contactless technologies • EuraRFID

Travaux Pratiques cartes sans contact et NFC:

Echange de commandes APDU avec une Mifare Classic 1K et réalisation d'un Smart Poster à partir d'une Mifare Ultralight.

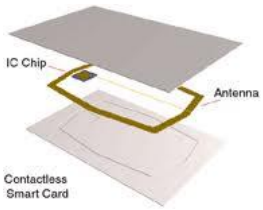
Le TP suivant se décompose en deux parties : la première consiste à échanger des commandes APDU avec une carte de proximité Mifare Classic 1K afin d'explorer sa mémoire et d'exploiter les données qu'elle contient. La deuxième partie consiste à réaliser un Smart Poster à partir d'une Mifare Ultralight, en utilisant les commandes assimilées lors de la première partie.
Les notions : « cartes de proximités », « commandes APDU », « Mifare Classic 1K», « Mifare Ultralight » et « Smart Poster » seront explicitées au fur et à mesure de l'avancement du TP.

Ces travaux pratiques nécessitent des prérequis ou une formation préalable aux bases de la RFID HF.

Note : Le matériel et les logiciels mis à votre disposition sont la propriété du CITC-EuraRFID. Ceux-ci ne peuvent en aucun cas être utilisés et/ou conservés en dehors de la séance de travaux pratiques.

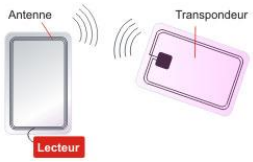
Citc • Contactless Innovation Technologies Center
Centre d'Innovation des Technologies sans Contact
CITC EuraRFID – EuraTechnologies – 165 avenue de Bretagne – 59000 LILLE
Tél. + 33 (0) 320 191 852 - Fax + 33 (0) 320 936 963
N° de SIRET : 511 568 602 00011
www.citc-eurarfid.com

Introduction :

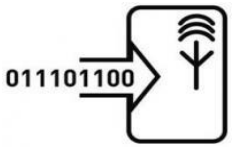


Les cartes à puce sans contact étudiées dans ce présent document reposent sur une technologie de communication en champ proche à haute fréquence, plus communément appelés RFID (**R**adio **F**requency **I**dentification) à haute fréquence.
La fréquence de fonctionnement de ces systèmes HF (haute fréquence) est de 13,56 MHz.

Il est à noter que d'autres plages de fréquences sont utilisées pour les technologies RFID : par exemple la technologie RFID basse fréquence fonctionne aux alentours de 125KHz contre ~868MHz[1] pour la RFID ultra haute fréquence (UHF RFID).



Ces systèmes RFID sont composés de deux entités qui communiquent entre elles par liaison Radio Fréquence (RF) : Un transpondeur (étiquette intelligente) et un lecteur.



Le transpondeur est composé d'une puce reliée à une antenne. Ainsi le lecteur transmet l'énergie nécessaire à l'alimentation[2] de la puce du transpondeur. Cette énergie sert également de medium de communication entre les deux entités.

En RFID haute fréquence, la lecture d'une carte (transpondeur) nécessite une distance inférieure à moins de 10 cm.
Il existe plusieurs normes, standards et protocoles définissant la communication sans contact à haute fréquence (à 13,56MHz) entre deux supports (carte et lecteur par exemple).
Les cartes sans contact proposées à l'étude sont basées sur la norme ISO 14443.

[1] Cette valeur correspond aux fréquences UHF dans la bande européenne allant de 865 MHz à 868 MHz.
[2] Nous traitons dans ce document uniquement le cas des transpondeurs passifs sans batterie.

Mise en Œuvre :

Nous allons, dans ce TP, envoyer des commandes à la carte afin de lire et de modifier le contenu de sa mémoire. Pour ce faire, nous fournissons des lecteurs USB de carte à puce sans contact compatible avec la norme PC/SC.




En fonction du lecteur et de votre système d'exploitation, il sera nécessaire d'installer les drivers compatibles pour la bonne détection du lecteur.

Ci-dessous les liens de téléchargement des drivers de quelques lecteurs :

Lecteur ACR 122 :
http://www.acs.com.hk/index.php?pid=drivers&id=ACR122U
Lecteur ACR 128 :
http://www.acs.com.hk/index.php?pid=drivers&id=ACR128
Lecteur Omnikey5321 :
http://www.hidglobal.com/driverDownloads.php?techCat=19&prod_id=171#

L'envoi de commandes se fera à partir d'une application développée par le CITC-EuraRFID disponible en téléchargement à l'adresse suivante :

http://www.citc-eurarfid.com/ali/alucard.zip

Le fichier téléchargé est un fichier compressé dans lequel se trouve une application « Alucard.exe ». Extraire cette application sur le bureau et double-cliquez dessus. Cette application ne nécessite pas d'installation préalable.
Connectez le lecteur au port USB de votre PC et lancez l'application, le ou les lecteurs connectés doivent apparaître dans la « *liste des lecteurs compatibles pc/sc* ».

Le standard pc/sc permet d'envoyer des commandes « haut niveau » à une carte en s'affranchissant des aspects propriétaires de chaque lecteur.

I. Echange de commandes APDU avec une Mifare Classic 1K.

Nous allons, dans cette partie du TP, échanger des commandes APDU[3] avec une carte sans contact de type Mifare 1K (cliquez sur le lien pour consulter la fiche technique).

Les commandes APDU sont des commandes « haut niveau » à destination de la carte. Ainsi, on observe toujours le schéma de communication suivant entre une carte et un terminal : une commande APDU suivie d'une réponse APDU.
Les tables suivantes illustrent respectivement des formats de commande APDU et de réponse APDU.

Commande APDU						
Champs obligatoires				*Champs conditionnels*		
CLA	INS	P1	P2	LC	Données	Le

Réponse APDU		
Champs conditionnels	*Champs obligatoires*	
Données	SW1	SW2

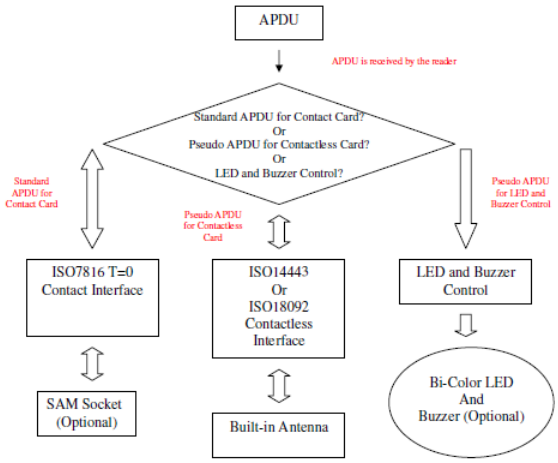
CLA: Octet de « Classe » utilisé pour identifier l'application

INS: Octet « d'Instruction » utilisé pour définir l'action

P1-P2 : Octets de "Paramètres" : données complémentaires au type d'action.

Lc : Nombre d'octets dans le champ de données

Le : Nombre d'octets attendu dans le champ de données de la réponse



[3] APDU : Application Protocol Data Unit

Identification de la carte :

Q1 : Donnez l'identifiant (UID) de la carte, sa taille en octet et la signification des deux derniers octets affichés dans la réponse.
La commande APDU permettant d'obtenir cette information est :

Commande	Class	INS	P1	P2	Le
Get Data	0xFF	0xCA	0x00	0x00	0x00

Réponse :

Afin d'accéder à la mémoire de la carte Mifare 1K, une authentification par blocs est nécessaire. La mémoire de la Mifare 1K est constituée de 16 secteurs de 4 blocs (cf. Annexe 1). Pour accéder au contenu d'un bloc, il faut au préalable s'authentifier sur ce bloc. Une fois un bloc authentifié, il n'est plus nécessaire de s'authentifier sur les blocs du même secteur.

MIFARE 1K Memory Map.

Sectors (Total 16 sectors. Each sector consists of 4 consecutive blocks)	Data Blocks (3 blocks, 16 bytes per block)	Trailer Block (1 block, 16 bytes)	
Sector 0	0x00 ~ 0x02	0x03	1K Bytes
Sector 1	0x04 ~ 0x06	0x07	
..			
Sector 14	0x38 ~ 0x3A	0x3B	
Sector 15	0x3C ~ 0x3E	0x3F	

Figure 1 : Organisation de la mémoire de la MIFARE 1K

Authentification :

Chargement de la clé dans la mémoire du lecteur.

Pour s'authentifier, il faut charger dans un premier temps la clé d'authentification dans la mémoire du lecteur. Deux types de clés sont utilisées : la clé A et la clé B.

Commande	Class	INS	P1	P2	Lc	Data In
Load Authentication Key	0xFF	0x82	Key Structure	Key Number	0x06	Auth Key (6 octets)

Key Structure : Type de mémoire (volatile ou non volatile) où la clé sera chargée dans le lecteur : 0x00 ou 0x20

Key Number : 0x00~0x01 : Emplacement de la clé, la clé est effacée lors de la déconnexion du lecteur.

Auth Key: Clé d'authentification. Généralement les clés par défaut sont A0 A1 A2 A3 A4 A5 ou FF FF FF FF FF FF.

Ci-dessous la liste des clés généralement utilisées par défaut :

- ✓ FF FF FF FF FF FF
- ✓ A0 A1 A2 A3 A4 A5
- ✓ D3 F7 D3 F7 D3 F7
- ✓ B0 B1 B2 B3 B4 B5
- ✓ 4D 3A 99 C3 51 DD
- ✓ 1A 98 2C 7E 45 9A
- ✓ AA BB CC DD EE FF
- ✓ 00 00 00 00 00 00

L'opération est validée par la réponse : 0x90 0x00

Difficultés rencontrées :

Authentification d'un bloc.

Ci-dessous, la commande permettant d'authentifier un bloc (et le secteur correspondant) de la Mifare 1K. Cette commande utilise la ou les clés chargées précédemment dans le lecteur pour procéder à l'authentification.

Commande	Class	INS	P1	P2	Lc	Data In
Authentication	0xFF	0x86	0x00	0x00	0x05	Données d'authentification

Données d'authentification :

Octet 0	Octet 1	Octet 2	Octet 3	Octet 4
0x01	0x00	Block Number	Key type	Key Number

Block Number : L'adresse du bloc sur lequel l'authentification est réalisée, (ce n'est pas l'adresse du secteur mais bien celui du bloc).

Key Type :

- ✓ 0x60 : La clé utilisée est une clé A
- ✓ 0x61 : La clé utilisée est une clé B

Key Number: Emplacement de la clé (cf. Load Authentication Key).

Si la réponse est différente de 0x90 0x00, l'opération ne s'est pas réalisée avec succès.

Une fois le bloc authentifié, il est possible de lire les blocs du secteur correspondant par le biais de la commande « Read Binary Blocks ».

Lecture & Ecriture d'un bloc.

Commande	Class	INS	P1	P2	Le
Read Binary Blocks	0xFF	0XB0	0x00	Block Number	Number of bytes to read

Block Number : (1 octet) c'est le bloc que l'on souhaite lire.
Number of bytes to read : (1 octet) Nombre d'octets à lire à partir du bloc sélectionné (maximum 16 octets).

Q2 : Lire les quatre premiers secteurs de la carte Mifare et remplir le tableau ci-dessous :

Mifare Classic 1K	
Secteur 1	
Secteur 2	
Secteur 3	
Secteur 4	

Q3 : Analyser les données du tableau :

Afin d'écrire un bloc de données dans la carte (le bloc de données doit être authentifié avec la clé en écriture), nous utilisons la commande « Update Binary Blocks »

Commande	Class	INS	P1	P2	Lc	Data In
Update Binary Blocks	0xFF	0XD6	0x00	Block Number	Number of bytes to update	Block Data

Block Number : 1 octet, le bloc que nous souhaitons écrire
Number of bytes to update: 1 octet, nombre d'octets à écrire:

- ✓ 16 octets pour une Mifare 1K
- ✓ 4 octets pour une Mifare Ultralight

Q4 : Ecrire le bloc 2 avec les données suivantes : 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F. Vérifier par la suite en lisant le bloc, que les données sont correctement inscrites.

Difficultés rencontrées :

II. Réalisation d'un Smart Poster NFC à partir d'une Mifare Ultralight.

La technologie NFC est une technologie de communication sans-fil à courte portée et à haute fréquence, permettant la communication entre des périphériques jusqu'à une distance d'environ 10 cm. Elle intègre la norme ISO/IEC 14 443 et le protocole Felica (protocole sans contact développé par Sony et populaire en Asie). Nous devrions retrouver un bon nombre de terminaux mobiles équipés de la technologie NFC prochainement, ce qui engendrera une accélération dans la mise en place des services mobiles sans contact, notamment avec les « smart posters » qui vont prochainement envahir le mobilier urbain.

Nous allons dans cette partie, à partir d'une Mifare Ultralight (512 bits), réaliser un Smart Poster en accord avec les spécifications NFC. Au passage d'un terminal NFC à proximité du Smart Poster, celui-ci déclenchera une action (requête d'appel vocal, SMS, lancer une URL…).

Nous travaillerons sur une carte de type Mifare Ultralight. Ci-dessous l'organisation de la mémoire de la puce.

Byte Number	0	1	2	3	Page
Serial Number	SN0	SN1	SN2	BCC0	0
Serial Number	SN3	SN4	SN5	SN6	1
Internal / Lock	BCC1	Internal	Lock0	Lock1	2
OTP	OTP0	OTP1	OTP2	OTP3	3
Data read/write	Data0	Data1	Data2	Data3	4
Data read/write	Data4	Data5	Data6	Data7	5
Data read/write	Data8	Data9	Data10	Data11	6
Data read/write	Data12	Data13	Data14	Data15	7
Data read/write	Data16	Data17	Data18	Data19	8
Data read/write	Data20	Data21	Data22	Data23	9
Data read/write	Data24	Data25	Data26	Data27	10
Data read/write	Data28	Data29	Data30	Data31	11
Data read/write	Data32	Data33	Data34	Data35	12
Data read/write	Data36	Data37	Data38	Data39	13
Data read/write	Data40	Data41	Data42	Data43	14
Data read/write	Data44	Data45	Data46	Data47	15

Figure 2 : Organisation de la mémoire de la MIFARE Ultralight

Pour lire les blocs mémoires de la Mifare Ultralight, il n'y a pas d'authentification à réaliser.

Q5 : Réutiliser les commandes de lecture et d'écriture vues précédemment pour lire l'ensemble des blocs mémoires de la carte.
(Valider le résultat avec la fonction Ultralight Explorer de l'application).

Pour réaliser une application sur un support NFC, il faut encapsuler les données dans le format NDEF. Le format NDEF (NFC Data Exchange Format) est un format de données commun pour les communications entre équipement NFC (NFC Forum).

Ainsi lors de la lecture de la mémoire du support sans contact, le NDEF va nous permettre d'identifier le support comme un support NFC.

Q6 : Pour ce faire, utiliser la commande écriture pour :

- ✓ Affecter la valeur 0xE1 au bloc OTP0
- ✓ Affecter la valeur 0x10 au bloc OTP1
- ✓ Affecter la valeur 0x06 au bloc OTP2
- ✓ Affecter la valeur 0x00 au bloc OTP3

Vérifier avec la commande lecture.

La valeur 0XE1 indique que des données NDEF sont présentes dans la mémoire.
La valeur 0x10 donne une indication sur la version supportée, ici version 1.0.
La valeur 0x06 indique la taille de la mémoire utile (48 octets dans notre cas)
La valeur 0x00 informe sur les conditions d'accès en écriture et en lecture.

Maintenant que le tag est « configuré » NFC, nous allons y inscrire les données.

Nous allons commencer par écrire 0x03 dans le bloc Data0 : Cette valeur indique que la mémoire n'est pas vide et qu'un message NDEF est présent dans la mémoire, il s'agit d'un drapeau indiquant le début du message.

De la même manière, un drapeau est utilisé pour indiquer la fin du message, il s'agit de la valeur 0xFE qu'il faudra inscrire à la fin de notre message.

Nous allons programmer une requête d'appel dans le tag NFC. Au passage d'un terminal NFC à proximité, l'action d'appeler le numéro programmé se déclenchera.

Ci-dessous un tableau avec les valeurs à inscrire pour réaliser notre Smart Poster :

Secteur 4	0X03 : début du message	Taille du message[4]	0xD1 : NDEF header[5]	0x01 : Taille du *Record Name*
Secteur 5	Taille des données[6]	0x55 (U)[7]: *Record Name*	0x?? : Code URI correspondant au tel:	Data : numéro de telephone
Secteur 6	0x??	0x??	0x??	0x??
---	0x??	0x??	0x??	0x??
Secteur n	0x??	FE		

Q7: A partir de la liste des codes URI et de la table ASCII (Cf. Annexes), compléter le tableau précédent.

Encoder la mémoire de la Mifare Ultralight avec les données de ce tableau et vérifier le bon fonctionnement de votre tag avec un terminal NFC.

Difficultés rencontrées :

[4] Taille des données **en hexadécimal** entre le début du message (0xD1) inclus et la fin du message (0xFE) non inclus.
[5] TNF = 0x01 (Well Known Type). SR=1, MB=1, ME=1
[6] Taille des données **en hexadécimal** entre Record Name non inclus et la fin des données (ici 0xFE) non inclus -> Taille(URI +Tel).
[7] 0x55 0x55 correspond dans la table ASCII à la lettre U. U représente le type URI (Uniform Resource Identifier).

contactless technologies • EuraRFID

Annexe 1 : Liste des codes URI

Hex	Protocol	Hex	Protocol
0x00	N/A. No prepending is done, and the URI field contains the unabridged URI.	0x13	urn:
0x01	http://www.	0x14	pop:
0x02	https://www.	0x15	sip:
0x03	http://	0x16	sips:
0x04	https://	0x17	tftp:
0x05	tel:	0x18	btspp://
0x06	mailto:	0x19	btl2cap://
0x07	ftp://anonymous:anonymous@	0x1A	btgoep://
0x08	ftp://ftp.	0x1B	tcpobex://
0x09	ftps://	0x1C	irdaobex://
0x0A	sftp://	0x1D	file://
0x0B	smb://	0x1E	urn:epc:id:
0x0C	nfs://	0x1F	urn:epc:tag:
0x0D	ftp://	0x20	urn:epc:pat:
0x0E	dav://	0x21	urn:epc:raw:
0x0F	news:	0x22	urn:epc:
0x10	telnet://	0x23	urn:nfc:
0x11	imap:	0x24..0xFF	RFU
0x12	rtsp://		

contactless technologies • EuraRFID

Annexe 2 : Correspondance avec les caractères ASCII

Dec	Hex	Graph.	Dec	Hex	Graph.	Dec	Hex	Graph.
32	20	(blank)	64	40	@	96	60	`
33	21	!	65	41	A	97	61	a
34	22	"	66	42	B	98	62	b
35	23	#	67	43	C	99	63	c
36	24	$	68	44	D	100	64	d
37	25	%	69	45	E	101	65	e
38	26	&	70	46	F	102	66	f
39	27	’	71	47	G	103	67	g
40	28	(	72	48	H	104	68	h
41	29	)	73	49	I	105	69	i
42	2A	*	74	4A	J	106	6A	j
43	2B	+	75	4B	K	107	6B	k
44	2C	,	76	4C	L	108	6C	l
45	2D	-	77	4D	M	109	6D	m
46	2E	.	78	4E	N	110	6E	n
47	2F	/	79	4F	O	111	6F	o
48	30	0	80	50	P	112	70	p
49	31	1	81	51	Q	113	71	q
50	32	2	82	52	R	114	72	r
51	33	3	83	53	S	115	73	s
52	34	4	84	54	T	116	74	t
53	35	5	85	55	U	117	75	u
54	36	6	86	56	V	118	76	v
55	37	7	87	57	W	119	77	w
56	38	8	88	58	X	120	78	x
57	39	9	89	59	Y	121	79	y
58	3A	:	90	5A	Z	122	7A	z
59	3B	;	91	5B	[	123	7B	{
60	3C	<	92	5C	\	124	7C	|
61	3D	=	93	5D	]	125	7D	}
62	3E	>	94	5E	^	126	7E	~
63	3F	?	95	5F	_			