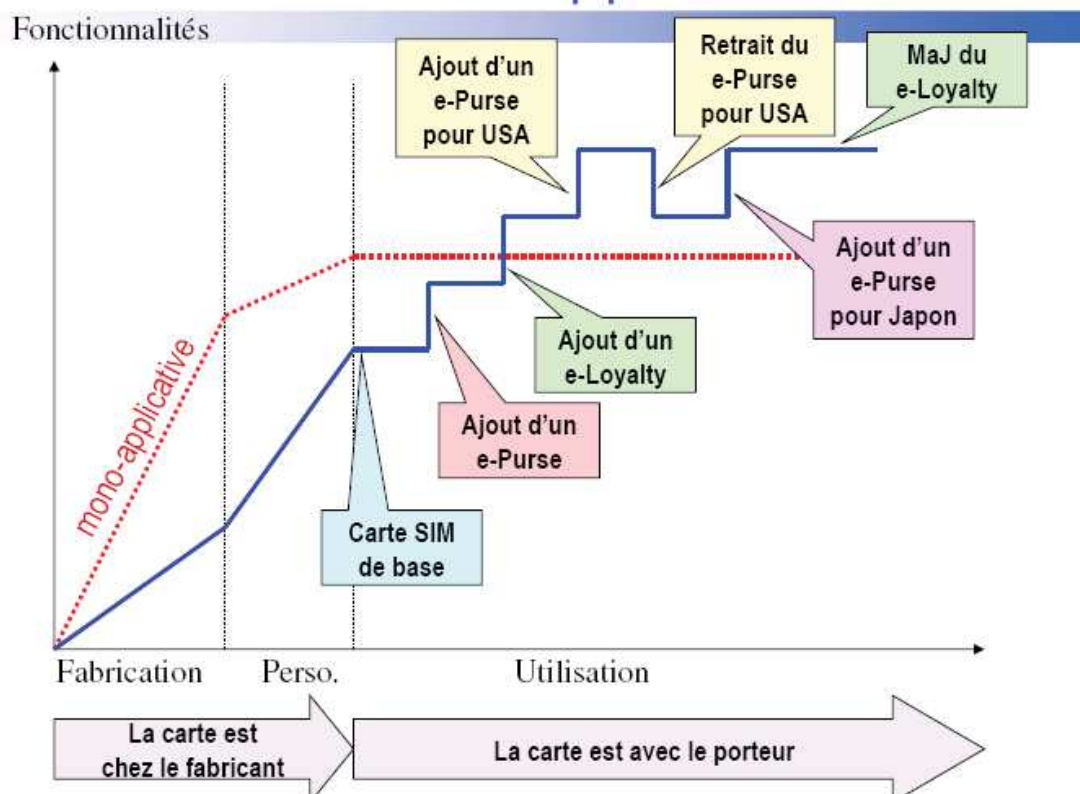


330

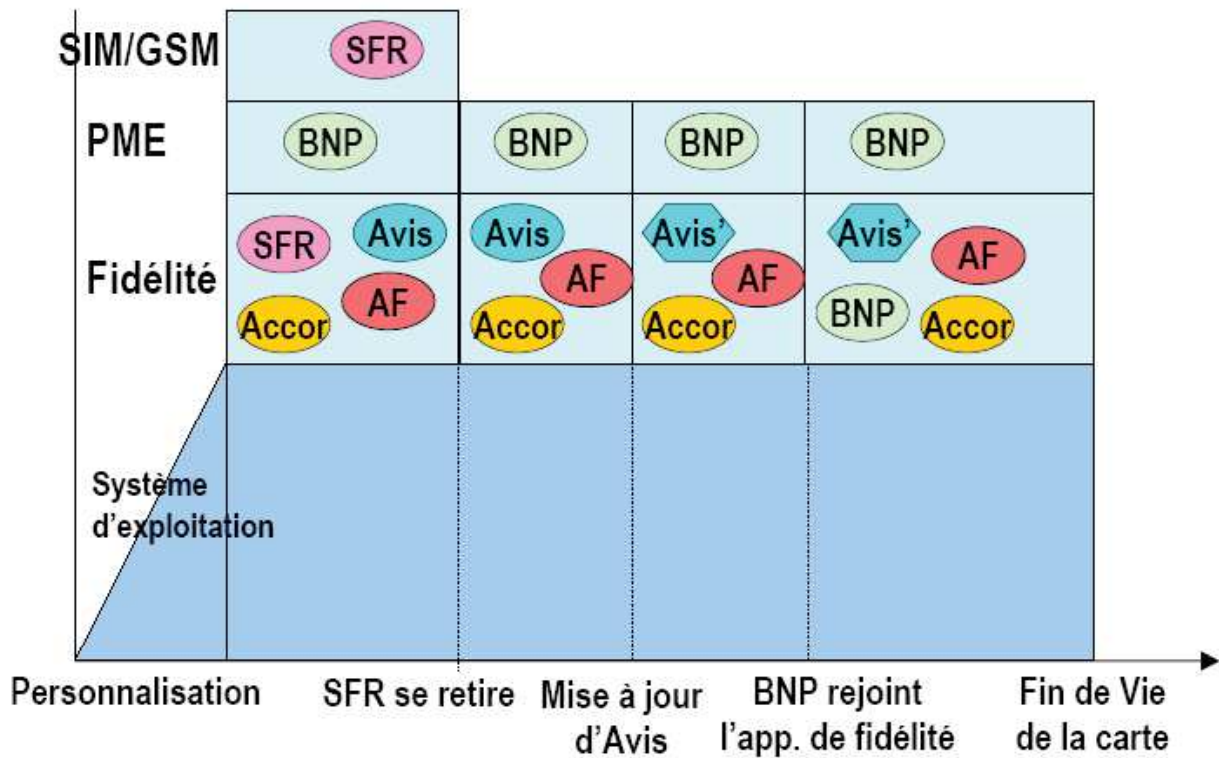
Une nouvelle génération : Les cartes multi-services

- But :
 - Permettre à plusieurs applications de coexister
 - Partager des données entre plusieurs applications (non redondance d'information)
 - Possibilité de charger/décharger des applications en cours de vie de la carte
- Avantages
 - une seule carte pour l'utilisateur
 - Possibilité d'évolution des cartes
- Inconvénients
 - Peur des partages
 - Tailles réduite de la place / application

Le cycle de vie des cartes multi applicative

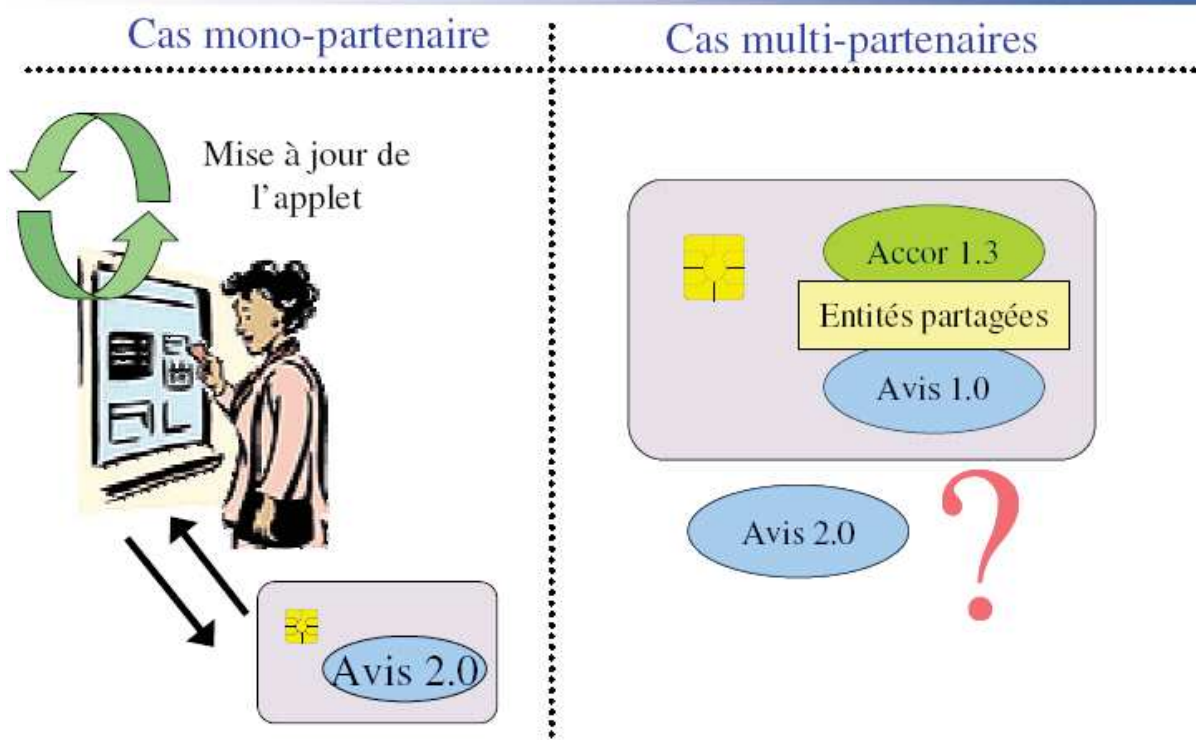


Cycle de vie d'une carte multi-partenaires



2007

Évolution d'applications Cartes multi-applications



Besoins des développeurs

- Sortir la carte d'une programmation «élitiste » :
 - Seul les programmeurs de longue date savent correctement écrire une application carte
 - Solution : utiliser des langages de programmation courants en informatique (C, Java, Visual Basic)
- Permettre de tester des solutions :
 - portage sur carte plus rapide => possibilité de test

Besoins des utilisateurs

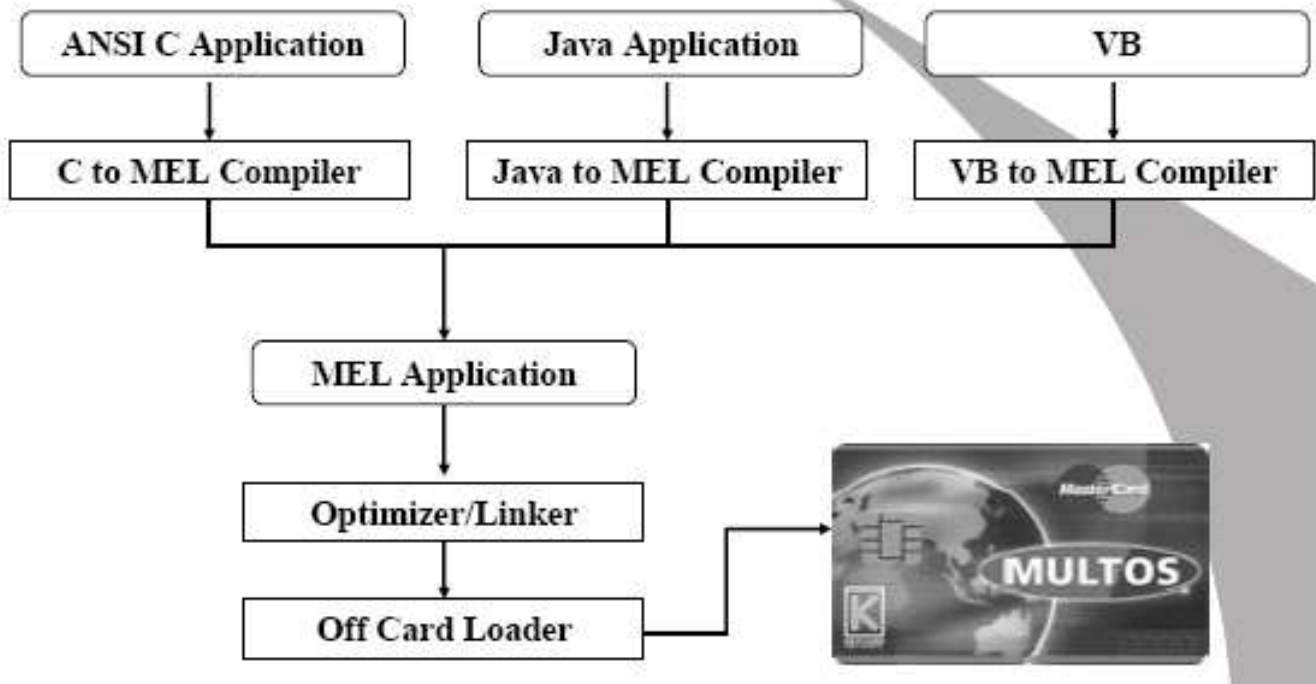
- Se simplifier la vie :
 - utiliser la même carte pour toutes les applications
 - Plusieurs applications sur une même carte
 - Pouvoir changer de prestataire sans avoir à recommander une carte
 - Chargement / déchargement d'application

335

Multos

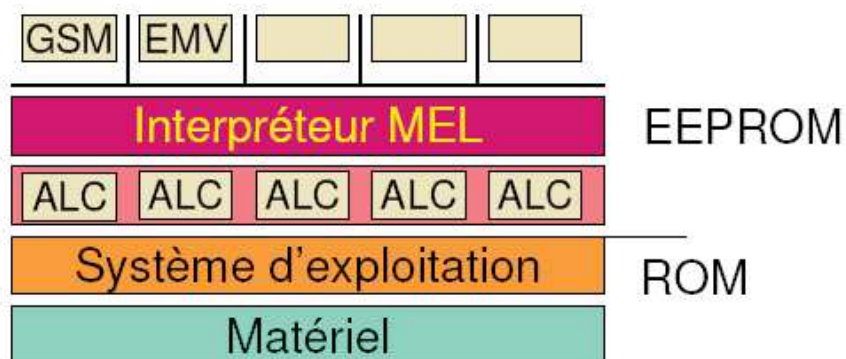


Multos Application Development



Multos

- Carte basée sur un interpréteur MEL (Multos Executable Language)



- Carte limitée par la norme 7816-4
- Plus d'info : <http://www.multos.com>

BasicCard

- Développement d'applications carte en langage Basic
 - Coté Carte
 - interpréteur de « bytecode » P-Code
 - Système de fichier DOS-like
 - Coté Terminal
 - Interpréteur de « bytecode » P-Code
 - Remarque : l'application terminal peut être écrite aussi en C/C++/C# avec un driver PS/SC ou MUSCLE en Java avec OCF
- Avantages
 - Aux applications en très petites séries (club de tennis, ...)
 - Pour une première approche pédagogique de la carte
- Inconvénients
 - Propriétaire (<http://www.zeitcontrol.de/>)
 - Pas de soutien des *Majors*

339

.NET SmartCard

<http://www.hiveminded.com/>

- Récent (2003)
 - Effet d'annonce ?? Souvenez vous de Windows SmartCard !!
- Caracteristiques
 - Implémentation de la CLI adapté à la carte
 - Multi-applications
 - Développement Multi-langage : C#, J#, VB, Jscript, Perl, ...
 - Isolation
 - Application Domain de .NET
 - Transactions
 - Multi-niveaux ?
 - Garbage Collector
 - Mark and Sweep (sans marquage en EEPROM)
 - Communication
 - Inter-applications
 - Channel : flux d'octets bidirectionnel
 - Terminal-Application : APDU, .NET Remoting, Javacard 2.2 RMI



.NET SmartCard

<http://www.hiveminded.com/>

- Implémentation de la CLI adapté à la carte
- Multi-applications
- Développement Multi-langage : C#, J#, VB, Jscript, Perl, ...
- Caractéristiques
 - Isolation
 - Application Domain de .NET
 - Transactions
 - Multi-niveaux ?
 - Garbage Collector
 - Mark and Sweep (sans marquage en EEPROM)
 - Communication
 - Inter-applications
 - Channel : flux d'octets bidirectionnel
 - Terminal-Application
 - APDU, .NET Remoting, Javacard 2.2 RMI

Technology Overview 1/2

Criteria	GP	Multos	WfSC	JavaCard
Concept	Platform	SCOS	SCOS	VM & API
Virtual Machine Technology	Interpreter	Interpreter	Interpreter	Interpreter
Byte Code	No predefined byte code	MEL	TBC	Java byte Code
Performance	N/A	2-3 times slower then native	TBC	2-3 times slower then native
VM & SCOS Extensions	Yes, subject to implementation	Yes, subject to ITSEC implementation	YES	Yes, subject to implementation
Security Level	EAL4+(France). EAL5+(Germany)	ITSEC 6 Achieved	Microsoft Approval	EAL4+, EAL5+
Based	Chip, SCOS, JVM, OP	Chip and SCOS	Chip, SCOS	Chip, SCOS, JVM
Crypto Engine	Yes (Optional)	YES	No	Yes (Optional)

Technology Overview 2/2

Criteria	GP	Multos	WfSC	JavaCard
Post Issuance Application Loading	Card Manager Security Domains, etc.	Application Load & Delete Certificates	Yes, protocol & security not defined	Yes, protocol & security not defined
Memory Requirements	~8K	~4K VM, ~7K Executive	TBC	Java Card VM: ~16K Java Card API: ~8K
Libraries	~8K for crypto	~15K for crypto	TBC	--
Programming Languages	No predefined language	MEL, C, even Java or VB	VB (Subset)	Java (Subset)
Extensible APIs	Yes (Third Party)	Yes, only after evaluation	Microsoft Verified code extensions	Yes, Third party
Processors	OS depended	Optimised for 8-bit processors	Designed for 8-bit processors	Optimised for 32-bit but available on 8-bit
Portability	Portable across any Java Card or WfSC	Fully portable across Multos implementations	Portable across WfSC implementations	Portable across 2.1 compliant Java Cards

343

Business Overview 1/3

Criteria	OP	Multos	WfSC	JavaCard
Specification Open/Closed	Open	"Open"	"Closed"	Open
Controlled/Maintained	Global Platform	MAOSCO	Microsoft	SUN
Current Version	OP 2.1	Multos 5	WfSC 1.1	Java Card 2.2
Model	Issuer Centric & delegated management	Issuer Centric	Not Specified	Not Specified
Licence Required	No	Yes	YES	Yes
From	N/A	MAOSCO	Microsoft Approval	SUN
What for	N/A	Multos OS & VM	Implementation	VM Implementation
Who Pays	N/A	SCOS Implementer	SCOS Implementer	SCOS Implementer

Business Overview 2/3

Criteria	GP	Multos	WfSC	JavaCard
Cost Evaluation	Depends on the level, country, deadline, etc.	~\$600,000 (Varies)	Depends on the level, country, deadline, etc.	Will add ~40%-150% to the cost of the chip
Licence	N/A	£250,000+ £50,000 per annum	TBC	\$400,000+ \$70,000 Per annum
Card	~\$3 PK <\$2 non PK	~\$3-\$5 PK Card	~\$2-\$8	~\$2-\$7
Application Load/Delete Royalties (Issuer)	No	£0.025	Depends on the model	No
Application Loading Cost (Issuer)	~\$0.30-\$0.60, manufacturer Specific	~\$0.30-\$0.60, manufacturer Specific	~\$0.30-\$0.60, manufacturer Specific	~\$0.30-\$0.60, manufacturer Specific

Business Overview 3/3

Criteria	GP	Multos	WfSC	JavaCard
Card Management Systems	Several available	Several available	Several available	Several available
Enhancing Acceptance	Liaise with ETSI (not official member yet)	Member of ETSI	TBC	Member of ETSI
Personalisation and Back Office Systems	> 30 Vendors at special prices	~ 17 vendors	TBC	TBC
SIM Cards	>200 million cards	Multos SIMS since early 2001	N/A	>200 million Java Cards (GSM 03.48 [GP cards] and GSM 03.29[JC SIM card])
Multi Sourced	YES	"YES"	YES	YES
Development Tools	Same as standard Java	MDS, Hitachi, QuickSmart, etc.	VB/C++ Programming Environments	SUN, Symantec, Cyberflex, Gemplus, etc.

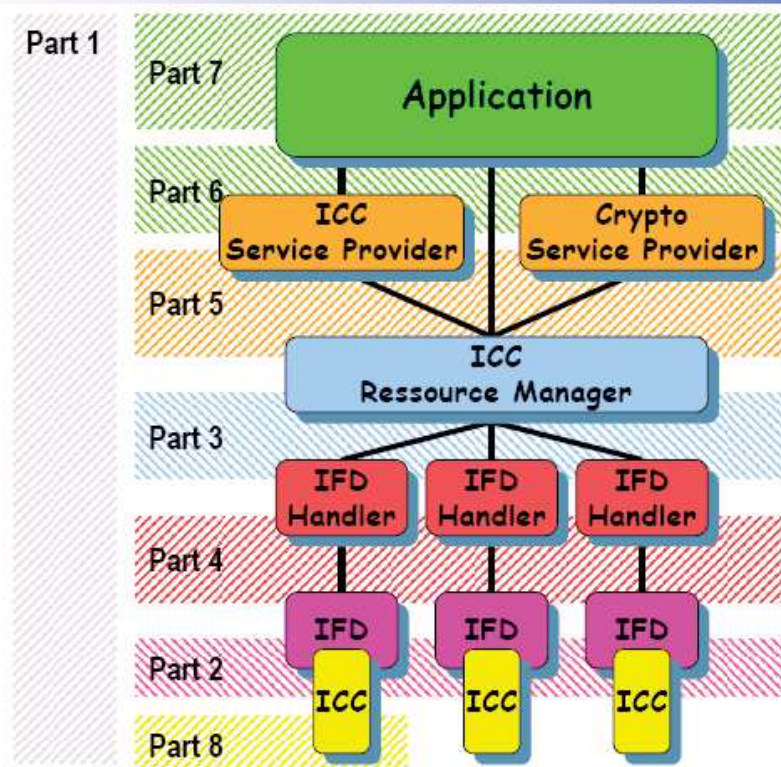
The Winning Platform...



PC/SC

- Define Specifications
 - Platform and OS independent
 - Windows
 - Linux, Apple (MUSCLE)
 - Compliant with ISO7816, support and endorse industry specifications (EMV, GSM...)

Architecture et Spécifications PCSC



349

Bibliographie

Cours de Gemplus

Cours de Cartes à Microprocesseur de :

Didier Donsez (Univ. Joseph Fourier, Grenoble 1)
Gilles Grimaud (Univ. Lille 1)
Sébastien Jean (Univ Pierre Mendès France, Grenoble 2)
Sylvain Lecomte (Univ. Valenciennes)
Jean-Jacques Vandewalle (Gemalto R&D)

Cours du CNAM de Samia Bouzefrane

« Smart Card Handbook » de Rankl & Effing

Courses of the RHUL : Markantonakis et Mayes

- Javacard
 - <http://java.sun.com/products/javacard/specs.html>
- Multos
 - www.multos.com
- Windows for Smart cards
 - www.microsoft.com/smartcards
- GlobalPlatform
 - <http://www.globalplatform.org/>
- PC Client APIs
 - www.pcscworkgroup.com
 - <http://www.opencard.org/>
- Smart card & Security
 - www.gemplus.com/publications
 - www.iaik.tugraz.at

Webographie

- Une page pleine de références
 - <http://paroissien.free.fr/>

350

Bibliographie

- Livres carte

- La Carte à Puce, Coll Que Sais Je ?, n°3492, Ed PUF, 1999
- Henry Dreifus & J. Thomas Monk, *smartcards -- A guide to building and managing smart card applications*, ISBN: 0-471-15748-1, New York: John Wiley & Sons, 1998.
- Scott B. Guthery & Timothy M. Jurgensen, *SmartCard Developer's Kit*, ISBN: 1-57870-027-2, Indianapolis, Indiana: Macmillan Technical Publishing, 1998.
 - <http://www.scdk.comm>

- Ressources

- Ressources du site CITI
 - <http://www.citi.umich.edu/projects/smartcard/>
- FAQ
 - <http://www.scdk.com/atsfaq.htm>
- Google
 - <http://directory.google.com/Top/Computers/Hardware/Systems/Smartcards/>

1

Bibliographie

- Articles introductifs

- "The smart card primer," Rinaldo Di Giorgio (JavaWorld, December 1997)
 - <http://www.javaworld.com/jw-12-1997/jw-12-javadev.html>
- How to write a Java Card applet: A developer's guide
 - http://www.javaworld.com/javaworld/jw-07-1999/jw-07-javacard_p.html
- Sun's Java Card page contains the specifications of the Java Card APIs, the Java Card Virtual Machine, and the Java Card Runtime Environment:
 - <http://java.sun.com/javacard>
- "Get a jumpstart on the Java Card," Rinaldo Di Giorgio (JavaWorld, February 1998):
 - <http://www.javaworld.com/javaworld/jw-02-1998/jw-02-javadev.html>
- "Understanding Java Card 2.0," Zhiqun Chen with Rinaldo Di Giorgio (JavaWorld, March 1998):
 - <http://www.javaworld.com/javaworld/jw-03-1998/jw-03-javadev.html>

2

Bibliographie

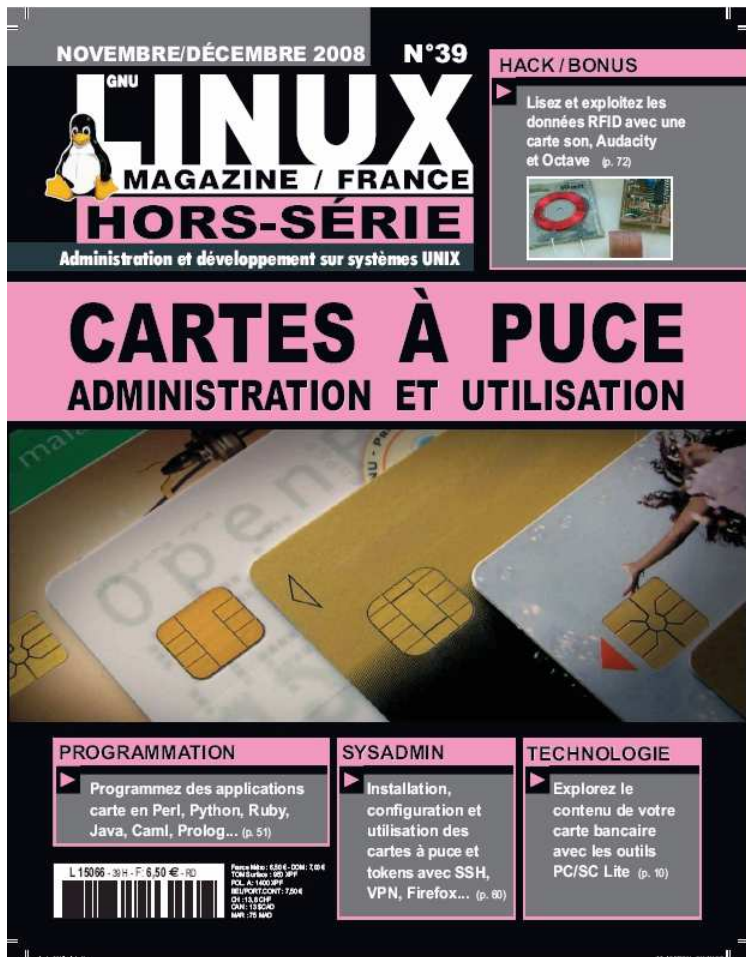
- API d'accès aux cartes
 - Uwe Hansmann, Martin S. Nicklous, Thomas Schäck, Frank Seliger, *Smart Card Application Development Using Java*, Ed Springer, 2000, ISBN: 3-540-65829-7,
 - <http://www.opencard.org/SCJavaBook>
 - Orienté OCF et livré avec une carte pour les tests
 - Open Card Framework (OCF)
 - <http://www.opencard.org/>
 - <http://www.gemplus.fr/developers/technologies/opencard/index.htm>
 - PC/SC
 - <http://www.smartcardsys.com/>
 - <http://www.microsoft.com/smartcard/>
 - <http://www.pcscworkgroup.com/>
 - MUSCLE pour Linux
 - <http://www.linuxnet.com/>

3

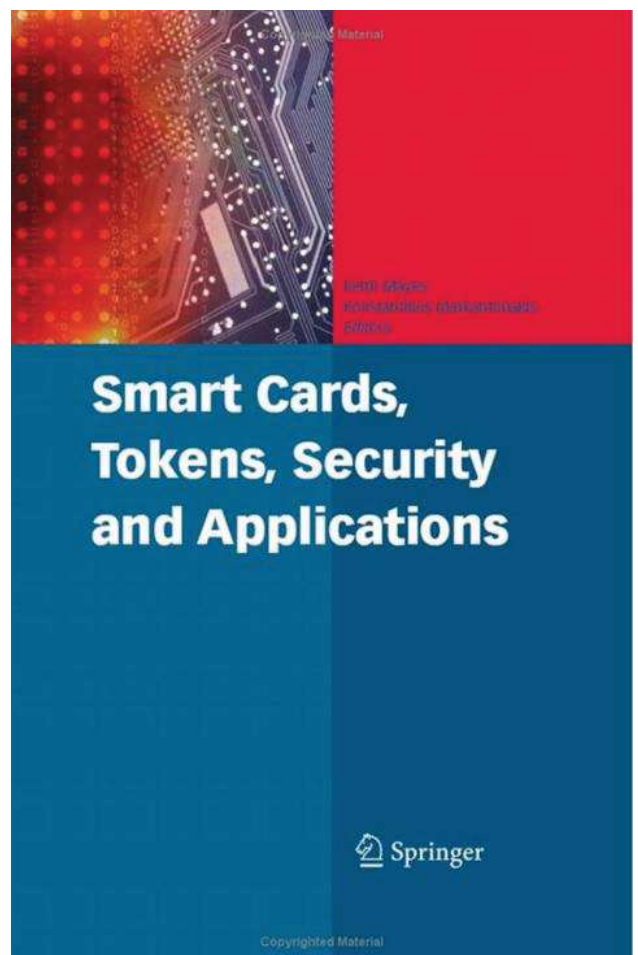
Bibliographie

- Jack M. Kaplan, *SmartCards -- The Global Information Passport*, ISBN: 1-850-32212-0, Boston, Massachusetts: Thomson Computer Press, 1996.
- Mike Hendry, *Smart Card Security and Applications*, ISBN: 0-89006-953-0, Norwood, Massachusetts: ARTECH House, Inc., 1997.
 - Présentation des concepts de la sécurité et des principales applications de la carte
- W. Rankl & W. Effing, *Smart Card Handbook*, ISBN: 0-47196-720-3, New York: John Wiley & Sons, 1997.
- Chuck Wilson, *Get Smart*, 338 pages (June 1, 2001), Mullaney Corporation; ISBN: 0967446058
 - Orienté décideurs (pas de programmation)

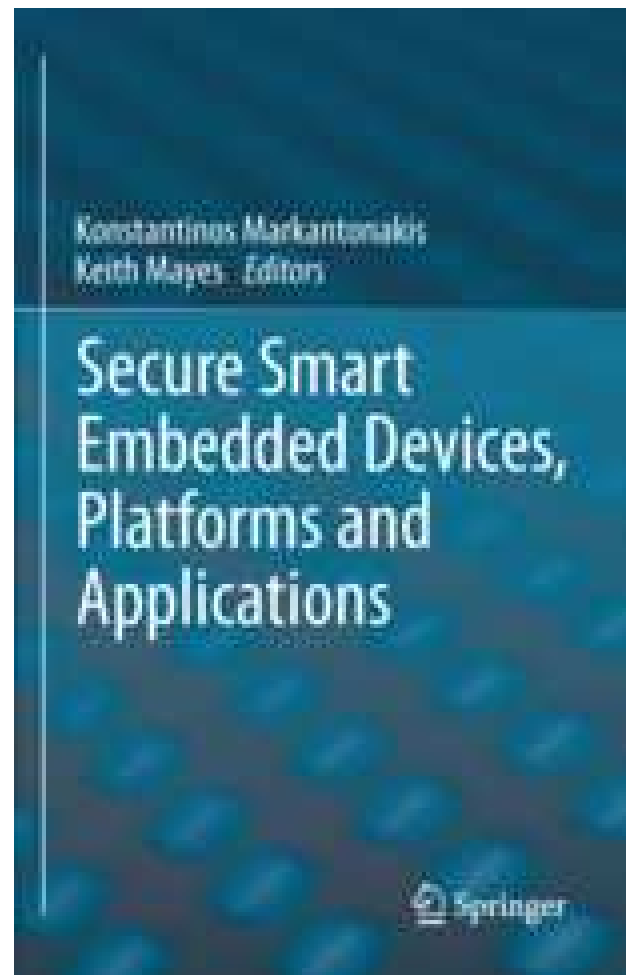
...



Smart Cards, Tokens, Security and Applications
Mayes, Keith; Markantonakis, Konstantinos (Eds.)
2008, XXXVIII, 392 p. 128 illus., Hardcover
ISBN: 978-0-387-72197-2



Secure Smart Embedded Devices, Platforms and Applications
Konstantinos Markantonakis, Keith Mayes
2014, XLI, 568 p. 135 illus., Hardcover
ISBN: 978-1-4614-7914-7



Les cartes à puce
BOUZEFRANE Samia, PARADINAS Pierre
Ouvrage 326 p. 15.6x23.4 cm Relié



Manifestations

Cartes (Paris en novembre) www.cartes.com

Conférence e-Smart (Nice, septembre)

Conférence Cardis (tous les deux ans)

Conférence CHES (tous les ans)

3GPP

...

359

Questions ?

© Original Artist
Reproduction rights obtainable from
www.CartoonStock.com



"My smart-card doesn't
understand me."

**Si votre carte ne vous comprend pas,
la comprenez vous mieux ?**





Java Card Grid Project

361



Présentation des grilles de calculs

Grille : ensemble de machines interconnectées.

Principe :

Partager de GROS calculs entre plusieurs machines.

Utilisateurs :

Organismes de la défense, CEA, laboratoires de recherche et universités, ...

Domaine d'utilisation :

Cryptographie, physique des particules, bio-informatique, observation de la terre, ...

Un des problèmes important : *La sécurité!*

362

Quelques projets utilisant la grille

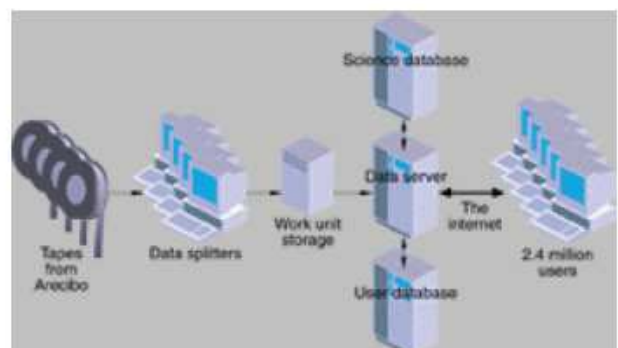


Décrypton

363

Seti@home

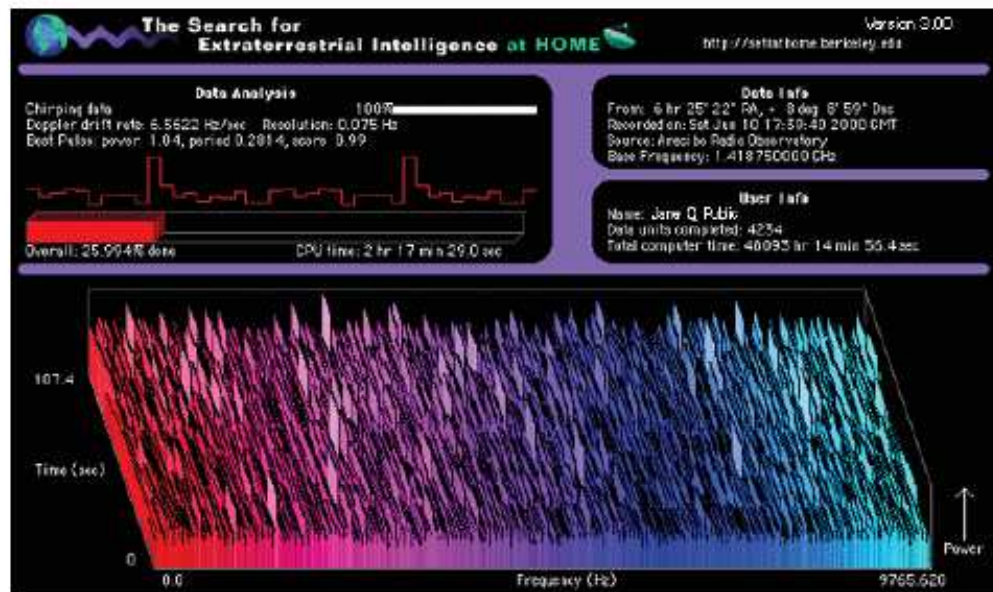
But : Recherche de signaux extra-terrestres.



364

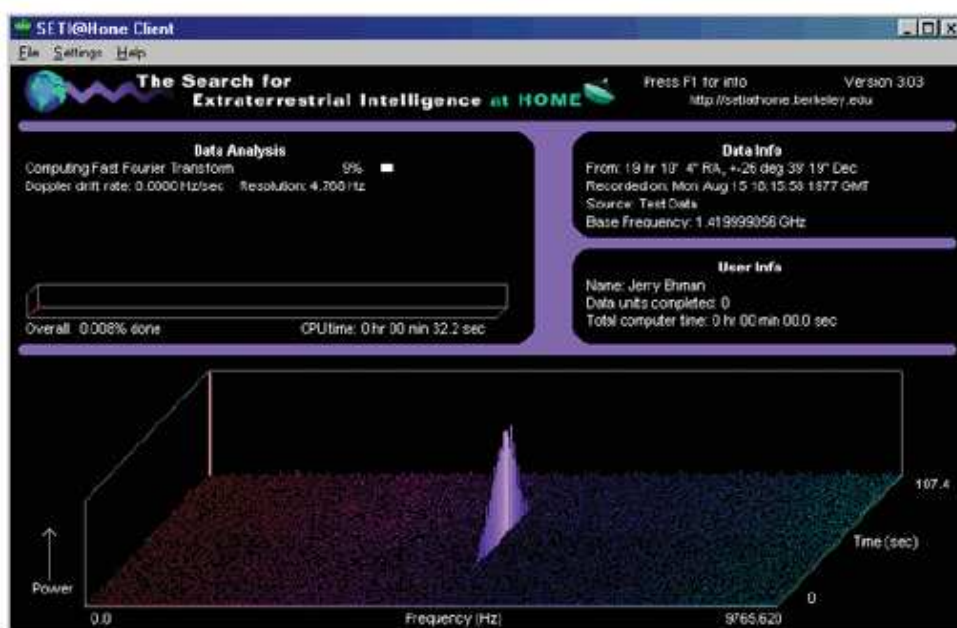
Les calculs

L'utilisateur télécharge un screensaver puis les données.



35

Les résultats



Nous ne
sommes PAS
SEULS dans
l'univers ?

SI! $\Rightarrow$ Il s'agit de résultats truqués! :-(

300

Les problèmes

Côté Seti@home :

- ☞ Les résultats positifs ou négatifs ne sont pas sûrs.
⇒ *solution* : Faire calculer les mêmes données plusieurs fois par différents utilisateurs

Côté utilisateur final :

- ☞ Code source non disponible.
⇒ éviter les faux résultats.
- ☞ Que fait le code sur la machine de l'utilisateur ?

367

Le projet “Java Cards Grid”

Objectifs :

- ☞ Garantir la sécurité du code et des calculs sur la grille.
- ☞ Proposer un framework logiciel exploitant du matériel sécurisé pour assurer la confiance des utilisateurs.

Avantages pour :

- les utilisateurs du *distributed computing* : une plus grande confiance dans le fournisseur des ressources de calculs ;
- les fournisseurs des ressources de calculs : accroître son marché avec des nouveaux clients comme le CEA, ...

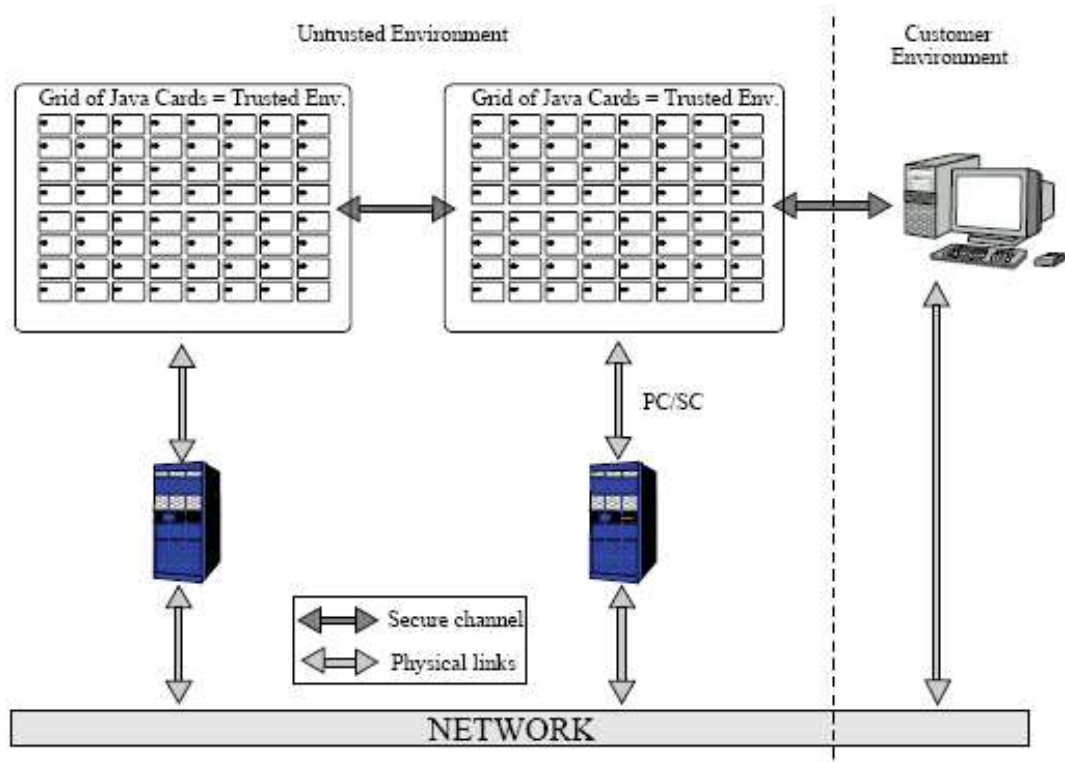
368

Buts de notre projet

- Premier but : Fournir un environnement sécurisé pour partager des ressources (matérielles, logicielles) et de la puissance de calcul
- Second but : Fournir une plate-forme pour expérimenter sur les réseaux ad hoc

369

Notre proposition



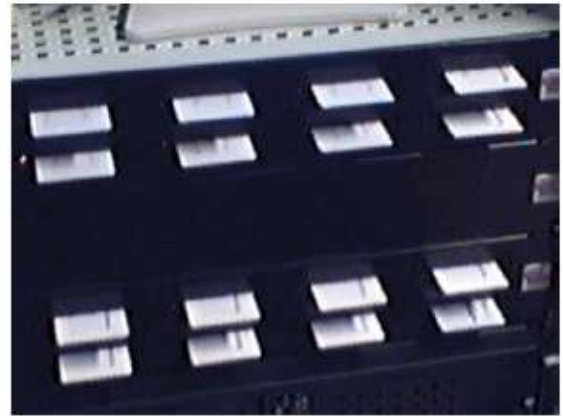


371

La Plate-forme Actuelle



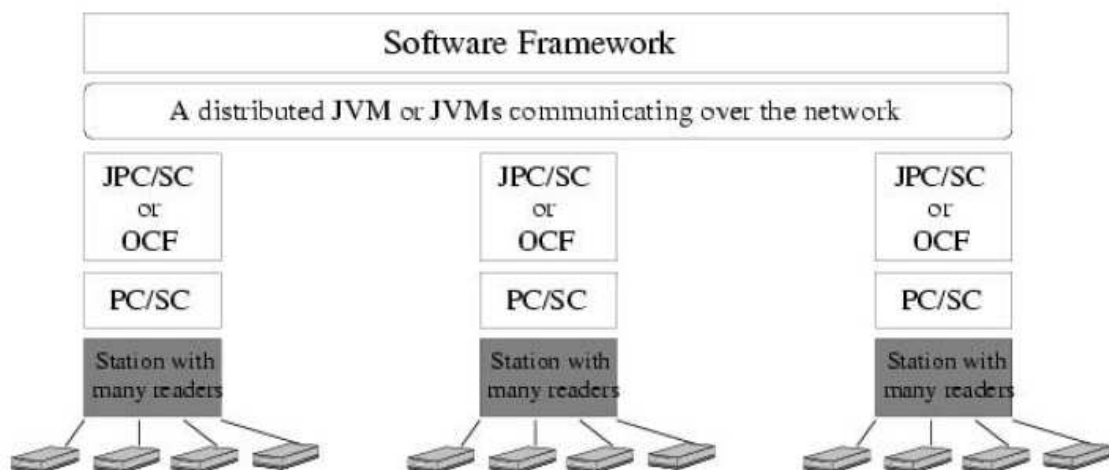
- 2 fois
 - 1 PC
 - 16 lecteurs de cartes
 - 1 point d'accès Wi-Fi
- Les deux machines sont connectées entre elles (réseau filaire)



373

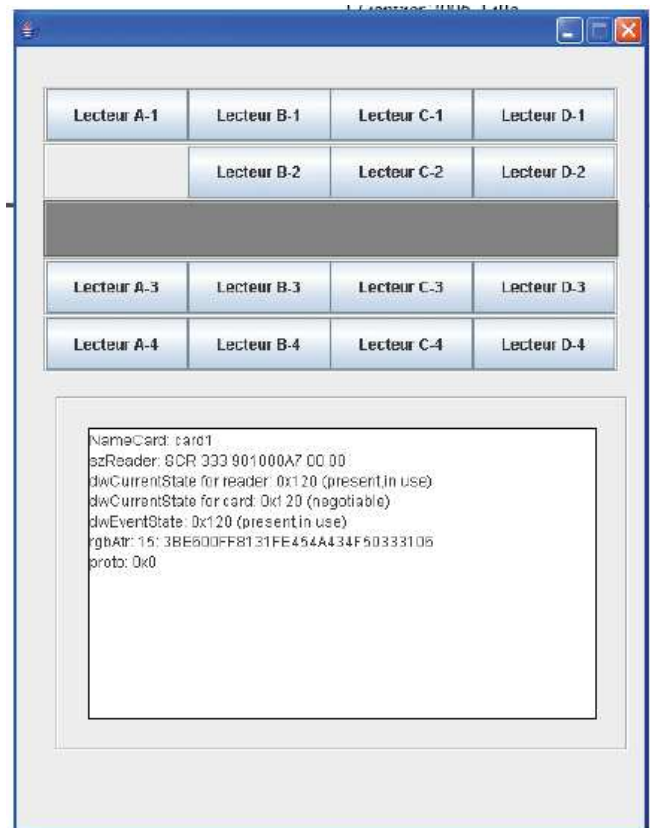
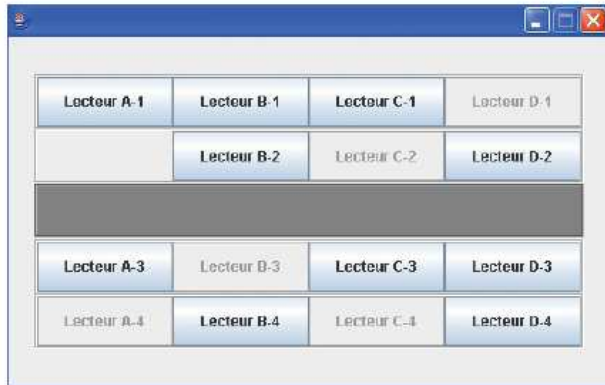


L'environnement logiciel



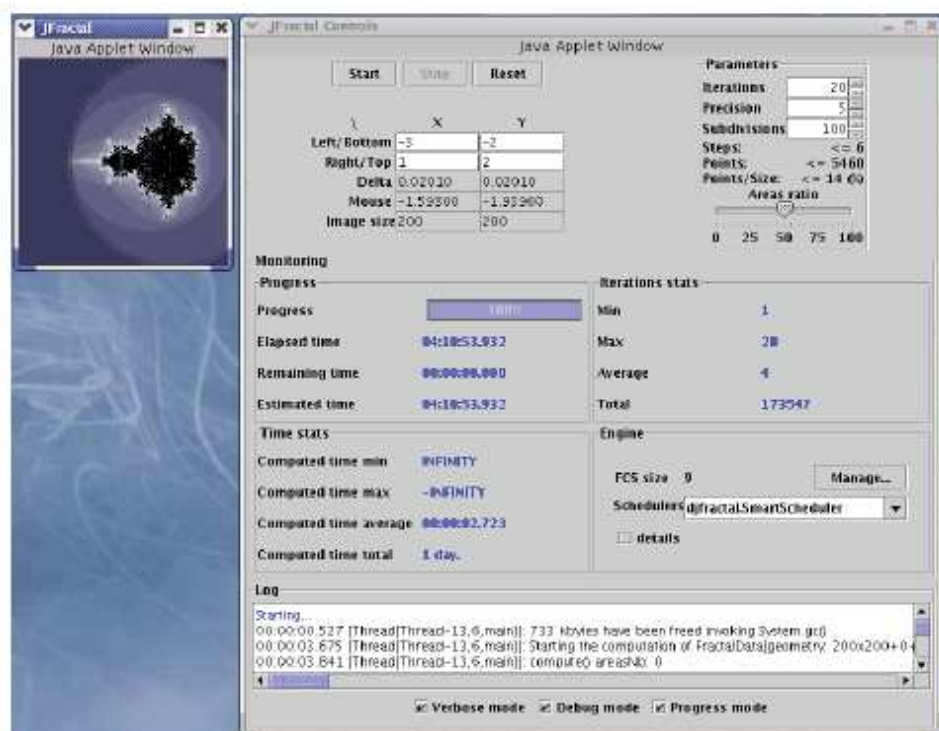
374

L'environnement d'administration



375

Première démonstration



376

Quelques benchmarks

9 Java Card onto Fujitsu mb94r215b (FRAM)	122 min
9 GemXpresso Pro R3 E32PK	202 min
9 JCOP31bio	253 min
9 GemXpresso Pro R3 64PK	376 min
9 SmartC@fé Expert	389 min

TAB. 1 – Benchmarks sans secure channel

Scalabilité en cours d'expérimentation.

1 Java Card onto Fujitsu mb94r215b (FRAM) $\Rightarrow 18\text{heures} = 9 * 120\text{min}$

377

Application CBP/Air France

- On imagine (☺) que le *US Customs and Border Protection* (CBP) souhaite fouiller les fichiers passagers des compagnies aériennes comme Air France, pour les vols utilisant un aéroport US.
- Problème : la CEE interdit explicitement la diffusion d'informations confidentielles sur les passagers.

378

Les Contraintes du Problèmes

- Code de fouille du CBP
 - Distribué sur les cartes
 - Doit demeurer confidentiel
- Fichier passagers d'Air France
 - Distribué sur les cartes
 - Doit demeurer confidentiel

3 / 3

Etat Actuel et Travaux en Cours

- État actuel
 - La plate-forme est opérationnelle
 - Prix « Best innovative technology » à e-smart 2005
- Travaux en cours
 - Gestionnaire de swap sécurisé
- Autres travaux liés :
 - Aspects liés au ad hoc
 - une thèse en cours avec la DGA

Perspectives

- Aspects logiciels
 - Gestion des pannes
 - Assemblage de composants
- Passage à l'échelle
 - Avec plus de cartes:
 - 1000 lecteurs envisagés
 - Sera mené avec un industriel
 - (confidentiel pour l'instant)
 - Avec de vrais processeurs
 - Trusted Computing Group

381

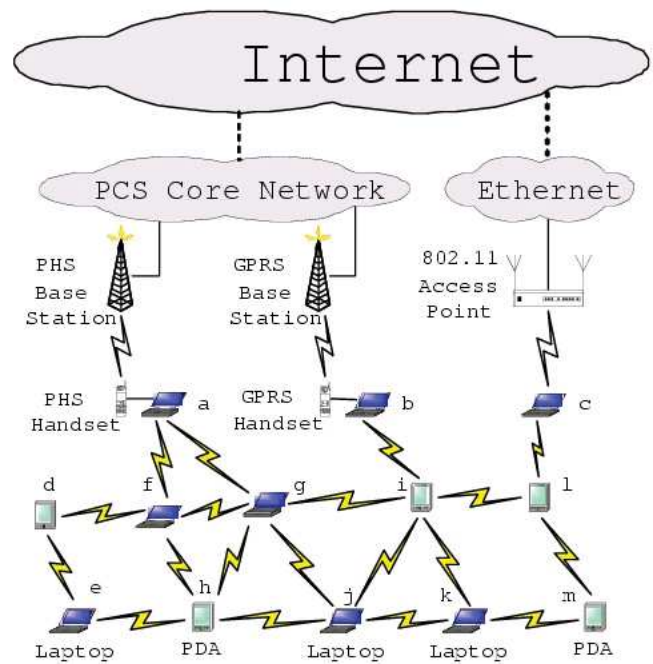
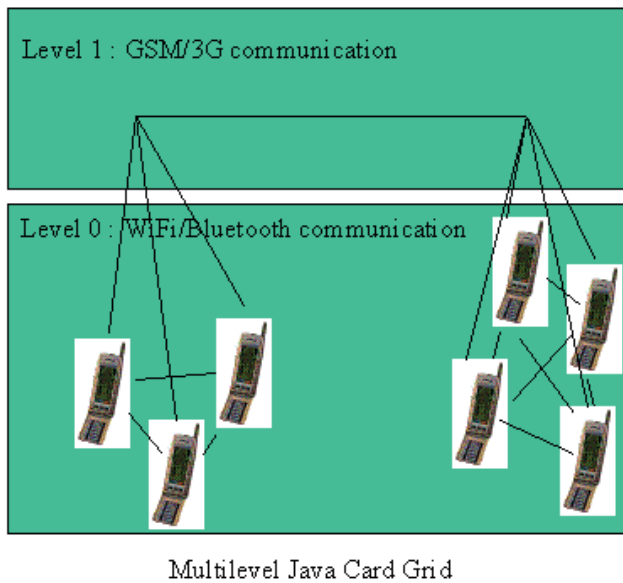
Soutiens industriels

- Sun Microsystems
 - IBM
 - Oberthur
 - Gemplus
 - Schumberger
 - Smartmount
 - SCM
 - Giesecke & Devrient GmbH
- Fujitsu



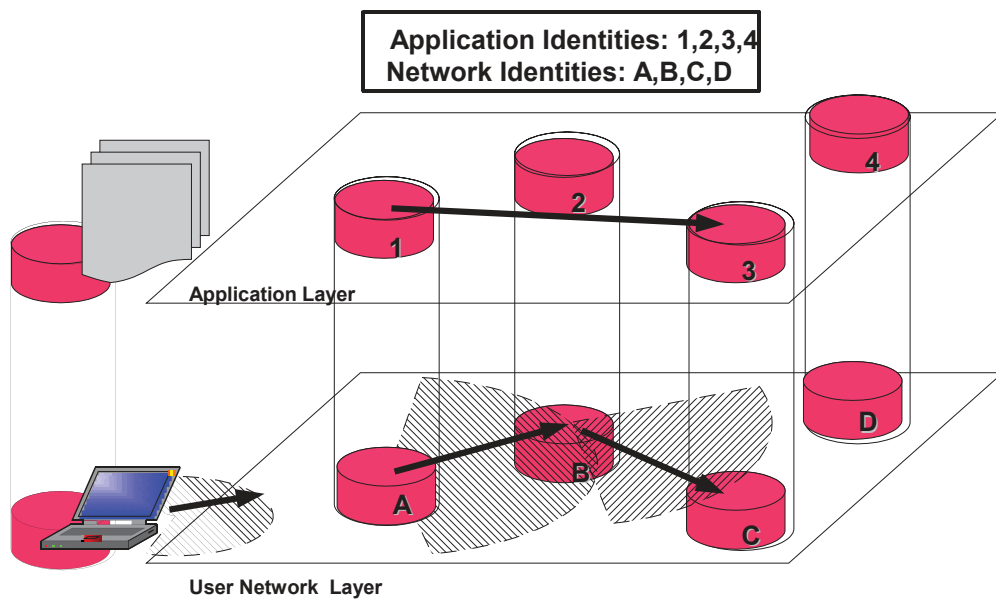
382

La multilevel Java Card Grid (1/2)



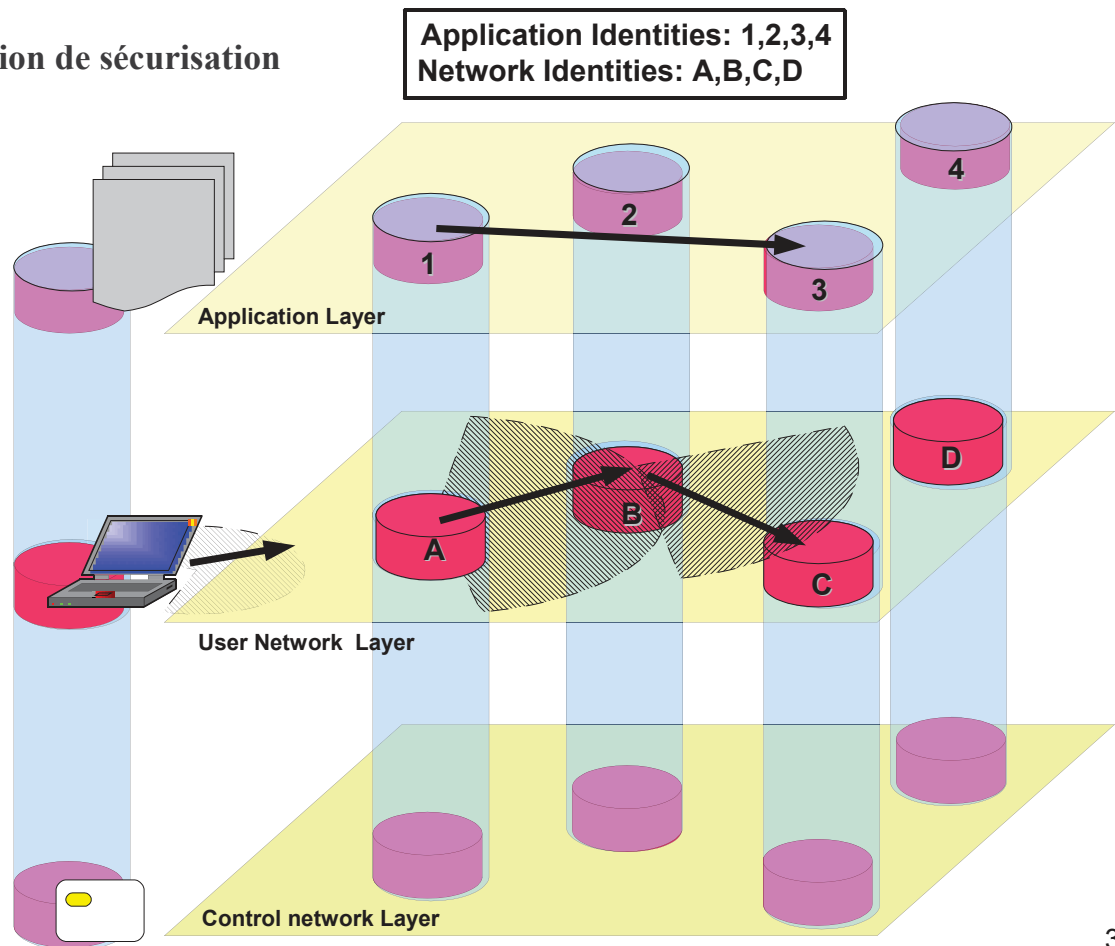
383

Les réseaux ad hoc



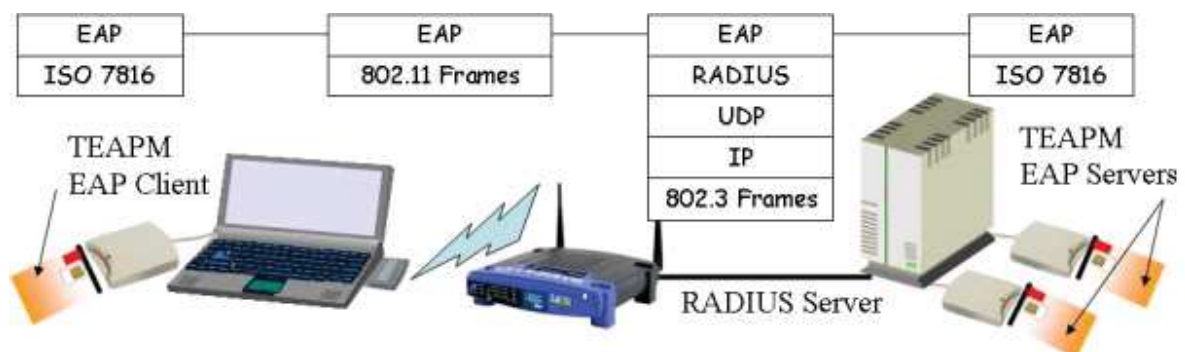
384

Une solution de sécurisation



385

Sécurisé le réseau WiFi avec EAP



EAP: Extensible Authentication Protocol

