Uso Seguro con RFID

Javier Rodriguez (@emb0scad0)
Github: https://githubs.com/pinguytaz
Web: www.pinguytaz.net

WORLD.PARTY2K23

Objetivos

- RFID y su seguridad
- Tipos de tarjetas RFID uso
- Seguridad y soluciones
  - Clonación
  - Proteccion de datos (almacenamiento)
- Ejemplo practico almacenamiento

Tipos de tarjetas

- Baja Frecuencia 120-150 Khz
  - EM4x1y acceso e identificación
  - T5577 (Configurable, RW, 363 bits, Password)
- Alta Frecuencia 13,56 MHz (encontramos NFC)
  - MiFare Classic y Classic CL2 (RW, operaciones, 1-4K, Password)
  - NTAG21x
  - MiFare Ultralight, Desfire, Plus
- Largo alcance UHF
- Anticolision
- Tamaños y formatos

EM4x0y (125kHz)

- EM410x
  - Solo lectura
  - UID
- EM4x05
  - Lectura Escritura
  - UID
  - 64 bytes de EEPROM

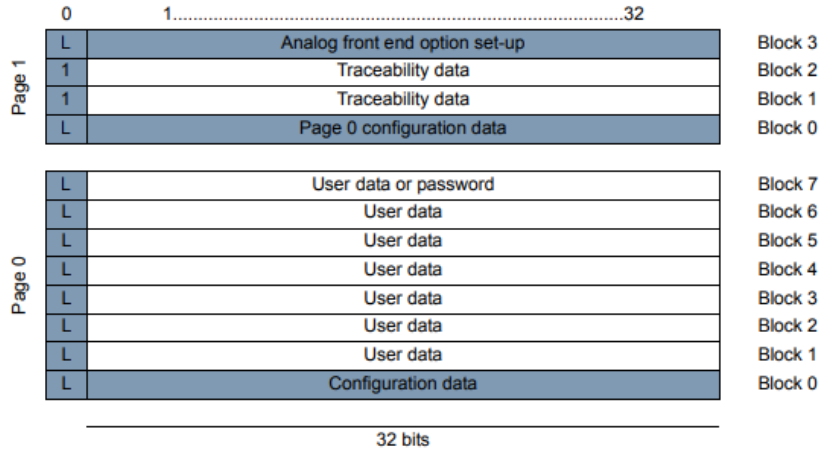
Addr. (dec)	Description	Type	B_0,. ..,b_{31}
0	Chip Type, Res Cap Customer code/ User free	RW	ct_0 - Ct_{31}
1	UID number	RA	uid_0 - uid_{31}
2	Password	WO	ps_0 - ps_{31}
3	User free	RW	us_0 - us_{31}
4	Configuration word	RW	co_0 - co_{31}
5	User free	RW	us_0 - us_{31}
6	User free	RW	us_0 - us_{31}
7	User free	RW	us_0 - us_{31}
8	User free	RW	us_0 - us_{31}
9	User free	RW	us_0 - us_{31}
10	User free	RW	us_0 - us_{31}
11	User free	RW	us_0 - us_{31}
12	User free	RW	us_0 - us_{31}
13	User free	RW	us_0 - us_{31}
14	Protection word 1	RP	pr_0 - pr_{31}
15	Protection word 2	RP	pr_0 - pr_{31}

Table 5

T5577 (Emulador)

- Frecuencia configurables
- 363 bits (11 bloques)
  - PWD B7-B0(b28)
- Configuracion B0 ASK,RF/32..(EM4100)
  - 00148041 (Con PWD)
  - 00148051 (Sin PWD)

Figure 4-2. Memory Map



Tipos

- EM4100 (00148041 / 00148051)
- HID (00107060 / 00107071)
- BLANK 00088040 / 000880F0

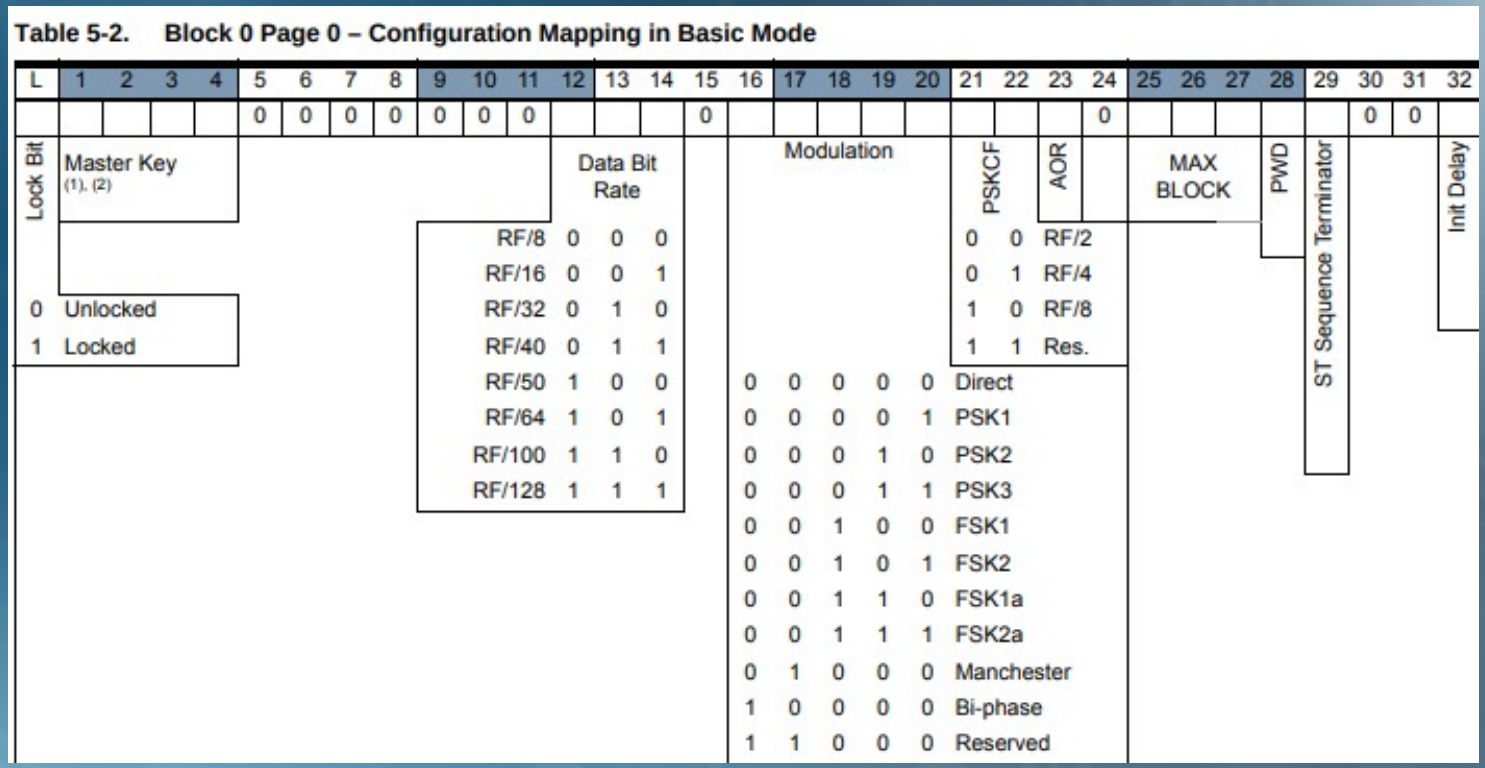


Table 5-2. Block 0 Page 0 – Configuration Mapping in Basic Mode

L	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
					0	0	0	0	0	0	0				0									0							0	0	

Lock Bit

0 Unlocked

1 Locked

Master Key (1), (2)

Data Bit Rate

RF/8	0	0	0
RF/16	0	0	1
RF/32	0	1	0
RF/40	0	1	1
RF/50	1	0	0
RF/64	1	0	1
RF/100	1	1	0
RF/128	1	1	1

Modulation

0	0	0	0	0	Direct
0	0	0	0	1	PSK1
0	0	0	1	0	PSK2
0	0	0	1	1	PSK3
0	0	1	0	0	FSK1
0	0	1	0	1	FSK2
0	0	1	1	0	FSK1a
0	0	1	1	1	FSK2a
0	1	0	0	0	Manchester
1	0	0	0	0	Bi-phase
1	1	0	0	0	Reserved

PSKCF

0	0	RF/2
0	1	RF/4
1	0	RF/8
1	1	Res.

AOR

MAX BLOCK

PWD

ST Sequence Terminator

Init Delay

NTAG21x

- Frecuencia 13,56 MHz
- UID 7bytes
- Anticolision
- x= 3(144B), 5(496B) y 6(872B)
- Configuración restricciones
- Uso NDEF (vcard, url, fichero)

NTAG213

Page Adr		Byte number within a page				
Dec	Hex	0	1	2	3	Description
0	0h	serial number				Manufacturer data and static lock bytes
1	1h	serial number				
2	2h	serial number	internal	lock bytes	lock bytes	
3	3h	Capability Container (CC)				Capability Container
4	4h	user memory				User memory pages
5	5h					
...	...					
38	26 h					
39	27 h					
40	28 h	dynamic lock bytes			RFUI	Dynamic lock bytes
41	29 h	CFG 0				Configuration pages
42	2Ah	CFG 1				
43	2Bh	PWD				
44	2Ch	PACK		RFUI		

aaa-008087

Fig 5. Memory organization NTAG213

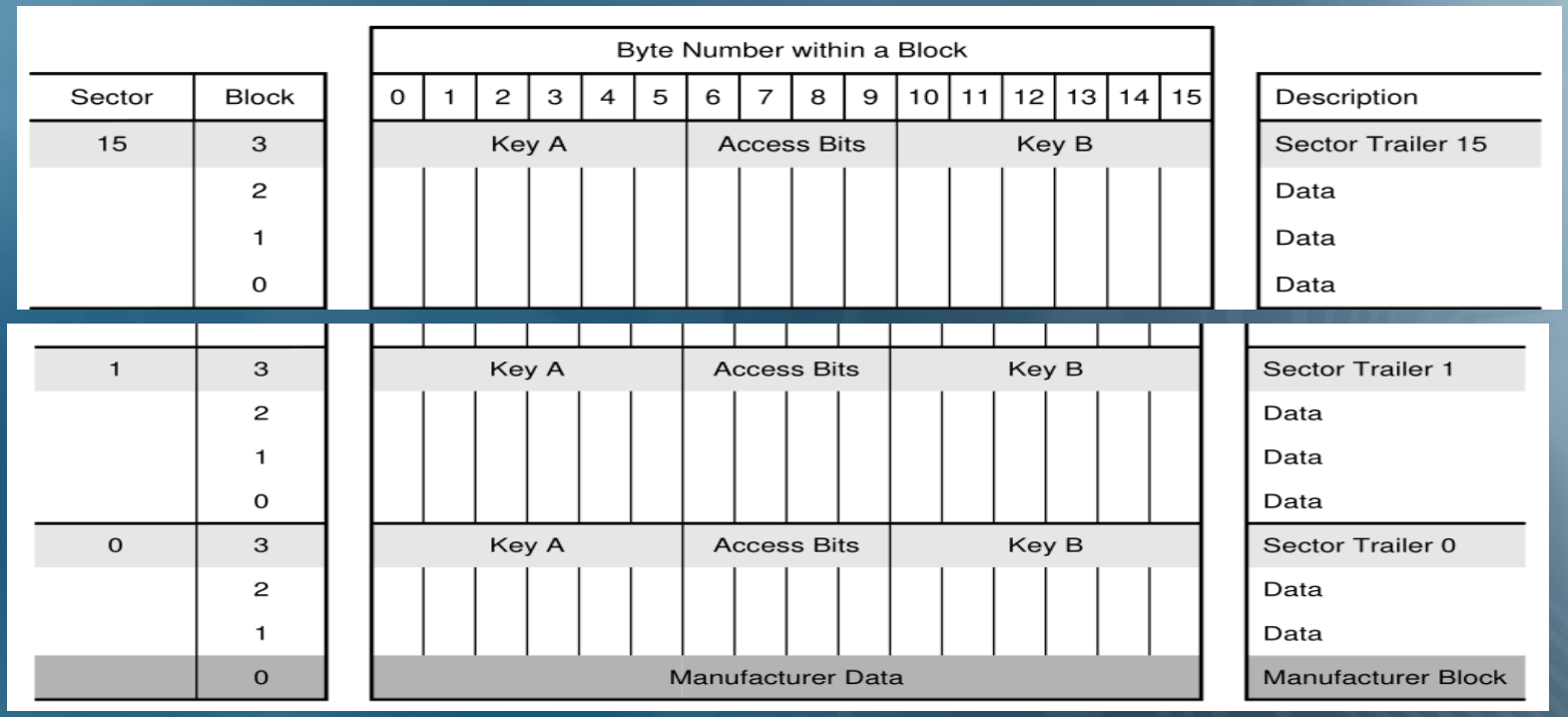
Clonación

- Evitar usar solo el identificador unico
- Usar un segundo factor de autentificación.
  - Pin
  - Biometria
- Cambiar datos de autentificacion tarjeta-Servidor.
- Calculos tiempos de inicialización
  - EM410x Solo lectura poco retraso
  - T5577 programable se toman mayor tiempo.

MiFare Classic 1K (13,56 MHz)

- UID
- 1KB (16 sectores de 4 bloques)
- Anticolisión
- Autentificacion por sector y tipo acceso.
- Operaciones en memoria
  - Lectura/escritura
  - Incrementos
  - Decrementos
  - Copiados

Classic 1K Organización



Sector	Block	Byte Number within a Block 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	Description
15	3	Key A | Access Bits | Key B	Sector Trailer 15
	2		Data
	1		Data
	0		Data
1	3	Key A | Access Bits | Key B	Sector Trailer 1
	2		Data
	1		Data
	0		Data
0	3	Key A | Access Bits | Key B	Sector Trailer 0
	2		Data
	1		Data
	0	Manufacturer Data	Manufacturer Block

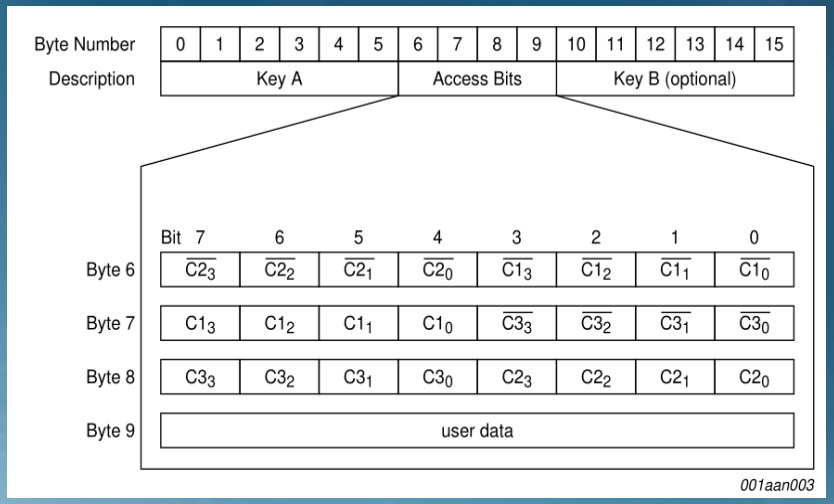
UID 7 bytes CL2 ATQA(0044) SAK(08)

NUID 4 bytes CL1 ATQA(0004) SAK(08)

Classic 1K Trailer

Byte Number	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Description	Key A						Access Bits				Key B (optional)					

- Trailer (bloque 3)
  - Clave A y B (6 bytes) FFFFFFFFFFFF
  - Tipo Acceso (4 Bits) FF078069



Access bits			Access condition for						Remark
			KEYA		Access bits		KEYB		
C1	C2	C3	read	write	read	write	read	write	
0	0	0	never	key A	key A	never	key A	key A	Key B may be read[1]
0	1	0	never	never	key A	never	key A	never	Key B may be read[1]
1	0	0	never	key B	key A|B	never	never	key B	
1	1	0	never	never	key A|B	never	never	never	
0	0	1	never	key A	key A	key A	key A	key A	Key B may be read, transport configuration[1]
0	1	1	never	key B	key A|B	key B	never	key B	
1	0	1	never	never	key A|B	key B	never	never	
1	1	1	never	never	key A|B	never	never	never	

Ejemplo

- Descripción
- Elección tipo de tarjeta
- Lectores
- Librerías, lenguajes
- Estructura de datos

Descripción

- Almacén en tarjeta RFID
- Acceso password
- Registros de claves
  - Clave-Valor (tarjeta-pin)

Elección Tarjeta

- Lectura Movil 13,56 MHz
  - Mifare Classic 1K
- Seguridad ¿? la ponemos nosotros
- Solo 752 Bytes
  - Trailes 16B*16Sectores = 256
  - Fabricante 16 Bytes
  - 1024 – 272 = 752Bytes para nosotros

Entorno desarrollo

- App Móvil el objetivo
- PN-532 (PoC uso en PC)
  - I2C
  - SPI
  - Serial (conexión PC por USB)
- LibNFC
  - https://github.com/nfc-tools/libnfc
- C

Estructura datos

- Ocultación
  - Posición aleatoria de inicio
    - Clave[0]
  - Hash-MD5 trastocado
    - Len & 15
    - [0] & 15
  - Encriptación / Desencriptación
    - XOR Clave-Password

Estructura datos

- PWD (MD5T)
- Numero registros
- Registros
  - Clave (MD5T) [16]
  - Longitud [1]
  - Valor[20]

- 17 PWD
- 37 Registro
  19 Registros

Referencias

- T5577

  https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-9187-RFID-ATA5577C_Datasheet.pdf
- MiFare Classic

  https://www.nxp.com/docs/en/data-sheet/MF1S50YYX_V1.pdf
- LibNFC (Libreria programación NFC)

  https://github.com/nfc-tools/libnfc

  http://www.libnfc.org/api/index.html
- Librerias Seguridad

  https://www.openssl.org/

PWD camuflado

Camuflado en aleatorios

Sector: 1 Bloque 0: B1892F69C027C8FBC7DABD321A7B1CA5
Sector: 1 Bloque 1: 9C6ACD8F4BBEA27E42E511F27AF1DF9B
Sector: 1 Bloque 2: E45D01E758E53AD862EC15803164ABE0
Sector: 2 Bloque 0: B2E0153FB5D9FDF091E973C1CCA2F3B7
Sector: 2 Bloque 1: 94B5102FCEA50646793A95E131F114B9
Sector: 2 Bloque 2: 7CFA28A6DDC8319671CD2D71786F507D
Sector: 3 Bloque 0: A48DE41F024A19A92881B1494B0023D0
Sector: 3 Bloque 1: AE1BFC8CBD7099F91A8C0C1A6E9F2E01
Sector: 3 Bloque 2: 5FA37F1B05867448DEFBB021A9F209DB
Sector: 4 Bloque 0: 53296B2FC0FC9186B0311130FE7EAD65
Sector: 4 Bloque 1: 8B8AADFC74FDA70033B83C88C37042B3
Sector: 4 Bloque 2: B4010B4CCBDF589B27DA2988C5759F07
Sector: 5 Bloque 0: AC168EC4672895E36D404B4BB911B438
Sector: 5 Bloque 1: 81C663EACA552FE9733A98AAB1CF869A
Sector: 5 Bloque 2: E5E9C987A5E6C2E1E33D0339DDA8D86D

Sector: 1 Bloque 0: 00000000000000000000000000000000
Sector: 1 Bloque 1: 00000000000000000000000000000000
Sector: 1 Bloque 2: 00000000000000000000000000000000
Sector: 2 Bloque 0: 00000000000000000000000000000000
Sector: 2 Bloque 1: 00000000000000000000000000000000
Sector: 2 Bloque 2: 0000000000000000000000006F507D
Sector: 3 Bloque 0: A48DE41F024A19A92881B1494B000000
Sector: 3 Bloque 1: 00000000000000000000000000000000
Sector: 3 Bloque 2: 00000000000000000000000000000000
Sector: 4 Bloque 0: 00000000000000000000000000000000
Sector: 4 Bloque 1: 00000000000000000000000000000000
Sector: 4 Bloque 2: 00000000000000000000000000000000
Sector: 5 Bloque 0: 00000000000000000000000000000000
Sector: 5 Bloque 1: 00000000000000000000000000000000
Sector: 5 Bloque 2: 00000000000000000000000000000000

Clave: 6f507de48da4 1f024a19a92881b1494b

6f507da48de4 1f024a19a92881b1494b

MD5 Trastocado

Inserciones

VISA PindeVisa

Sector: 2 Bloque 2: 00000000000000000000000006F507D
Sector: 3 Bloque 0: A48DE41F024A19A92881B1494B0193D2
Sector: 3 Bloque 1: 07A5A30A548F404AE593385A7B640906
Sector: 3 Bloque 2: 203D25263A0805040000000000000000

mail polo@pop.com

Sector: 2 Bloque 2: 00000000000000000000000006F507D
Sector: 3 Bloque 0: A48DE41F024A19A92881B1494B0293D2
Sector: 3 Bloque 1: 07A5A30A548F404AE593385A7B640906
Sector: 3 Bloque 2: 203D25263A0805040000000000000000
Sector: 4 Bloque 0: 000000B83A886A6F437CCD9AC15473FD
Sector: 4 Bloque 1: 5C17880C1D0E0503031C0E064B20030C

polo@pop.com LaclavedelMAIL

Sector: 2 Bloque 2: 00000000000000000000000006F507D
Sector: 3 Bloque 0: A48DE41F024A19A92881B1494B0393D2
Sector: 3 Bloque 1: 07A5A30A548F404AE593385A7B640906
Sector: 3 Bloque 2: 203D25263A0805040000000000000000
Sector: 4 Bloque 0: 000000B83A886A6F437CCD9AC15473FD
Sector: 4 Bloque 1: 5C17880C1D0E0503031C0E064B20030C
Sector: 4 Bloque 2: 00000000000000005259FC201F5AA1EF
Sector: 5 Bloque 0: 831E551705033A550E3C0E0F0321060A
Sector: 5 Bloque 1: 144B0F222C0A20000000000000000000

Ejecutando programa

```
11
12  void md5T(char valor[], unsigned char res[16])
13  {
14      EVP_MD_CTX *context = EVP_MD_CTX_new();
15      const EVP_MD* md = EVP_md5();
16      unsigned char md_value[EVP_MAX_MD_SIZE];
17      unsigned int  md_len;
18
19      /*  Lo primero es calcular el MD5 correcto */
20      EVP_DigestInit_ex2(context, md, NULL);
21      EVP_DigestUpdate(context, valor, strlen(valor));
22      EVP_DigestFinal_ex(context, md_value, &md_len);
23      EVP_MD_CTX_free(context);
24
25      for (unsigned int i = 0 ; i < md_len ; ++i)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
Sector: 15 Bloque 2: 4F8295648C94F1F6C59D505611FBA8A4
┌─[1] javier@delfin:/media/Datos1/PRJs/Charlas/RFID_Hack27/SRC
└─> ./Clav_NFC.elf Clave -v mail
Visualizamos el registro mail
Utiliza la version de libnfc 1.8.0
Resultado polo@pop.com
DEBUG: Volcado del contenido
Sector: 1 Bloque 0: 07E995B18BC92A095138B8880CB37ABE
Sector: 1 Bloque 1: 72206753933A8B0DD8EE49E900598333
Sector: 1 Bloque 2: E2651813AFF6B5B0A0E85DDDF01CECDC
Sector: 2 Bloque 0: CDDF1D87783BA8FBACEE57B746355341
Sector: 2 Bloque 1: CE134FE717E7D3033665C2DE33C2BEC5
Sector: 2 Bloque 2: FA20ABD4D2BF90DB374EB5EA676F507D
Sector: 3 Bloque 0: A48DE41F024A19A92881B1494B0393D2
```