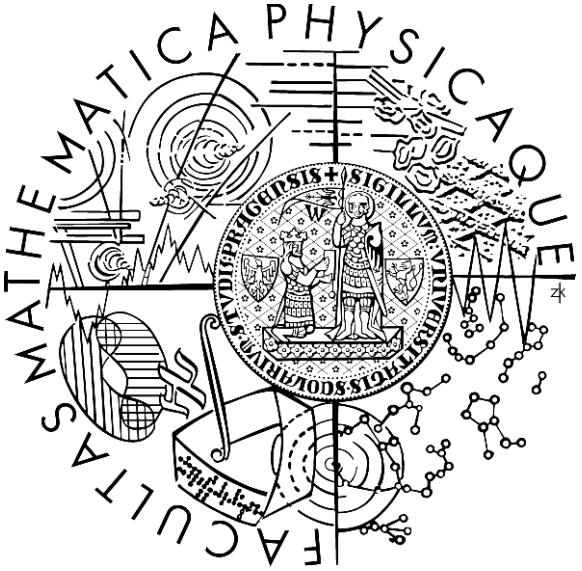
Univerzita Karlova v Praze

Matematicko-fyzikální fakulta

BAKALÁŘSKÁ PRÁCE



Jan Kučera

Courtoisův útok na MIFARE

Katedra algebry

Vedoucí bakalářské práce: doc. RNDr. Jiří Tůma DrSc.

Studijní program: Matematika, matematické metody informační bezpečnosti.

2010

Děkuji doc. Jiřímu Tůmovi za odborné vedení bakalářské práce a ing. Tomášovi Rosovi za konzultace v oblasti elektrotechnické. Rovněž mé díky patří autorům Karstenu Nohlovi, Flaviu Garciovi a Nicolasovi Courtoisovi za trpělivost a ochotu zodpovědět některé mé dotazy ohledně jejich práce.

Prohlašuji, že jsem svou bakalářskou práci napsal(a) samostatně a výhradně s použitím citovaných pramenů. Souhlasím se zapůjčováním práce.

V Praze dne 27.5.2010 Jan Kučera

1 Obsah

Název práce: Courtoisův útok na MIFARE

Autor: Jan Kučera

Katedra (ústav): Katedra algebry

Vedoucí bakalářské práce: doc. RNDr. Jiří Tůma DrSc.

e-mail vedoucího: tuma@karlin.mff.cuni.cz

Abstrakt: Když v roce 1948 H. Stockmann přišel s nápadem využívat ke komunikaci odražený signál, ještě nevěděl, bude-li to k něčemu dobré. O padesát let později je svět zaplaven bezkontaktními identifikačními zařízeními. Jsou ale bezpečné? Práce čtenáře seznamuje s problematikou RFID zařízení a jejich použitím a bezpečností, popisuje algoritmus Crypto1 používaný v produktech MIFARE a shrnuje dosavadní útoky na tento algoritmus. Cílem práce je nedávno publikované útoky nastudovat a pokusit se je implementovat. Implementace algoritmu Crypto1 včetně popsaného útoku je součástí přiloženého CD.

Klíčová slova: MIFARE, RFID, Crypto1, ISO 14443

Title: MIFARE attack by Courtois

Author: Jan Kučera

Department: Department of Algebra

Supervisor: doc. RNDr. Jiří Tůma DrSc.

Supervisor's e-mail address: tuma@karlin.mff.cuni.cz

Abstract: When H. Stockmann came in 1948 with an idea to use a reflected power for communication purposes, he had no idea whether it would be any good. Fifty years later and the world is flooded with contactless identification devices. But are they safe enough? This work explains RFID basics, its use and security problems, describes the Crypto1 algorithm, which is being used in MIFARE products, and summes attacks known so far. The aim of the work is to study recently published attacks and try to implement them. An implementation of the Crpyto1 algorithm including the described attack is available on the enclosed CD.

Keywords: MIFARE, RFID, Crypto1, ISO 14443

2 RFID

Pojmem RFID (Radio-frequency identification) se označuje identifikace pomocí rádiových vln. Jedná se tedy o obdobu systému čárových kódů, ve kterém identifikace probíhá pomocí optického snímání. V této kapitole se seznámíme se základními pojmy a potřebným technickým pozadím v oblasti RFID.

2.1 Úvod do problematiky

Počátky RFID sahají až do období druhé světové války, kdy bylo potřeba na dálku identifikovat vlastní letecké i ostatní jednotky. IFF (identification, friend or foe) tehdy vyžadovala vyslání rádiových impulsů identifikující stanicí, které následně letadlo detekovalo a jako odpověď vyslalo svůj kód přidělený před odletem. S drobnými obměnami se tento systém používá v armádě dodnes.

V roce 1948 publikoval Harry Stockmann článek s myšlenkou přenášet data k vysílači modifikací nebo manipulací vysílaného signálu. Uvedl i několik praktických pokusů, ale nebyl si ještě jist, zda má tento nápad nějaké praktické uplatnění [**Sto48**]. Bylo ještě třeba vynalézt transistor, integrovaný obvod, mikroprocesor. V padesátých letech si D. B. Harris patentoval metodu, jak přijímač napájet pouze pomocí vysílaných vln, bez potřeby jakéhokoliv dalšího externího zdroje energie [**Har52**]. Zbývalo tyto výsledky dát dohromady, což se na začátku šedesátých let podařilo R. F. Harringtonovi a mezi první komerční aplikace o pár let později patřila ochrana zboží proti zlodějům. Zájemce o podrobnější historii může nahlédnout například do [**Lan05**].

Dnes se RFID používá téměř ve všech oblastech – kromě již zmíněné ochrany v obchodních řetězcích také jako jízdenky ve veřejné dopravě (u nás např. OpenCard), v systému mýtného, v logistice ke sledování zásilek, kontejnerů nebo zavazadel, ke sledování zvířat, v přístupových systémech do budov (včetně průkazů ISIC/ITIC), v knihovnách k evidenci knih nebo vyhledávání dalších informací (čtenářské recenze, ceny), nebo také v kreditních kartách či elektronických pasech.

Systém tedy sestává z vysílače – terminálu a přijímače – transpondéru, jenž spolu komunikují pomocí rádiových, tj. elektromagnetických vln. Transpondéry se dělí na aktivní, které mají vlastní zdroj nápajení (baterii apod.), a pasivní, které jsou napájeny energií ze zmíněných elektromagnetických vln, přesněji řečeno vložením do elektromagnetického pole terminálu. Tato energie však není nijak závratně velká, a tak jsou pasivní transpondéry značně omezeny i co se výpočetního výkonu týče.

Aktivní transpondéry si mohou dovolit vysílat informace samostatně a jsou tak schopny komunikovat na relativně velké vzdálenosti. Oproti tomu pasivní transpondéry, kterými se budeme zabývat v této práci, nemají dost energie na to, aby mohly samy vysílat, ale protože se napájí z vysílaných vln, mohou ovlivňovat, kolik energie spotřebovávají, mohou tzv. měnit zátěž. Terminál, jelikož se o jedná o jeho zdroj energie, to samozřejmě pozná, čehož může transpondér vyžít ke komunikaci – změnou zátěže podle určitých pravidel (kódováním) tak předává informace terminálu.

V každém případě se ke komunikaci používají rádiové vlny, které může každý odposlechnout, aniž by k tomu potřeboval nákladné či nápadné zařízení. Například AM pásmo pro běžné rozhlasové stanice začíná na frekvenci 148 kHz a jedna z používaných frekvencí pro RFID technologie je 125 kHz. Proto je na místě věnovat pozornost bezpečnosti, zajistit jak utajení dat během komunikace, tak ochranu úložiště v transpondéru. Z těchto důvodů byla zavedena některá opatření, jako šifrování a autentizace. Protože však šifrování snižuje kapacitu transpondérů a autentizace snižuje rychlost čtení dat, je třeba zvolit vhodný kompromis.

Možné útoky jsou: [**Har10**]

- Napodobování, zahrnuje přehrání a klonování dat a nahrání nebezpečného kódu. Útočník může zachytit komunikaci a znovu ji přehrát terminálu, aniž by znal její obsah, nebo může naklonovat obsah jednoho transpondéru na druhý. Nebezpečný kód, pokud by se útočníkovi podařilo nahrát jej do terminálu a zajistit jeho spuštění, by teoreticky mohl poškodit celý systém. Tento útok se však nepovažuje za příliš reálný, zejména z důvodu velmi malého množství přenášených dat.
- Získání dat z transpondéru, buď neautorizovaným čtením jeho paměti nebo odposlechem a dekódováním řádné komunikace mezi transpondérem a terminálem; případně neoprávněná manipulace – vymazání dat pro znehodnocení transpondéru nebo úprava (např. ke změně ceny výrobku).
- Vyřazení systému z provozu, tzv. denial of service attack. Například použitím tak velkého množství karet (nebo speciálně navržených karet), že je terminál zahlcen a není schopen řádného provozu. Samozřejmě také připadá v úvahu mechanické či elektronické poškození terminálu nebo transpondéru.

Organizce ISO, na základě práce RFID Expert Group založené asociací AIM (Association for Automatic Identification and Mobility), vydala v roce 2009 pravidla a doporučení ISO/IEC TR 24729-4 pro zajištění bezpečnosti v oblasti RFID. Stále však nejsou k dispozici standardy k zajištění jednotného, bezpečného a interoperabilního systému.

Tato práce shrnuje a popisuje známé útoky v oblasti získávání dat z transpondéru, které mohou případně vést k jeho naklonování. Specializuje se na implementaci RFID jednoho konkrétního výrobce, NXP Semiconductors (dříve součást společnosti Philips), nazvanou MIFARE Classic, která je na trhu již od roku 1995 [**Boo08**].

2.2 Základní terminologie

transpondér Zařízení s pamětí na data, většinou v držení uživatelem v podobě tzv. „karty“. Další běžnou formou je např. nálepka na zboží, známka pro zvířata apod. Většinou se napájí elektromagnetickým polem. V anglické literatuře se označuje jako **tag**.

terminál Zařízení určené ke komunikaci s transpondérem, tzv. „čtečka“. Vyžaduje externí napájení a je zapojena do dalšího systému, kterému zpřístupňuje data uložená na transpondéru. Generuje elektromagnetické pole pro jeho napájení. V anglické literatuře **reader** nebo terminal.

emulátor Zařízení, které terminál považuje za transpondér. Lze jím simulovat chování, které běžný transpondér neumožňuje. Například nastavení libovolného výrobního čísla, generování konkrétní výzvy při autentizaci, zasílání neplatných paritních bitů nebo CRC dat. Mívá vlastní napájení a může tak disponovat znatelně větším výpočetním výkonem než transpondér. V roli emulátoru může být i stolní počítač.

PKE Public Key Encryption. Algoritmy založené na veřejném klíči (RSA, eliptické křivky atd.)

BCC Block Check Character, délka 1 bajt, xor předchozích 4 bajtů UID.

CRC Cyclic Redundancy Check k detekci chyb během přenosu, délka 16 bitů (2 bajty), výpočet viz. [**ISO013**].

SEL Select. Příkaz terminálu k vybrání transpondéru. Terminál může komunikovat nejvýše s jedním transpondérem současně, a ten musí pro tento účel vybrat. Příkaz má 8 bitů (1 bajt), za ním následuje NVB a část nebo celé UID transpondéru, který má být vybrán. Pokud je UID celé, následuje ještě kontrolní bajt BCC a CRC data. Zasílá se vždy otevřeně.

NVB Number of Valid Bits. Počet platných bitů UID, které následují, součást příkazu na vybrání transpondéru. Délka 8 bitů – první čtyři bity udávají počet platných bajtů, zbylé čtyři bity počet dalších platných bitů.

SAK Select Acknowledge. Odpověď transpondéru potvrzující jeho úspěšné vybrání terminálem. Odpověď má 8 bitů (1 bajt) a je doplněna o CRC data. Zasílána vždy otevřeně.

ATQA	Answer to Request. Odpověď transpondéru ohlašující jeho přítomnost v dosahu terminálu. Délka 16 bitů (2 bajty), posílá se vždy otevřeně.
REQA	Request Command. Výzva pro všechny transpondéry (typu A) v dosahu, aby ohlásily svou přítomnost. Délka 8 bitů (1 bajt), zasílá se vždy otevřeně.
UID	Unique Identifier. Výrobní číslo o délce 32 bitů, tj. 4 bajtů. Originální transpondéry neumožňují toto číslo měnit. Přenáší se vždy otevřeně.
PICC	Proximity Integrated Circuit Card – transpondér.
PCD	Proximity Coupling Device – terminál.
LF	low frequency
HF	high frequency
UHF	ultra high frequency
bps	bits per second, 1 kbps = 1000 bps
B	bajt, 1 GB = 1024 MB, 1 MB = 1024 kB, 1 kB= 1024 B
S, START	„bit“ s významem „začátek komunikace“. Definice viz kapitola 2.5.
E, END	„bit“ s významem „konec komunikace“. Definice viz kapitola 2.5.

2.3 Notace a konvence

Použité značení:

$a \oplus b$	exklusivní součet, $a \oplus b \coloneqq (a + b)\ mod\ \mathbb{F}_2$
$a \wedge b$	a a zároveň b, $a \wedge b \coloneqq (a \cdot b)\ mod\ \mathbb{F}_2$
$a \vee b$	a nebo b nebo obojí, $a \vee b \coloneqq (a \oplus b + a \wedge b)\ mod\ \mathbb{F}_2$
$\bar{x}$	negace, $\bar{x} = x \oplus 1$
$\{x\}$	zašifrovaná hodnota x
$u \doteq v$	u je zaokrouhleno na v
$u \cong v$	u je přibližně rovno v
$\Pr[P]$	pravděpodobnost jevu P
0x	uvození hexadecimálního zápisu čísla

Paritní bity bývají v záznamech komunikace v citované literatuře ([**KHG08**], [**Gar09**], [**Cou09**]) označovány vykřičníkem za hodnotou, ke které se paritní kontrola vztahuje, pokud nejsou správně spočteny; v opačném případě nejsou uváděny vůbec. Jelikož se, jak je uvedeno dále v práci, paritní bity nepočítají z přenášených hodnot, nemá toto značení v záznamech komunikace žádný význam a působí spíše rušivě. Proto v této práci uvádím záznam paritních bitů jejich samotnou hodnotou, nikoliv posouzením, zda jsou z nějakého pohledu spočteny správně.

Zdaleka největším problémem je pak pořadí bitů při uvádění vícebitových hodnot, které bohužel ovlivňují i popis a schéma použité šifry, a značně ztěžují případné implementace. Většina autorů se pro jistotu o této problematice ve svých pracech nezmiňuje. Běžnou vývojářskou praxí je uvádět nejvýznamnější bit vlevo, bit nejméně významný vpravo, a číslovat je od nejméně významného bitu, tj. např.

$$b = (b_7, \dots, b_0) = (0,1,0,1,1,1,0,1) = 2^6 + 2^4 + 2^3 + 2^2 + 2^0 = 93 = 0\text{x}5\text{D}.$$

Dle normy [**ISO013**] se bity také číslují od nejméně významného bitu, jen se začíná jedničkou, tedy $b = (b_8, \dots b_1)$. Vzduchem se ovšem přenáší v opačném pořadí – nejméně významný bit jako první. Záznam z komunikace tedy vypadá takto:

START	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8	END

Při uvádění bajtů se zachovává význam bitů, takže $[1,0,1,1,1,0,1,0] = 0\text{x}5\text{D}$.

V citované literatuře se ovšem bity číslují od nejvíce významného bitu, tedy

$$(b_0, \dots, b_7) = (1,0,1,1,1,0,1,0) = 2^7 + 2^5 + 2^4 + 2^3 + 2^1 = 0\text{xBA}.$$

Situace se stává ještě složitější, uvážíme-li hodnoty o více než osmi bitech. Obdobně jako výše, v literatuře se číslují bity po jednom za sebou od nejvýznamnějšího po nejméně významný, v praxi naopak. Dle normy se ovšem hodnota rozdělí na bajty (po osmi bitech), ty se zasílají ve stejném pořadí jako se zapisují (tj. nejvýznamnější bajt se odešle první), avšak pro každý bajt se odešle nejméně významný bit jako první. Máme-li například hodnotu 0x124E, odešle se jako:

START	0x12								0x4E								END
	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8	
	0	1	0	0	1	0	0	0	0	1	1	1	0	0	1	0	

(v citované literatuře budou takto přenesená data uvedena jako 0x7248)

V této práci nezbývá, než se držet zápisu používaného v citované literatuře z důvodu srozumitelnosti s ohledem na dostupnou literaturu. Jen je potřeba mít při implementaci na vědomí, že posun posuvného registru, tak jak jsou registry zde a jinde zavedeny, ve skutečnosti znamená rozdělit registr na bajty, obrátit pořadí bajtů, v každém bajtu

obrátit pořadí bitů, provést posun dle definice a výsledek zase stejným postupem převést zpět.

Z těchto důvodů se snažím v práci alespoň vyvarovat hexadecimálním zápisům vícebajtových čísel; přesto se však domnívám, že popsat a definovat celou šifru tak, aby přímo odpovídala implementaci, by bylo záslužným činem v této oblasti.

2.4 Průmyslové standardy

Problematikou RFID se zabývá celá řada standardů. Jedno ze základních rozdělení, dle frekvenčního rozsahu užívaného ke komunikaci, je v tabulce 2.4-1.

pásmo	rozsah	použití	standardy
LF	< 135 kHz	identifikace zvířat, přístupové systémy, klíčky do zapalování	ISO/IEC 18000-2
HF	13,553-13,567 MHz	smart card, přístupové systémy, platební karty, občanské průkazy, pasy, jízdenky	ISO/IEC 18000-3, ISO/IEC 14443, ISO/IEC 15963, ISO/IEC 18092, ISO/IEC 21481
UHF	433 MHz	aktivní transpondéry pro nákladní dopravu a vojenskou logistiku v USA a zemích NATO	ISO/IEC 18000-7
	840-960 MHz	sledování zboží po jednotlivých kusech, ochrana proti krádežím, zavazadla v letectví, doprava	ISO/IEC 18000-6 ISO/IEC 29143
	2,45 GHz	správa položek	ISO/IEC 18000-4
		určování polohy v reálném čase	ISO/IEC 24730-2 ISO/IEC 24730-5

Tabulka 2.4- Frekvenční rozdělení RFID

V této práci se budeme zabývat systémem pracujícím na frekvenci $f_c = 13{,}56$ MHz dle normy ISO/IEC 14443, typ A.

Každý standard obvykle zajišťuje tyto vrstvy:

- *Fyzickou vrstvu*, která definuje požadované fyzikální vlastnosti komponent, rozměry, maximální výkony apod. V našem případě je toto předmětem normy ISO/IEC 14443-1.
- *Transportní vrstvu*, která definuje, jak se kódují jednotlivé bity do analogového signálu, stanovuje používané frekvence a přenosové rychlosti. V našem případě ISO/IEC 14443-2.
- *Antikolizní protokol,* který určuje, jak se transpondéry v elektromagnetickém poli detekují a identifikují a jak realizovat komunikaci v případě, že je v dosahu více transpondérů najednou. ISO/IEC 14443-3.

- *Aplikační vrstvu*, která definuje společné příkazy pro aplikace běžící na transpondérech (např. přečti hodnotu na dané adrese). ISO/IEC 14443-4.

Implementace MIFARE Classic není kompatibilní s ISO/IEC 14443-4, využívá proprietární příkazy, které jsou shrnuty v kapitole 4.4.

2.5 Transportní vrstva

Následuje stručné shrnutí informací obsažených v normě [**ISO012**].

2.5.1 Časování a přenosová rychlost

- Tolerance pro komunikační frekvenci jest ±7 kHz.
- Norma stanovuje několik možných přenosových rychlostí:
  - $f_c/128 = \frac{13\,560\,000}{128} = 105\,937{,}5$ bps $\cong$ 106 kbps
  - $f_c/64 = 211\,856$ bps
  - $f_c/32 = 423\,750$ bps
  - $f_c/16 = 847\,500$ bps

V některých citovaných zdrojích se uvádí množství dotazů na transpondér za sekundu, případně časová náročnost útoku, a ne vždy si zdroje odpovídají. To může být způsobeno jednak různou přenosovou rychlostí – nejběžnější je 106 kbps, vyšší rychlosti netriviálně zvyšují nároky na schopnosti hardware a výpočetní výkon (používají například i odlišné kódování bitů a modulaci [**ISO012**], tj. způsob, jakým se bity přenáší po analogovém signálu). Například v [**Gar09**] je u jednoho útoku uvedeno, že zhruba 1500 dotazů na transpondér trvá méně než jednu vteřinu (přitom [**Cou09**] uvádí 2 dotazy za vteřinu (s vypínáním elektromagnetického pole) – při osobní komunikací uvedl schopnost maximálně asi 10 dotazů za vteřinu za cenu ztráty přesnosti načasování, která je ovšem ve většině útoků kritická). Také je třeba mít na paměti, že ve všech útocích se tak dlouho zasílají náhodná data transpondéru, dokud neodpoví. Ovšem skutečnost, že transpondér na dotaz neodpoví, znamená, že se zablokuje a útočník musí takový stav nejdříve rozpoznat a v lepším případě kartu resetovat příkazem, v horším vypnout elektromagnetické pole, počkat, dokud se kondenzátory na transpondéru nevybijí (dle [**Gar09**] asi 30 μs), pak znovu pole zapnout, počkat až se stabilizuje a znovu transpondér detekovat a vybrat (viz. následující kapitola).

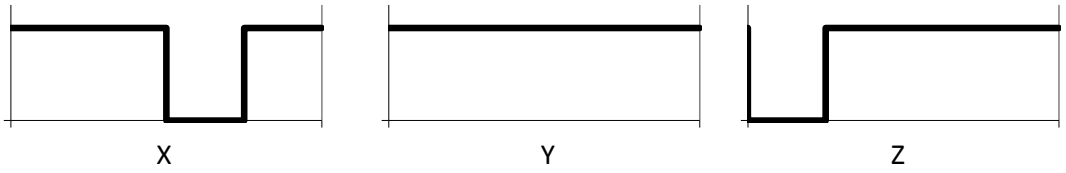
Řekněme, že máme v poli pouze jeden transpondér a z dřívější komunikace již známe jeho UID. Pak reset a vybrání znamená přenos minimálně 135 bitů, s pauzami při nejvyšší rychlosti celkem cca 340 μs (určeno normou). Pokus o špatnou autentizaci

(viz. kapitola 4.3) znamená přenos minimálně 112 bitů a pauzy celkem 340 µs. Při nejvyšší rychlosti trvá přenesení 1 bitu zhruba 1.2 µs, celkem nás tedy jeden neúspěšný pokus stojí v tom nejlepším případě 0,98 ms bez jakékoliv manipulace s elektromagnetickým polem, což se špičkovým hardwarem představuje teoretickou možnost přes 1020 pokusů za vteřinu na hraně normy jak pro terminál, tak pro transpondér. Uváděných 1500 dotazů není reálných.

2.5.2 Kódování bitů při přenosu

Následující kódování probíhá při běžné přenosové rychlosti 106 kbps, a je rovněž definováno v [**ISO012**].

Při komunikaci směrem z terminálu do transpondéru jsou definovány tři různé průběhy signálu, viz. obrázek 2.5-1.

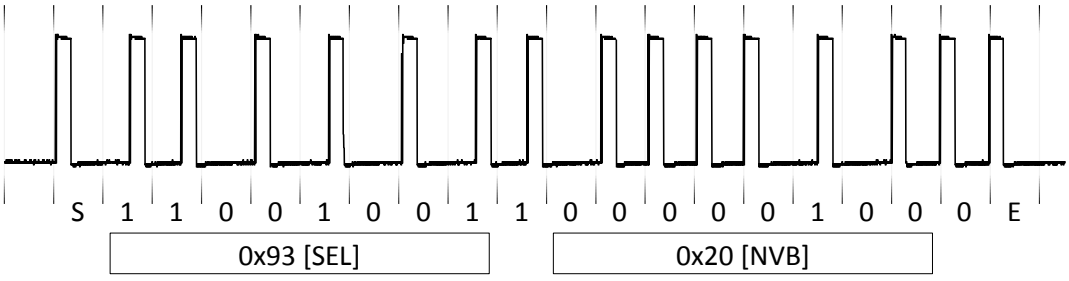


Obrázek 2.5- Kódování bitů z terminálu do transpondéru

Jednotlivé bity pak mají přiřazeny tyto průběhy:

bit	průběh
0	Z v případě, předchází-li tomuto bitu bit 0 nebo bit START; jinak Y
1	X
START	Z
END	jako bit 0 následovaný Y

Reálný záznam komunikace je pro ilustraci na obrázku 2.5-2.

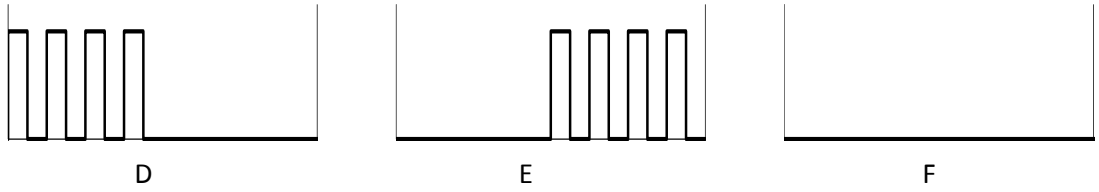


Obrázek 2.5- Záznam výběru transpondéru terminálem

Norma říká, že za každým odeslaným bajtem následuje lichý paritní bit. V původním znění není výslovně uvedeno, že se počítá z přenášeného bajtu (tj. tak, aby počet jedniček v každých přenesených devíti bitech byl lichý); tento zřejmý požadavek byl

doplněn v druhé revizi. Jak uvidíme dále, výpočet paritních bitů v implementaci MIFARE Classic se jí stal osudným.

Pro komunikaci směrem z transpondéru do terminálu se používá tzv. kódování Manchester. Stanovují se následující průběhy signálu:

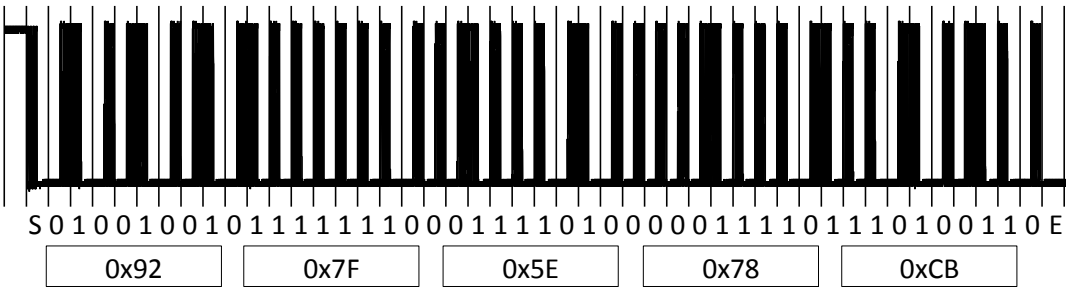


Obrázek 2.5- Kódování bitů z transpondéru do terminálu

Jednotlivé bity jsou pak definovány takto:

bit	průběh
0	E
1	D
START	D
END	F

A reálný záznam:



Obrázek 2.5- Záznam zaslání UID a BCC transpondérem

I tímto směrem se za každým bajtem zasílá lichý paritní bit.

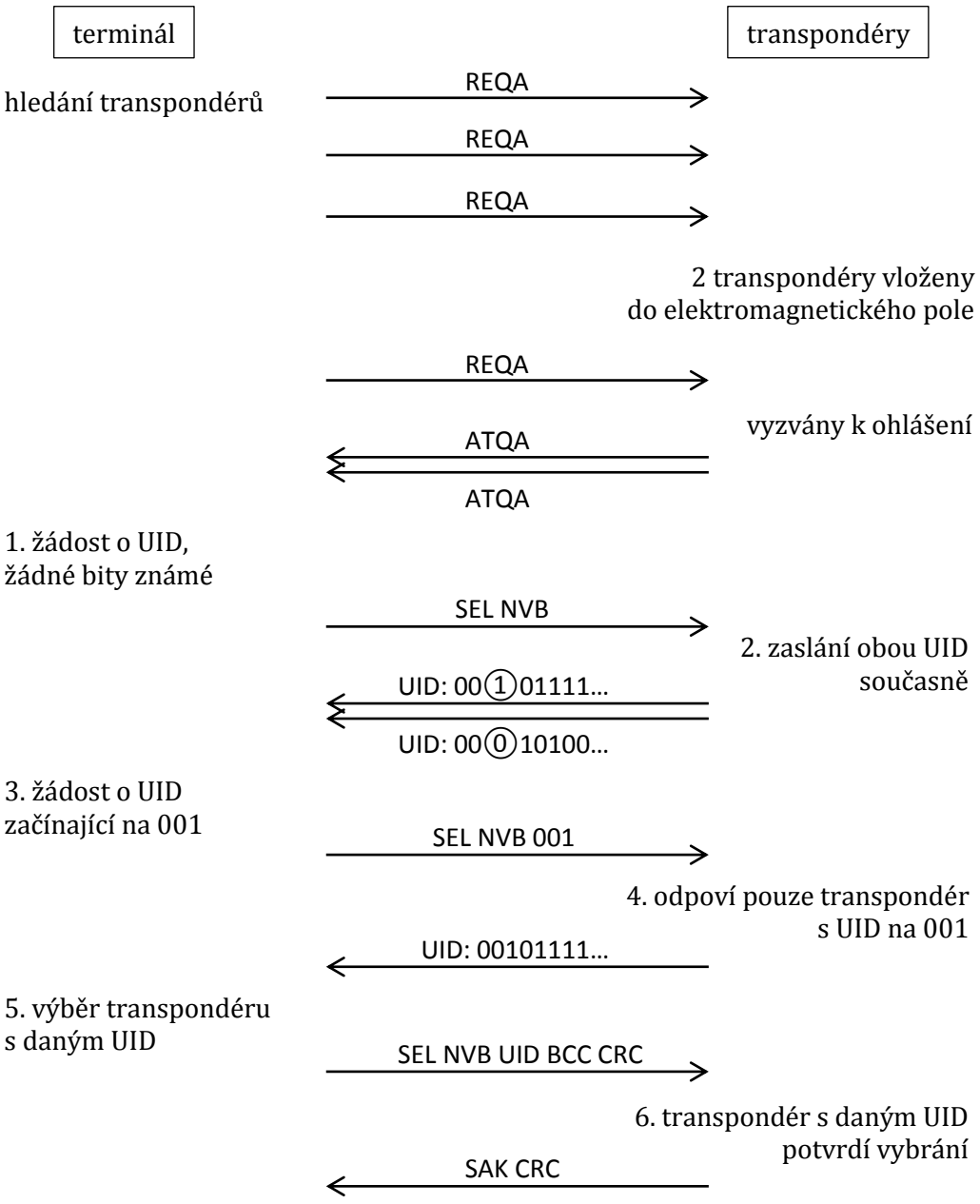
2.6 Detekce transpondéru a antikolizní protokol

Po vložení transpondéru do elektromagnetického pole terminálu transpondér vyčkává, dokud není terminálem vyzván ke svému ohlášení. Tím se zajistí, že nebude narušena případná probíhající komunikace terminálu s jiným transpondérem v dosahu.

Terminál neustále v pravidelných intervalech vysílá žádost o identifikaci transpondéru, označovanou jako REQA. Všechny transpondéry, které jsou v dosahu a vyčkávají na tuto výzvu, zašlou naráz odpověď ATQA. Přesný formát REQA a ATQA lze najít v [**ISO013**].

Jakmile terminál obdrží odpověď, začne tzv. antikolizní smyčku. Jejím úkolem je rozpoznat, kolik transpondérů je v dosahu, případně jaká mají výrobní čísla – UID.

Průběh smyčky je na obrázku 2.6-1. Popis a vysvětlení pojmů následuje.



Obrázek 2.6- Antikolizní protokol

Protokol probíhá následovně:

1. Terminál potřebuje k vybrání transpondéru znát jeho UID. Odešle tedy požadek na výběr transpondéru SEL s tím, že je mu známo prvních nula bitů UID.
2. Všechny transpondéry zašlou ve stejný okamžik své UID spolu s kontrolním bajtem BCC. Všimněme si na obrázku 2.5-3, že jedničce odpovídá modulace v první polovině „bitu“, nule modulace v druhé polovině. Pokud tedy terminál obdrží modulovaný signál po celé délce, znamená to, že přijal jedničku i nulu zároveň; říkáme, že nastala kolize.
3. Terminál si zvolí hodnotu bitu v místě kolize (typicky jedničku), a znovu odešle požadavek na výběr transpondéru s tím, že tentokrát je mu známo tolik bitů, kolik obdržel před kolizí plus zvolený bit (v našem příkladě jsou mu tedy známy 3 bity), a tyto bity odešle dohromady s požadavkem.
4. Tentokrát na požadavek reagují pouze transpondéry, jejichž (UID, BCC) začíná na zaslanou posloupnost bitů. Tyto transpondéry znovu ve stejný okamžik zašlou své (UID, BCC). Ostatní transpondéry se další komunikace již neúčastní, dokud jim nebude znovu zaslán požadavek na sdělení UID, kterému budou moci vyhovět.
5. Terminál opět prověří, zda při přenosu nedošlo ke kolizi. Pokud ano, pokračuje krokem 3. Pokud již ke kolizi nedošlo, obdržel kompletní UID jednoho transpondéru, a může jej vybrat. Znovu tedy odešle požadavek na výběr terminálu a uvede, že zná všech 40 bitů (UID a BCC). Ty připojí k požadavku a doplní dvěma bajty kontroly CRC.
6. Transpondér, jehož UID terminál zaslal, se přepne do stavu „vybráno“ a potvrdí přijetí příkazu odpovědí SAK doplněnou o CRC.

Dále pak již probíhá komunikace v rámci aplikační vrstvy, tj. systém s terminálem se může domlouvat s transpondérem pomocí protokolů, na kterých jsou domluveny.

Poznámka. V popisu uvažujeme pro jednoduchost pouze transpondéry mající UID o délce 32 bitů. Standard povoluje UID i o délkách 56 nebo 80 bitů, ale transpondéry MIFARE Classic nejsou ten případ, a tak je popis antikolize v takové situaci nad rámec této práce. Zájemce může najít příslušné informace i s příklady ve zmíněné normě [**ISO013**].

3 MIFARE

V této části se blíže seznámíme s transpondérem, na který jsou dále v práci popsány útoky. Transpondéry MIFARE Standard se za dobu své existence po světě velice rozšířily, od elektronického jízdného (např. časopis ISO Focus+ uvádí přes 10 milionů karet Oyster pro dopravu v Londýně k roku 2007), přes přístupové karty do různých budov (včetně např. Dánské vlády [**Boo08**]), až po studentské průkazy ISIC.

3.1 Rodina MIFARE a bezpečnostní algoritmy

V dnešní době nabízí výrobce několik různých typů transpondérů lišících se mimo jiné v použitých bezpečnostních algoritmech, viz. tabulka 3.1-1 poskládaná z informací na internetových stránkách výrobce [**http://nxp.com/**]. Pro úplnost obsahuje i informace o kompatibilitě aplikačního protokolu se standardem ISO/IEC 14443-4.

transpondér	Crypto1	3DES	AES	PKE	14443-4
MIFARE Ultralight					
MIFARE Ultralight C		●			
MIFARE Standard 1K	●				
MIFARE Standard 4K	●				
MIFARE Standard Mini	●				
MIFARE Plus	●		●		
MIFARE DESFire		●	●		●
MIFARE ProX	●	●		●	●
SmartMX	●	●	●	●	●

Tabulka 3.1- Přehled bezpečnostních algoritmů v transpondérech MIFARE

Odposlechem komunikace transpondéru s terminálem nebo vlastní antikolizní procedurou dle kapitoly 2.6 lze alespoň přibližně zjistit typ transpondéru z odpovědi SAK, případně ATQA. Tabulka odpovědí dle typu karet je v příloze B.

Útoky představené v následujících kapitolách jsou namířeny na transpondéry se šifrou Crypto1 a původním generátorem náhodných čísel, což jsou transpondéry typu MIFARE Standard 1K, 4K a Mini.

Transpondéry řady ProX a SmartMX jsou duální, tj. mají i kontaktní rozhraní pro komunikaci, a společně s řadou DESFire obsahují programovatelný mikroprocesor. Ostatní transpondéry mají pevně dané možnosti a funkčnost.

MIFARE Plus je řada vyvinutá jako přímá náhrada transpondérů Standard v reakci na publikované útoky. Obsahují odlišný náhodný generátor a jsou schopny šifrování pomocí AES. Některé z transpondérů byly certifikovány na úroveň EAL4+ Common Criteria [**http://www.commoncriteriaportal.org/**].

3.2 Struktura MIFARE Standard 1K

Transpondér typu MIFARE Standard 1K patří do rodiny MIFARE Classic a obsahuje paměť velikosti 1024 bajtů, rozdělenou do 16 sektorů po čtyřech blocích o 16 bajtech:

sektor	blok	bajt																popis
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
0	0	UID				c	data výrobce											blok výrobce
	1																	data
	2																	data
	3	klíč A						přístup				klíč B						hlavička 0
1	0																	data
	1																	data
	2																	data
	3	klíč A						přístup				klíč B						hlavička 1
⋮	⋮	⋮																⋮
15	0																	data
	1																	data
	2																	data
	3	klíč A						přístup				klíč B						hlavička 15

Obrázek 3.2- Struktura MIFARE Standard 1K

Vyjma nultého sektoru, který má v nultém bloku speciální informace pouze pro čtení, se každý sektor skládá ze 48 bajtů uživatelských dat a 16 konfiguračních bajtů, které kromě dvou klíčů obsahují i pravidla, jak k danému sektoru přistupovat. V případě, že druhý klíč není potřeba, je možné tento prostor rovněž využít pro uživatelská data.

Pro zajímavost z hlediska bezpečnosti uveďme princip pravidel pro přístup k jednotlivým sektorům. V hlavičce každého sektoru je vymezeno po třech bitech přístupových údajů pro každý blok příslušného sektoru. Pro datové bloky je jejich význam uveden v tabulce 3.2-1, pro blok hlavičky v tabulce 3.2-2. Blok výrobce je vždy pouze pro čtení a pravidly se neřídí.

x	y	z	čtení hodnoty	zápis hodnoty	zvýšení hodnoty o 1	snížení hodnoty o 1
0	0	0	A nebo B	A nebo B	A nebo B	A nebo B
0	0	1	A nebo B			A nebo B
0	1	0	A nebo B			
0	1	1	B	B		
1	0	0	A nebo B	B		
1	0	1	B			
1	1	0	A nebo B	B	B	A nebo B
1	1	1				

Tabulka 3.2- Klíče vyžadované při operacích s datovými bloky

x	y	z	čtení A	zápis A	čtení pravidel	zápis pravidel	čtení B	zápis B
0	0	0		A	A		A	A
0	0	1			A		A	
0	1	0		B	A nebo B			B
0	1	1		B	A nebo B	B		B
1	0	0		B	A nebo B			B
1	0	1			A nebo B	B		
1	1	0			A nebo B			
1	1	1			A nebo B			

Tabulka 3.2- Klíče vyžadované při operacích s blokem hlavičky

Operace bez vyplněného klíče nejsou povoleny. Z tabulky 3.2-2 vyplývá, že za žádných podmínek nelze z pouhého přístupu k transpondéru zjistit klíč A. Dále stojí za zmínku, že přístup k datovému bloku s pravidlem (001) umožňuje provoz transpondéru jako jednorázové karty s nabitým kreditem, s pravidlem (110) pak i s možností opakovaného dobíjení autoritou znající klíč B.

Operace zvýšení a snížení (a zbylé dvě zde neuvedené) používají speciální formát datového bloku a nejsou v této práci využity. Případný zájemce může najít další podrobnosti v dokumentaci výrobce.

Každý bit je uložen jednou neinvertovaný a jednou invertovaný, celkem 12 bitů rozmístěných dle obrázku 3.2-2 zabírá tedy 3 bajty. Při pokusu o zápis neplatné kombinace invertované a neinvertované hodnoty bitu se celý sektor nenávratně zablokuje.

	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
bajt 6	$\overline{y_3}$	$\overline{y_2}$	$\overline{y_1}$	$\overline{y_0}$	$\overline{x_3}$	$\overline{x_2}$	$\overline{x_1}$	$\overline{x_0}$
bajt 7	x_3	x_2	x_1	x_0	$\overline{z_3}$	$\overline{z_2}$	$\overline{z_1}$	$\overline{z_0}$
bajt 8	z_3	z_2	z_1	z_0	y_3	y_2	y_1	y_0

Obrázek 3.2- Rozložení přístupových bitů v hlavičce sektoru

Čtvrtý bajt pravidel (tj. devátý bajt hlavičky) je nevyužitý a lze jej použít pro data aplikace s tím, že pro něj platí přístupová pravidla jako pro blok hlavičky.

3.3 Tříprůchodové autentizační schéma

Po zapnutí transpondéru jeho vložením do elektromagnetického pole proběhne nejdříve antikolizní protokol tak, jak je popsán v kapitole 2.5. Připomeňme, že jeho úkolem je pouze vybrat transpondér ke komunikaci, a uplatní se tedy zejména je-li v dosahu více transpondérů najednou. Tento protokol probíhá bez jakéhokoliv šifrování a během něj transpondér musí nutně prozradit typ (např. MIFARE Standard 1K) a jedinečné výrobní číslo (UID).

Bohužel je třeba podotknout, že nemálo reálných aplikací v tomto bodě končí, a k identifikaci používají pouze nešifrované výrobní číslo transpondéru. V takovém případě stačí útočníkovi buď odchytit komunikaci transpondéru s terminálem, anebo rovnou iniciovat komunikaci s transpondérem, který je povinen své výrobní číslo prozradit čistě z organizačních důvodů. Na podvržení výrobního čísla pak není zapotřebí žádného zvláštního hardwarového ani softwarového vybavení.

V lepším případě je k identifikaci potřeba číslo uložené v některém ze sektorů transpondéru chráněného tajným klíčem. Přístup do takového sektoru pak probíhá dle následujícího autentizačního schématu:

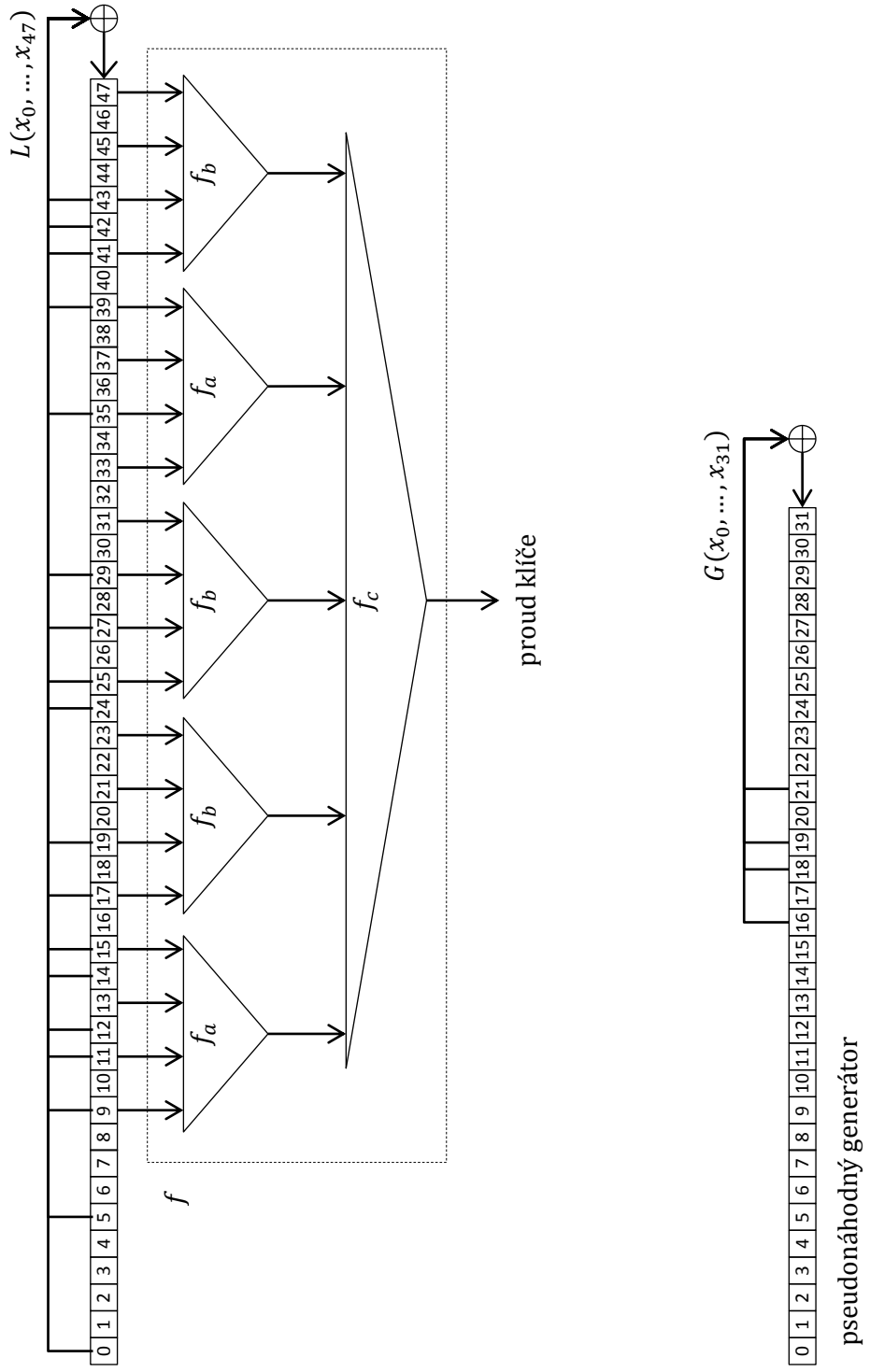
0. Předpokládá se, že terminál i transpondér nezávisle na sobě znají tajný klíč, pomocí kterého lze požadovaná data rozšifrovat.
1. Terminál zažádá o přístup do příslušného sektoru.
2. Transpondér si z daného sektoru přečte přístupové informace, zvolí číslo a zašle jej terminálu, aby na něm prokázal znalost klíče. (první průchod)
3. Terminál vrátí důkaz o znalosti klíče a sám si zvolí číslo, na kterém musí i transpondér prokázat svou znalost klíče. (druhý průchod)
4. Transpondér ověří odpověď (porovnáním se svým výpočtem) a vrátí důkaz o znalosti klíče. (třetí průchod)
5. Terminál ověří odpověď (porovnáním se svým vlastním výpočtem).

Tím je proces autentizace dokončen, a je-li vše v pořádku, terminál pokračuje např. příkazem pro čtení hodnoty z registru.

V případě MIFARE Classic se znalost klíče prokazuje inicializací vestavěné šifry tímto klíčem a následného zašifrování zaslaných čísel. Komunikace je od druhého bodu až do vypnutí transpondéru zašifrována. Podrobný popis a příklad viz. následující kapitoly.

4 Proudová šifra Crypto1

Bezpečnost informací uložených v paměti transpondérů zajišťuje algoritmus Crypto1. Jedná se o proudovou šifru s posuvným registrem o délce 48 bitů. Její základní schéma je na obrázku 4-1.



Obrázek 3.3- Základní schéma Crypto1

Registr má zpětnou lineární vazbu, dle [**Gar09**] s generujícím polynomem $x^{48} + x^{43} + x^{39} + x^{38} + x^{36} + x^{34} + x^{33} + x^{31} + x^{29} + x^{24} + x^{23} + x^{21} + x^{19} + x^{13} + x^9 + x^7 + x^6 + x^5 + 1$. Tamtéž je rovněž definováno:

Definice 4-. Funkce zpětné vazby L: $\mathbb{F}_2^{48} \rightarrow \mathbb{F}_2$ je definována předpisem

$$L(x_0, \dots, x_{47}) \coloneqq x_0 \oplus x_5 \oplus x_9 \oplus x_{10} \oplus x_{12} \oplus x_{14} \oplus x_{15} \oplus x_{17} \oplus x_{19} \oplus \\ \oplus x_{24} \oplus x_{25} \oplus x_{27} \oplus x_{29} \oplus x_{35} \oplus x_{39} \oplus x_{41} \oplus x_{42} \oplus x_{43}.$$

Každým taktem hodin na transpondéru se vygeneruje bit proudu klíče pomocí nelineární funkce f, a registr se o jeden bit posune.

4.1 Minimum z teorie booleovských funkcí

V dnešní době se v oboru kryptografie a kryptoanalýzy pracuje téměř výhradně s booleovskými funkcemi a jejich vlastnostmi, připomeňme tedy některé nejzákladnější pojmy, které budeme potřebovat.

Definice 4.1-. Booleovská funkce je funkce f: $\mathbb{F}_2^n \rightarrow \mathbb{F}_2^m$. Řekneme, že f je binární booleovská funkce právě když $m = 1$.

Definice 4.1-. Řekneme, že booleovská funkce f je lineární, pokud existuje $w \in \mathbb{F}_2^n$ tak, že $f(a) = w^T \cdot a$, tj. $f = w_1 a_1 \oplus w_2 a_2 \oplus \dots \oplus w_n a_n$.

Booleovské funkce zapisujeme buď tabulkou s výčtem hodnot, nebo předpisem pomocí operací logický součin (∧), logický součet (∨), exklusivní součet (⊕) a případně negace tak, jak jsou uvedeny v přehledu značení v kapitole 2.3.

Existuje několik speciálních tvarů zápisu booleovských funkcí. V technických oborech se pro návrhy zapojení obvodů využívá tzv. disjunktivní normální forma, v kryptografii naopak algebraická normální forma (ANF).

Definice 4.1-a. Řekneme, že booleovská funkce f je v algebraické normální formě, pokud píšeme

$$f(a) = w_0 \oplus \bigoplus_{1 \le i \le n} w_i a_i \oplus \bigoplus_{1 \le i < j \le n} w_{ij} a_i a_j \oplus \dots \oplus w_{12\dots n} a_1 a_2 \cdots a_n.$$

Alternativní definice, která dává i návod na vyjádření funkce v tomto tvaru jest:

Definice 4.1-3b. Buď $[f] = \big(f(0, \dots, 0, 0), f(0, \dots, 0, 1), \dots, f(1, \dots, 1, 1)\big)$ vektor hodnot funkce $f \in \mathbb{F}_2^n$ a nechť

$$A_0 = (1), \qquad A_n = \begin{pmatrix} A_{n-1} & A_{n-1} \\ 0 & A_{n-1} \end{pmatrix} \ \forall n \ge 1.$$

Pak vektor koeficientů ANF je definován jako

$$(\boldsymbol{w_0}, \dots, \boldsymbol{w_{12\dots n}}) \equiv [\boldsymbol{f}] \cdot \boldsymbol{A_n} \ (\mathbf{mod}\ \mathbf{2}).$$

Dále definujme několik vlastností booleovských funkcí.

Definice 4.1-. Řekneme, že binární booleovská funkce f je úplná, pokud pro každé $i = 1 \dots n$, existuje $(x_1, \dots, x_{i-1}, x_{i+1}, \dots x_n) \in \mathbb{F}_2^{n-1}$ tak, že

$$\boldsymbol{f}(\boldsymbol{x_1}, \dots \boldsymbol{x_{i-1}}, \mathbf{0}, \boldsymbol{x_{i+1}}, \dots, \boldsymbol{x_n}) \neq \boldsymbol{f}(\boldsymbol{x_1}, \dots \boldsymbol{x_{i-1}}, \mathbf{1}, \boldsymbol{x_{i+1}}, \dots, \boldsymbol{x_n});$$

jinými slovy, pokud závisí na všech vstupních proměnných.

Definice 4.1-. Řekneme, že binární booleovská funkce f je balancovaná, pokud

$$\left|\boldsymbol{f^{-1}}(\mathbf{0})\right| = \left|\boldsymbol{f^{-1}}(\mathbf{1})\right| = \boldsymbol{2^{n-1}};$$

jinými slovy, pokud nabývá obou hodnot přesně s poloviční pravděpodobností.

Definice 4.1-. Řekneme, že binární booleovská funkce f splňuje striktní propagační kritérium, pokud pro každé $(x_1, \dots, x_{i-1}, x_{i+1}, \dots x_n) \in \mathbb{F}_2^{n-1}$

$$\mathbf{Pr}[\boldsymbol{f}(\boldsymbol{x_1}, \dots, \boldsymbol{x_{i-1}}, \mathbf{0}, \boldsymbol{x_{i+1}}, \dots, \boldsymbol{x_n}) \neq \boldsymbol{f}(\boldsymbol{x_1}, \dots, \boldsymbol{x_{i-1}}, \mathbf{1}, \boldsymbol{x_{i+1}}, \dots, \boldsymbol{x_n})] = \frac{\mathbf{1}}{\mathbf{2}}$$

$$\forall \boldsymbol{i} = \mathbf{1} \dots \boldsymbol{n}, \ \ \forall \boldsymbol{j} = \mathbf{1} \dots \boldsymbol{n};$$

jinými slovy, pokud změna libovolného vstupního bitu změní výstupní hodnotu v právě polovině případů.

4.2 Výstupní nelineární funkce

Výstupní funkce je složená, dle [**Gar09**] definována předpisem

$$\begin{aligned} f(x_0 \dots x_{47}) \coloneqq f_c(&f_a(x_9, x_{11}, x_{13}, x_{15}),\\ &f_b(x_{17}, x_{19}, x_{21}, x_{23}),\\ &f_b(x_{25}, x_{27}, x_{29}, x_{31}),\\ &f_a(x_{33}, x_{35}, x_{37}, x_{39}),\\ &f_b(x_{41}, x_{43}, x_{45}, x_{47})) \end{aligned}$$

kde:

$$f_a(y_0, y_1, y_2, y_3) \coloneqq \big((y_0 \vee y_1) \oplus (y_0 \wedge y_3)\big) \oplus \Big(y_2 \wedge \big((y_0 \oplus y_1) \vee y_3\big)\Big)$$

$$f_b(y_0, y_1, y_2, y_3) \coloneqq \big((y_0 \wedge y_1) \vee y_2\big) \oplus \big((y_0 \oplus y_1) \wedge (y_2 \vee y_3)\big)$$

$$\begin{aligned} f_c(y_0, y_1, y_2, y_3, y_4) \coloneqq &\Big(y_0 \vee \big((y_1 \vee y_4) \wedge (y_3 \oplus y_4)\big)\Big)\\ &\oplus \Big(\big(y_0 \oplus (y_1 \wedge y_3)\big) \wedge \big((y_2 \oplus y_3) \vee (y_1 \wedge y_4)\big)\Big) \end{aligned}$$

Všechny tyto funkce jsou balancované:

Tvrzení 4.2-. Nechť $y_0, y_1, y_2, y_3 \in \mathbb{F}_2$ jsou náhodné s rovnoměrným rozdělením, na sobě nezávislé proměnné. Pak

$$\mathbf{Pr}[f_a(y_0, y_1, y_2, y_3) = 1] = \frac{1}{2}.$$

Důkaz.

$$\begin{aligned}
&\Pr[f_a(y_0, y_1, y_2, y_3) = 1] = \\
&= \Pr\left[\left((y_0 \vee y_1) \oplus (y_0 \wedge y_3)\right) \oplus \left(y_2 \wedge \left((y_0 \oplus y_1) \vee y_3\right)\right) = 1\right] = \\
&= \Pr\left[\left((y_0 \vee y_1) \oplus (y_0 \wedge y_3)\right) \neq \left(y_2 \wedge \left((y_0 \oplus y_1) \vee y_3\right)\right)\right] =: P_a
\end{aligned}$$

Dále máme pro y_0 a y_1:

y_0	y_1	P_a
0	0	$\Pr[0 \oplus 0 \neq y_2 \wedge y_3] = \Pr[1 = y_2 \wedge y_3] = \Pr[y_2 = 1] \cdot \Pr[y_3 = 1] = \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{4}$
0	1	$\Pr[1 \oplus 0 \neq y_2 \wedge 1] = \Pr[1 \neq y_2] = \Pr[y_2 = 0] = \frac{1}{2}$
1	0	$\Pr[1 \oplus y_3 \neq y_2 \wedge 1] = \Pr[y_3 = y_2] =$ $\Pr[y_3 = 0 \mid y_2 = 0] + \Pr[y_3 = 1 \mid y_2 = 1] = \Pr[y_3 = 0] \cdot \Pr[y_2 = 0] +$ $\Pr[y_3 = 1] \cdot \Pr[y_2 = 1] = \frac{1}{2} \cdot \frac{1}{2} + \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{4} + \frac{1}{4} = \frac{1}{2}$
1	1	$\Pr[1 \oplus y_3 \neq y_2 \wedge y_3] = \Pr[y_3 = y_2 \wedge y_3] = 1 - \Pr[y_3 = 1] \cdot \Pr[y_2 = 0] =$ $1 - \frac{1}{2} \cdot \frac{1}{2} = \frac{3}{4}$

Celkem tedy

$$P_a = \frac{1}{4} \cdot \frac{1}{4} + \frac{1}{2} \cdot \frac{1}{4} + \frac{1}{2} \cdot \frac{1}{4} + \frac{3}{4} \cdot \frac{1}{4} = \frac{1}{16} + \frac{2}{16} + \frac{2}{16} + \frac{3}{16} = \frac{8}{16} = \frac{1}{2}.$$

■

Tvrzení 4.2-. Nechť $y_0, y_1, y_2, y_3 \in \mathbb{F}_2$ jsou náhodné s rovnoměrným rozdělením, na sobě nezávislé proměnné. Pak

$$\mathbf{Pr}[f_b(y_0, y_1, y_2, y_3) = 1] = \frac{1}{2}.$$

Důkaz.

$$\begin{aligned}
&\Pr[f_b(y_0, y_1, y_2, y_3) = 1] = \\
&= \Pr\left[\left((y_0 \wedge y_1) \vee y_2\right) \oplus \left((y_0 \oplus y_1) \wedge (y_2 \vee y_3)\right) = 1\right] = \\
&= \Pr\left[\left((y_0 \wedge y_1) \vee y_2\right) \neq \left((y_0 \oplus y_1) \wedge (y_2 \vee y_3)\right)\right] =: P_b
\end{aligned}$$

Dále pro y_0, y_1 máme:

y_0	y_1	P_b
0	0	$\Pr[0 \vee y_2 \neq 0 \wedge (y_2 \vee y_3)] = \Pr[y_2 \neq 0] = \frac{1}{2}$
0	1	$\Pr[0 \vee y_2 \neq 1 \wedge (y_2 \vee y_3)] = \Pr[y_2 \neq y_2 \vee y_3] = \Pr[y_2 = 0] \cdot \Pr[y_3 = 1] = \frac{1}{4}$
1	0	
1	1	$\Pr[1 \vee y_2 \neq 0 \wedge (y_2 \vee y_3)] = \Pr[1 \neq 0] = 1$

Celkem tedy

$$P_b = \frac{1}{2} \cdot \frac{1}{4} + \frac{1}{4} \cdot \frac{1}{2} + 1 \cdot \frac{1}{4} = \frac{1}{8} + \frac{1}{8} + \frac{2}{8} = \frac{4}{8} = \frac{1}{2}.$$

■

Tvrzení 4.2-. Nechť $y_0, y_1, y_2, y_3, y_4 \in \mathbb{F}_2$ jsou náhodné s rovnoměrným rozdělením, na sobě nezávislé proměnné. Pak

$$\mathbf{Pr}[f_c(y_0, y_1, y_2, y_3, y_4) = 1] = \frac{1}{2}.$$

Důkaz.

$$\Pr[f_c(y_0, y_1, y_2, y_3, y_4) = 1] =$$
$$= \Pr\Big[\Big(y_0 \vee \big((y_1 \vee y_4) \wedge (y_3 \oplus y_4)\big)\Big) \oplus \Big(\big(y_0 \oplus (y_1 \wedge y_3)\big) \wedge \big((y_2 \oplus y_3) \vee (y_1 \wedge y_4)\big)\Big) = 1\Big] =$$
$$= \Pr\Big[\Big(y_0 \vee \big((y_1 \vee y_4) \wedge (y_3 \oplus y_4)\big)\Big) \neq \Big(\big(y_0 \oplus (y_1 \wedge y_3)\big) \wedge \big((y_2 \oplus y_3) \vee (y_1 \wedge y_4)\big)\Big)\Big] =: P_c$$

Podobným postupem jako v předchozích důkazech balancovanosti dostaneme:

y_0	y_3	y_4	P_c
0	0	0	$\Pr[0 \neq y_2] = \frac{1}{2}$
0	0	1	$\Pr[1 \neq y_2 \vee y_1] = \Pr[y_2 = 0] \cdot \Pr[y_1 = 0] = \frac{1}{4}$
0	1	0	$\Pr[0 \neq y_1 \wedge (y_2 \oplus 1)] = \Pr[y_1 = 1] \cdot \Pr[y_2 = 0] = \frac{1}{4}$
0	1	1	$\Pr\big[0 \neq y_1 \wedge \big((y_2 \oplus 1) \vee y_1\big)\big] = \Pr[y_1 = 1] = \frac{1}{2}$
1	0	0	$\Pr[1 \neq y_2] = \frac{1}{2}$
1	0	1	$\Pr[1 \neq y_2 \vee y_1] = \frac{1}{4}$
1	1	0	$\Pr[1 \neq (1 \oplus y_1) \wedge (y_2 \oplus 1)] = 1 - \Pr[y_1 = 0] \cdot \Pr[y_2 = 0] = \frac{3}{4}$
1	1	1	$\Pr\big[1 \neq (1 \oplus y_1) \wedge \big((y_2 \oplus 1) \wedge y_1\big)\big] = 1$

A celkem:

$$P_c = \frac{1}{8}\left(\frac{1}{2}+\frac{1}{4}+\frac{1}{4}+\frac{1}{2}+\frac{1}{4}+\frac{3}{4}+1\right) = \frac{1}{8}\cdot\frac{16}{4} = \frac{1}{2}$$

■

Důsledek 4.2-. Nechť $x_0, \dots, x_{47} \in \mathbb{F}_2$ jsou náhodné s rovnoměrným rozdělením, na sobě nezávislé proměnné. Pak výstupní nelineární funkce je balancovaná.

Důkaz. Výstupy funkcí f_a a f_b jsou na sobě nezávislé z předpokladu důsledku a definice složené funkce, a tedy i vstupní proměnné f_c jsou na sobě nezávislé. Dále z tvrzení 4.2-1 a 4.2-2 plyne, že jsou náhodné s rovnoměrným rozdělením, čímž jsou splněny předpoklady pro tvrzení 4.2-3, a složená funkce je rovněž balancovaná. ■

4.3 Náhodný generátor

Pro účely autentizace je v transpondérech generátor náhodných čísel ve formě dalšího, 32bitového posuvného registru s lineární zpětnou vazbou generovanou polynomem $x^{16} + x^{14} + x^{13} + x^{11} + 1$. Počáteční stav registru je 101010… [**Nohl, osobní komunikace**] .

Definice 4.3-. Funkce zpětné vazby generátoru G: $\mathbb{F}_2^{32} \to \mathbb{F}_2$ je definována jako

$$G(x_0, \dots, x_{31}) \coloneqq x_{16} \oplus x_{18} \oplus x_{19} \oplus x_{21}.$$

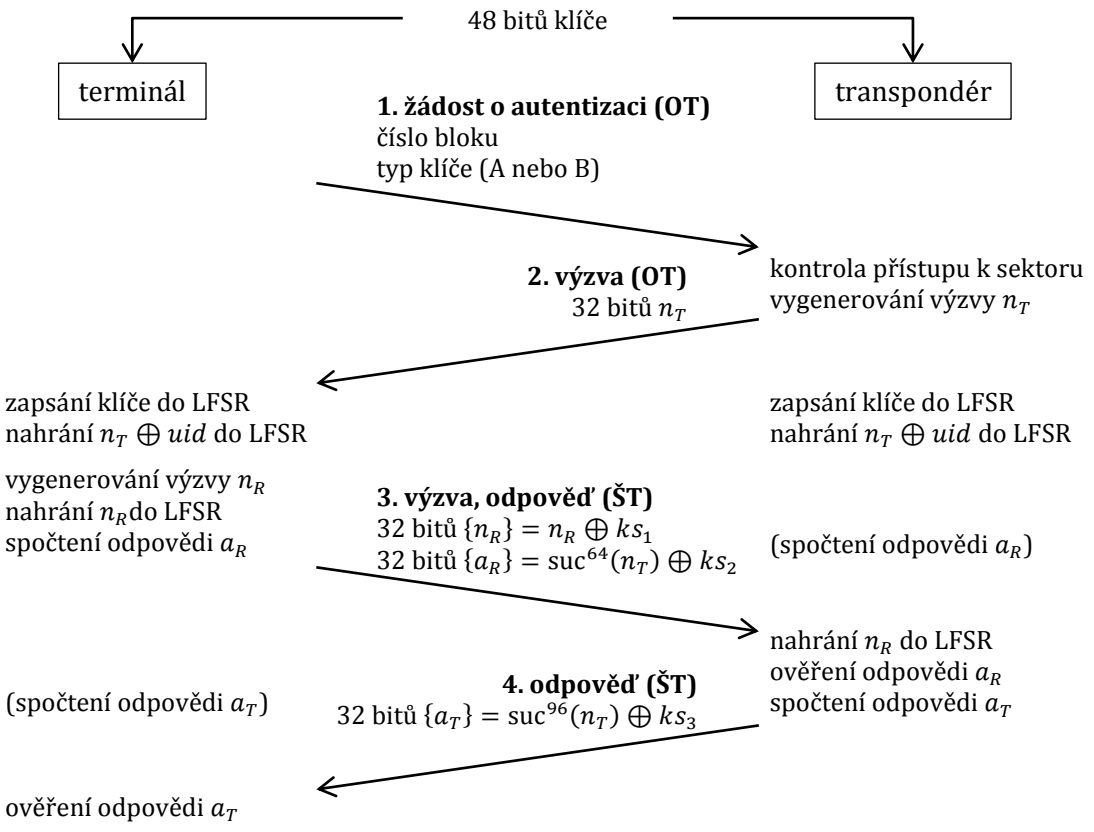
Definice 4.3-. Definujme funkci $\mathbb{F}_2^{32} \to \mathbb{F}_2^{32}$ pro výpočet odpovědí při autentizaci:

$$\mathbf{suc}(x_0, \dots, x_{31}) \coloneqq \big(x_1, \dots, x_{31}, G(x_0, \dots, x_{31})\big).$$

4.4 Inicializace šifry a autentizační protokol

Po ukončení antikolizní procedury, která je definována ISO normou a probíhá bez šifrování, vyčkává transpondér na příkazy terminálu pro aplikační vrstvu, to jest MIFARE.

V této fázi komunikace jsou možné pouze dva kroky – buď transpondér vypnout příkazem HALT, nebo se autentizovat pro přístup k vybranému sektoru příkazem AUTH. Autentizace pak probíhá dle obrázku 4.4-1. Formální definice použitého značení následuje po popisu protokolu.



Obrázek 4.4- Schéma autentizačního protokolu

1. Terminál zašle otevřeně žádost o autentizaci do zvoleného bloku a typ klíče, kterým se chce autentizovat.
2. Transpondér přečte přístupové bity příslušného sektoru, na základě kterých rozhodne, zda je žádost přípustná. Pokud není, vrátí otevřeně chybovou hlášku. Pokud je, přečte z náhodného generátoru aktuální hodnotu n_T, a odešle ji jako výzvu terminálu.

   Obě strany nyní znají klíč K, kterým se bude šifrovat, výzvu transpondéru n_T a jeho výrobní číslo uid, které bylo předáno během antikolizní procedury. Na obou stranách tedy může proběhnout inicializace šifry: do posuvného registru se přímo nakopíruje tajný klíč, a dále se tam za pomoci zpětné vazby nahraje $n_T \oplus uid$. Komunikace od této chvíle probíhá šifrovaně.
3. Terminál vygeneruje svou vlastní výzvu n_R libovolným způsobem, a nahrávaje ji se zpětnou vazbou do posuvného registru, odešle transpondéru. Speciálně, sama hodnota n_R ovlivňuje bity proudu klíče, kterými se šifruje. Terminál spočítá svou odpověď a_R na výzvu tranpondéru n_T (stačí 64 taktů registru náhodného generátoru) a odešle ji společně se svou výzvou transpondéru.
4. Transpondér obdrží $n_R \oplus ks_1$ a postupným dešifrováním bit po bitu rovněž nahraje n_R do posuvného registru šifry. Vzájemná autentizace je úspěšná,

pokud se nyní jak šifra na straně terminálu, tak na straně transpondéru nacházejí ve stejném stavu. Transpondér pak spočítá (dalšími takty registru náhodného generátoru) svou odpověď a odešle ji terminálu. Tato odpověď závisí jen a pouze na n_T, hodnota n_R však ovlivňuje proud klíče.

Formálně máme:

Označme

- $K \in \mathbb{F}_2^{48}$ klíč, $K = (k_0, \dots, k_{47})$;
- $n_T \in \mathbb{F}_2^{32}$ výzvu transpondéru, $n_T = (n_{T,0}, \dots, n_{T,31})$;
- $n_R \in \mathbb{F}_2^{32}$ výzvu terminálu, $n_R = (n_{R,0}, \dots, n_{T,31})$;
- $uid \in \mathbb{F}_2^{32}$ výrobní číslo transpondéru, $uid = (u_0, \dots, u_{31})$;
- $\alpha_i \in \mathbb{F}_2^{48}$ vnitřní stav registru šifry v čase i, $\alpha_i = (a_i, \dots, a_{i+47})$;
- $ks_i \in \mathbb{F}_2^{32}$ proud klíče v čase i, $ks_i = (b_{32i}, \dots, b_{32i+31})$, kde $b_i \coloneqq f(a_i, \dots, a_{i+47})$.

Hodnoty a_i jsou pro počáteční hodnoty i definovány:

$$\begin{aligned} a_i &\coloneqq k_i && \forall i \in [0,47] \\ a_{i+48} &\coloneqq L(a_i, \dots, a_{i+47}) \oplus n_{T,i} \oplus u_i && \forall i \in [0,31] \\ a_{i+80} &\coloneqq L(a_{i+32}, \dots, a_{i+79}) \oplus n_{R,i} && \forall i \in [0,31] \end{aligned}$$

Pro ostatní $i \in \mathbb{N}$ se na vývoji vnitřního stavu šifry podílí už jen zpětná vazba registru:

$$a_{i+112} \coloneqq L(a_{i+64}, \dots, a_{i+111}).$$

Odpovědi na výzvy se počítají pomocí zpětné vazby náhodného generátoru:

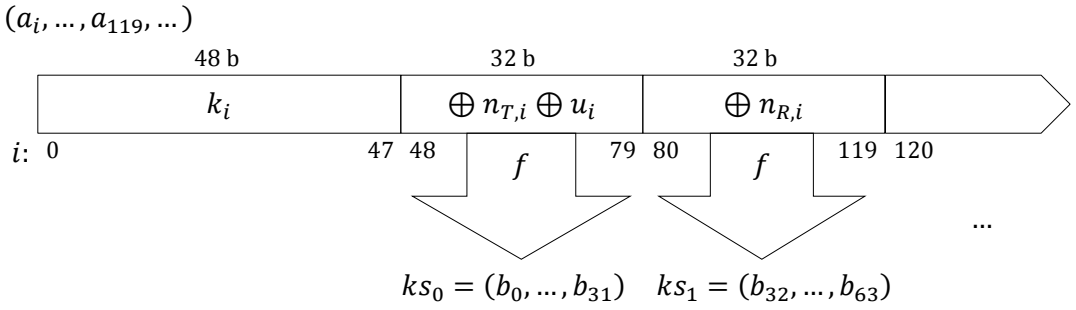
$$\begin{aligned} a_R &\coloneqq \mathrm{suc}^{64}(n_T) \\ a_T &\coloneqq \mathrm{suc}^{96}(n_T) \end{aligned}$$

dle definice 4.3-2, kde $\mathrm{suc}^2(\vec{x}) \coloneqq \mathrm{suc}(suc(\vec{x}))$ atd.

Šifrování pak představuje operace xor:

$$\begin{array}{lll} \{n_R\} \coloneqq n_R \oplus ks_1 & \{n_{R,i}\} \coloneqq n_{R,i} \oplus b_{i+32} & \forall i \in [0,31] \\ \{a_R\} \coloneqq a_R \oplus ks_2 & \{a_{R,i}\} \coloneqq a_{R,i} \oplus b_{i+64} & \forall i \in [0,31] \\ \{a_T\} \coloneqq a_T \oplus ks_3 & \{a_{T,i}\} \coloneqq a_{T,i} \oplus b_{i+96} & \forall i \in [0,31] \\ \dots & \dots & \dots \end{array}$$

Pro rychlejší orientaci je vývoj stavu znázorněn na obrázku 4.4-2.



Obrázek 4.4- Vývoj vnitřního stavu šifry

Po úspěšné autentizaci je transpondér připraven přijímat příkazy pro práci s bloky.

Reálný záznam autentizace vypadá následovně [**Ros09**]:

krok		**zdroj**	**šifrový text**	**otevřený text**
detekce		PCD		REQA 26
		PICC		ATQA 04 00
antikolize	1.	PCD		SEL NVB 93 20
	2.	PICC		UID BCC 2A 69 8D 43 8D
	5.	PCD		SEL NVB UID BCC 93 70 2A 69 8D 43 8D
	6.	PICC		SAK CRC 08 B6 DD
autentizace	1.	PCD		AUTH CRC 60 04 D1 3D
	2.	PICC		n_T 3B AE 03 2D
	3.	PCD	$\{n_R\}$ $\{a_R\}$ C4 94 A1 D2 6E 96 86 42	n_R a_R BB 03 1F 2D 7F CF 34 C3
	4.	PICC	$\{a_T\}$ 84 66 05 9E	a_T 86 9D BB D5
aplikační protokol		PCD	{READ} {CRC} 7D DE A6 B3	READ CRC 30 04 CD D1
		PICC	{16 bajtů data} E7 EE E3 AB 0F 89 BB ED 44 B1 91 CE EF 8A 4D CE {CRC} 4E 41	16 bajtů data 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 CRC CD EA

Obrázek 4.4- Záznam autentizace

V zaznamenané komunikaci byl použit klíč 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF.

4.5 Příkazy pro transpondéry MIFARE

Příkazy aplikačního protokolu byly odchyceny a dešifrovány v [**KHG08**]:

význam	**terminál**	**transpondér**	**terminál**	**transpondér**
AUTH A	`60 xx CRC`	32b výzva	32b výzva 32b odpověď	32b odpověď
AUTH B	`61 xx CRC`	32b výzva	32b výzva 32b odpověď	32b odpověď
READ	`30 xx CRC`	16 bajtů data CRC		
WRITE	`A0 xx CRC`	ACK nebo NACK	16 bajtů data CRC	
HALT	`50 00 CRC`			
ACK		`A`		
NACK		`4` (zamítnuto)		
NACK		`5` (chyba přenosu)		

Tabulka 4.5- Příkazy aplikační vrstvy MIFARE

Hodnoty jsou uvedeny v hexadecimálním zápisu.

Na místo `xx` se dosadí číslo bloku, na který se daný příkaz vztahuje.

Syntaxe ostatních příkazů (zvýšení/snížení hodnoty atd.) lze nalézt tamtéž.

5 Útoky

Implementace ani algoritmus Crypto1 nebyl doposud výrobcem zveřejněn. Karsten Nohl z University of Virginia ohlásil v roce 2007 na konferenci Chaos Communication Congress v Berlíně ve své prezentaci „Mifare—Little Security, despite Obscurity" úspěšnou zpětnou analýzu algoritmu na transpondérech MIFARE. V následujících letech pak i další týmy (Flavio D. Garcia a kolektiv na Radboud University Nijmegen; Nicolas T. Courtois na University College London) zveřejnily model šifry a z nich odvozené, stále rychlejší útoky. Výrobce se pokusil zabránit tomuto zveřejnění i soudní cestou [**Boo08**], avšak soud rozhodl v jeho neprospěch.

5.1 Hlavní slabiny algoritmu

5.1.1 Paritní bity

Jednou z největších slabin algoritmu jsou paritní bity. Norma [**ISO013**] týkající se transportní vrstvy MIFARE vyžaduje, aby každý přenášený bajt byl doplněn o lichý paritní bit. I když autoři normy pro transportní vrstvu stěží mohli zamýšlet kontrolu integrity jinde než na transportní vrstvě, neuvedení této skutečnosti explicitně v revizi normy, která byla v době vzniku MIFARE k dispozici, pravděpodobně vedlo výrobce k přesvědčení, že si může paritní bit využít pro vlastní potřebu.

A tak se paritní bity staly slabinou, bez které by žádný z dosud publikovaných útoků nebyl možný. Výrobce se dopustil hned několika implementačních chyb:

1. Paritní bity se počítají z otevřeného textu. Pro n_T, n_R, a_R a a_T tedy máme:

   Definice 5.1-.

   $$\boldsymbol{p_j} \coloneqq \boldsymbol{n_{T,8j}} \oplus \ldots \oplus \boldsymbol{n_{T,8j+7}} \oplus \mathbf{1}$$
   $$\boldsymbol{p_{j+4}} \coloneqq \boldsymbol{n_{R,8j}} \oplus \ldots \oplus \boldsymbol{n_{R,8j+7}} \oplus \mathbf{1}$$
   $$\boldsymbol{p_{j+8}} \coloneqq \boldsymbol{a_{R,8j}} \oplus \ldots \oplus \boldsymbol{a_{R,8j+7}} \oplus \mathbf{1}$$
   $$\boldsymbol{p_{j+12}} \coloneqq \boldsymbol{a_{T,8j}} \oplus \ldots \oplus \boldsymbol{a_{T,8j+7}} \oplus \mathbf{1}$$

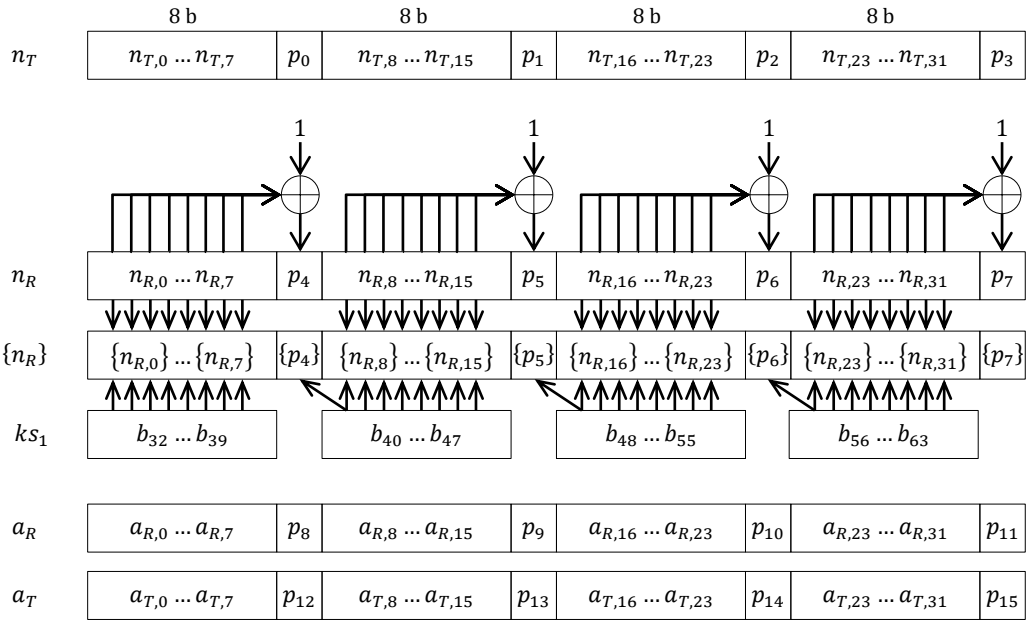
2. Paritní bity se šifrují stejným bitem klíče jako následující bit dat:

   Definice 5.1-.

   $$\{\boldsymbol{p_j}\} \coloneqq \boldsymbol{p_j} \oplus \boldsymbol{b_{8+8j}} \quad \forall \boldsymbol{j} \in [\mathbf{0}, \mathbf{11}]$$

3. Kontrola a případná reakce na správnost paritních bitů probíhá ještě před ověřením zaslaných dat. Je-li alespoň jeden z paritních bitů chybný,

transpondér neodpoví a zablokuje se stejně jako po příkazu HALT. Pokud jsou paritní bity v pořádku, ale odpověď a_R není správná, transpondér odpoví známou konstantní hodnotou NACK (chyba přenosu), a to zašifrovaně, přestože terminál neprokázal schopnost dešifrovat. Případný útočník tak může snadno získat 4 bity proudu klíče.



Obrázek 5.1- Výpočet a šifrování paritních bitů

terminál		**transpondér**
$\{\boldsymbol{p_4}\} \dots \{\boldsymbol{p_{11}}\}$	$\{\boldsymbol{n_R}\}, \{\boldsymbol{a_R}\}$	
špatně		bez odpovědi
OK	špatně	{NACK} (chyba přenosu)
OK	OK	$\{a_T\}$

Tabulka 5.1- Odpovědi transpondéru na paritní bity

5.1.2 Pseudonáhodný generátor

Druhou slabinou, přinejmenším usnadňující některé typy útoků, je pseudonáhodný generátor. Ačkoliv jsme jej v kapitole 4.3 definovali pro snadnější pochopení autentizačního protokolu jako 32bitový, pro účely generování náhodných čísel je ekvivalentní registru 16bitovému, neboť lineární zpětná vazba s prvními 16 bity nepracuje. Ve stávající literatuře se ostatně jako 16bitový zavádí rovnou. Přestože má tedy generovaná posloupnost nejdelší možnou periodu [**Gar09**], generuje pouze 65 535 čísel. Při nejběžněji používané přenosové rychlosti 105,9 kHz (odpovídající taktu 9,44 μs) to znamená opakování sekvence každých 618,62 ms.

Jelikož se aktuální hodnota generátoru jakožto posuvného registru s lineární zpětnou vazbou používá přímo jako výzva n_T při autentizaci, musí tato hodnota splňovat nutnou a postačující podmínku danou zpětnou vazbou, a to

$$n_{T,k} \oplus n_{T,k+2} \oplus n_{T,k+3} \oplus n_{T,k+5} \oplus n_{T,k+16} = 0 \qquad \forall k \in [0,15].$$

Sečtením přes všechna k dostáváme nutně

$$n_{T,0} \oplus n_{T,1} \oplus n_{T,3} \oplus n_{T,4} \oplus n_{T,18} \oplus n_{T,21} \oplus n_{T,22} \oplus n_{T,23} \oplus n_{T,24} \oplus n_{T,25} \oplus n_{T,26} \oplus \\ \oplus n_{T,27} \oplus n_{T,28} \oplus n_{T,29} \oplus n_{T,30} \oplus n_{T,31} = 0.$$

Obdobně pro všechny ostatní generované hodnoty, zejména pak pro správné odpovědi a_R a a_T, zasílané zašifrovaně. Lze tak předem vyloučit některé hodnoty proudu klíče.

Zásadní je ovšem nesmyslné rozhodnutí registr generátoru při každém zapnutí resetovat na fixní počáteční stav. Jak uvádí [**Noh08**], tento krok je zcela zbytečný, vyžaduje jen další hardware a ničí náhodnost získanou předchozí komunikací, ev. šumem. Zejména však činí tento generátor deterministickým a umožňuje tak útočníkovi přímo ovlivňovat autentizační výzvy tím, že bude kontrolovat čas mezi zapnutím generátoru (vložením transpondéru do elektromagnetického pole) a požadavkem na autentizaci.

5.2 Útoky hrubou silou

Zvolené autentizační schéma je velmi dobře navrženo i z hlediska útoků hrubou silou, neboť transpondér nikdy neprozradí cokoliv, co by souviselo s tajným klíčem dokud terminál neprokáže, že tento klíč zná, a to zasláním 64 bitového kryptogramu $(\{n_R\},\{a_R\})$, kde si prvních 32 bitů volí libovolně. Pravděpodobnost, že útočník uhodne odpověď je tedy 2^{-32}. Protože $\{a_R\} = \mathrm{suc}^{64}(n_T) \oplus ks_2$, tj. útočník zná ks_2, které zasílá, může při neúspěšném pokusu zamítnout všech $2^{48-32} = 2^{16}$ klíčů, které vedou ke stejnému ks_2. K úspěšnému útoku je tedy potřeba zhruba 2^{32} dotazů na transpondér, což při délce 0,5 vteřiny na jeden dotaz [**Cou09**] znamená přibližně 68 let, nepočítaje čas pro hledání zamítnutých klíčů.

Garcia a kol. [**Gar09**] pak uvádí modifikaci útoku hrubou silou s využitím slabin v paritních bitech. Útočník zašle náhodný kryptogram a s pravděpodobností 2^{-8} je všech osm paritních bitů v pořádku. V takovém případě mu transpondér odpoví zašifrovanými čtyřmi bity NACK, celkem tedy prozradí 12 bitů entropie 48 bitového klíče. Dostatečným opakováním tohoto pokusu (v praxi průměrně šestkrát, viz. tamtéž) lze z těchto informací zrekonstruovat klíč – stačí prohledat prostor klíčů a zjistit, který z nich ve všech (šesti) případech dá správné paritní bity a stejně zašifrovaný NACK.

Tento útok, narozdíl od předchozího, vyžaduje přibližně jen $6 \cdot 2^8 = 1536$ dotazů na transpondér, prohledávání klíčů je pak možné provádět offline.

5.3 Garciovy útoky

Následující útoky jsou popsány v [**Gar09**].

5.3.1 S konstantním n_T

Během tohoto útoku je nutné přesným časováním komunikace udržovat konstantní výzvu transpondéru n_T. Najdeme takové n_R, které nám umožní zmenšit prostor klíčů k prohledání z 2^{48} na přibližně $2^{32,77}$.

Na začátek je třeba zmínit, že k získání klíče stačí znát vnitřní stav posuvného registru v kterémkoliv čase, a posunout jej nazpět do výchozího stavu, neboť zpětná vazba je lineární.

Uvažujme dva autentizační pokusy a předpokládejme, že K, uid a n_T jsou v obou případech stejné. n_R buď výzva terminálu v prvním případě, n'_R výzva v druhém, ostatní značení obdobně. n_R se skládá ze čtyř bajtů po osmi bitech. Budeme sledovat, zda poslední bity v těchto bajtech ovlivňují proud klíče.

Definice 5.3-. Nechť $j \in \{0, 1, 2, 3\}$ a n_R a n'_R se shodují v prvních $6 + 8j$ bitech a liší se v $7 + 8j$-tém bitu, tj. $n_{R,0+8j} = n'_{R,0+8j} \dots n_{R,6+8j} = n'_{R,6+8j}$ a $n_{R,7+8j} \neq n'_{R,7+8j}$. Řekneme, že n_R má vlastnost F_j, pokud $b_{40+8j} \neq b'_{40+8j}$.

Ukážeme, že útočník může ze znalosti $\{n_R\}$ říci, zda má n_R vlastnost F_j.

1. Buď tedy $\{n_R\}$ pevné.
2. Náhodně zvolme $\{a_R\}$.
3. Zkusme se autentizovat postupně se všemi 256 možnými kombinacemi paritních bitů $\{p_4\} \dots \{p_{11}\}$, až dokud transpondér neodpoví chybovou hláškou, tj. dokud nám nepotvrdí správnou kombinaci paritních bitů.
4. Nyní změňme poslední bit j-tého bajtu $\{n_R\}$. Bity před tím ponechme stejné, a bity po tom, včetně odpovědi, zvolme libovolně. Tedy $\{n'_{R,7+8j}\} \coloneqq \{n_{R,7+8j}\} \oplus 1$.
5. Znovu se zkusme autentizovat, tentokrát pomocí $\{n'_R\}\{a'_R\}$, pro všechny kombinace paritních bitů $\{p'_4\} \dots \{p'_{11}\}$, dokud nenajdeme tu správnou. Jelikož jsme prvních $j - 1$ bajtů $\{n'_R\}$ nechali shodných s $\{n_R\}$, odpovídající paritní bity $\{p'_4\} \dots \{p'_{4+j-1}\}$ jsou rovněž shodné s $\{p_4\} \dots \{p_{4+j-1}\}$.
6. **Tvrzení 5.3-. n_R má vlastnost F_j právě když $\{p'_{4+j}\} = \{p_{4+j}\}$.**

Důkaz. Změnou bitu šifrového textu se změní odpovídající bit otevřeného textu (z vlastností exklusivního součtu), v našem případě

$$n'_{R,7+8j} = \{n'_{R,7+8j}\} \oplus b'_{39+8j} = \{n_{R,7+8j}\} \oplus 1 \oplus b_{39+8j} =$$
$$= (n_{R,7+8j} \oplus b_{39+8j}) \oplus 1 \oplus b_{39+8j} = n_{R,7+8j} \oplus 1 .$$

Tím pádem se změní i odpovídající paritní bit počítaný z otevřeného textu:

$$p'_{4+j} = \bigoplus_{i=0}^{7} n'_{R,i+8j} = \left(\bigoplus_{i=0}^{7} n_{R,i+8j} \right) \oplus 1 = p_{4+j} \oplus 1$$

Dále máme:

$$\{p_{4+j}\} = \{p'_{4+j}\} \Leftrightarrow 0 = \{p_{4+j}\} \oplus \{p'_{4+j}\}$$
$$\Leftrightarrow 0 = p_{4+j} \oplus b_{40+8j} \oplus p'_{4+j} \oplus b'_{40+8j}$$
$$\Leftrightarrow 0 = p_{4+j} \oplus b_{40+8j} \oplus (p_{4+j} \oplus 1) \oplus b'_{40+8j}$$
$$\Leftrightarrow 1 = b_{40+8j} \oplus b'_{40+8j}$$
$$\Leftrightarrow b_{40+8j} \neq b'_{40+8j} \Leftrightarrow n_R \text{ má vlastnost } F_j.$$

■

Nyní ukážeme, že pravděpodobnost, že n_R má vlastnost F_j, je relativně vysoká, $\frac{3}{32}$. Je třeba si uvědomit, že n_R se postupně nahrává do posuvného registru od nejnižšího bitu, a tak pravděpodobnost, že poslední bit kteréhokoliv bajtu ovlivní proud klíče, je ve všech čtyřech případech F_j stejná a to rovná pravděpodobnosti, že nelineární výstupní funkce f závisí právě na posledním bitu registru.

Nejdříve, některé vlastnosti funkcí, ze kterých je f složena:

Tvrzení 5.3-. Nechť $y_0, \dots, y_3$ jsou náhodné proměnné s rovnoměrným rozdělením z $\mathbb{F}_2$. Pak

$$\mathbf{Pr}[f_b(y_0, y_1, y_2, y_3) = f_b(y_0, y_1, y_2, y_3 \oplus 1)] = \frac{3}{4}.$$

Důkaz.

$$\Pr[f_b(y_0, y_1, y_2, y_3) = f_b(y_0, y_1, y_2, y_3 \oplus 1)]$$
$$= \Pr[f_b(y_0, y_1, y_2, 0) = f_b(y_0, y_1, y_2, 1)]$$
$$= \Pr[f_b(y_0, y_1, y_{2,0}) \oplus f_b(y_0, y_1, y_{2,1}) = 0] =: P_\beta$$

Vyjádříme si f_b v algebraické normální formě (viz. příloha A). Pak máme:

$$f_b(y_0, y_1, y_2, y_3) \equiv y_2 \oplus y_0 y_1 \oplus y_0 y_2 \oplus y_0 y_3 \oplus y_1 y_2 \oplus y_1 y_3$$

$$\oplus\, y_0y_1y_2 \oplus y_0y_2y_3 \oplus y_1y_2y_3$$

Dosazením za y_3 získáme:

$$P_\beta = \Pr[y_2 \oplus y_0y_1 \oplus y_0y_2 \oplus y_1y_2 \oplus y_0y_1y_2 \oplus$$
$$\oplus\, y_2 \oplus y_0y_1 \oplus y_0y_2 \oplus y_0 \oplus y_1y_2 \oplus y_1 \oplus y_0y_1y_2 \oplus y_0y_2 \oplus y_1y_2 = 0\,] =$$
$$= \Pr[y_0 \oplus y_1 \oplus y_0y_2 \oplus y_1y_2 = 0] =$$
$$= \Pr[(y_0 \oplus y_1) \oplus y_2(y_0 \oplus y_1) = 0] \overset{z:=y_0\oplus y_1}{=} \Pr[z \oplus y_2z = 0] =$$
$$= \Pr[z = y_2z] = \Pr[z = 0 \vee y_2 = 1] = 1 - \Pr[z = 1 \wedge y_2 = 0] =$$
$$= 1 - \Pr[z = 1] \cdot \Pr[y_2 = 0] = 1 - \Pr[y_0 \neq y_1] \cdot \frac{1}{2} =$$
$$= 1 - (\Pr[y_0 \neq 0\,|y_1 = 0] + \Pr[y_0 \neq 1\,|y_0 = 1]) \cdot \frac{1}{2} =$$
$$= 1 - (\Pr[y_0 \neq 0] \cdot \Pr[y_1 = 0] + \Pr[y_0 \neq 1] \cdot \Pr[y_1 = 1]) \cdot \frac{1}{2} =$$
$$= 1 - \left(\frac{1}{2} \cdot \frac{1}{2} + \frac{1}{2} \cdot \frac{1}{2}\right) \cdot \frac{1}{2} = 1 - \left(\frac{1}{4} + \frac{1}{4}\right) \cdot \frac{1}{2} = 1 - \frac{1}{2} \cdot \frac{1}{2} = 1 - \frac{1}{4} = \frac{3}{4}$$

■

Tvrzení 5.3-. Nechť $y_0, \dots, y_4$ jsou náhodné proměnné s rovnoměrným rozdělením z $\mathbb{F}_2$. Pak

$$\Pr[f_c(y_0, y_1, y_2, y_3, y_4) = f_c(y_0, y_1, y_2, y_3, y_4 \oplus 1)\,] = \frac{5}{8}.$$

Důkaz.

$$\Pr[f_c(y_0, y_1, y_2, y_3, y_4) = f_c(y_0, y_1, y_2, y_3, y_4 \oplus 1)\,]$$
$$= \Pr[f_c(y_0, y_1, y_2, y_3, 0) = f_c(y_0, y_1, y_2, y_3, 1)\,]$$
$$= \Pr[f_c(y_0, y_1, y_{2,0}) \oplus f_c(y_0, y_1, y_{2,1}) = 0\,] =: P_\gamma$$

Po vyjádření f_c v algebraické normální formě (viz. příloha A) máme:

$$f_c(y_0, y_1, y_2, y_3, y_4) \equiv y_0 \oplus y_4 \oplus y_0y_2 \oplus y_0y_3 \oplus y_0y_4 \oplus y_3y_4$$
$$\oplus\, y_0y_1y_3 \oplus y_0y_1y_4 \oplus y_0y_3y_4 \oplus y_1y_2y_3 \oplus y_1y_3y_4$$
$$\oplus\, y_0y_1y_2y_4 \oplus y_1y_2y_3y_4$$

Dosazením za y_4 získáme:

$$P_\gamma = \Pr[y_0 \oplus y_0y_2 \oplus y_0y_3 \oplus y_0y_1y_3 \oplus y_1y_2y_3 \oplus$$
$$\oplus\, y_0 \oplus 1 \oplus y_0y_2 \oplus y_0y_3 \oplus y_0 \oplus y_3 \oplus y_0y_1y_3 \oplus y_0y_1 \oplus y_0y_3 \oplus y_1y_2y_3 \oplus y_1y_3 \oplus$$
$$\oplus\, y_0y_1y_2 \oplus y_1y_2y_3 = 0] =$$
$$= \Pr[y_0 \oplus y_3 \oplus y_0y_3 \oplus y_0y_1 \oplus y_1y_3 \oplus y_0y_1y_2 \oplus y_1y_2y_3 = 1] =$$
$$= \Pr[(y_0 \oplus y_3) \oplus y_0y_3 \oplus y_1(y_0 \oplus y_3) \oplus y_1y_2(y_0 \oplus y_3) = 1]$$

Pro y_0, y_1 máme:

y_0	y_1	P_Y
0	0	$\Pr[0 \oplus 0 \oplus y_1 \cdot 0 \oplus y_1 y_2 \cdot 0 = 1] = \Pr[0 = 1] = 0$
1	1	$\Pr[0 \oplus 1 \oplus y_1 \cdot 0 \oplus y_1 y_2 \cdot 0 = 1] = \Pr[1 = 1] = 1$
0	1	$\Pr[1 \oplus 0 \oplus y_1 \oplus y_1 y_2 = 1] = \Pr[y_1 \oplus y_1 y_2 = 0] = \Pr[y_1 = y_1 y_2] =$ $= \Pr[y_1 = 0 \vee y_1 = 1] = 1 - \Pr[y_0 = 1] \cdot \Pr[y_1 = 0] =$ $= 1 - \frac{1}{2} \cdot \frac{1}{2} = 1 - \frac{1}{4} = \frac{3}{4}$
1	0	

Celkem:

$$0 \cdot \frac{1}{4} + 1 \cdot \frac{1}{4} + \frac{3}{4} \cdot \frac{1}{2} = \frac{1}{4} + \frac{3}{8} = \frac{5}{8}$$

■

Pro složenou funkci pak dostáváme:

Tvrzení 5.3-. Nechť $y_0, \dots, y_{19}$ jsou náhodné proměnné s rovnoměrným rozdělením z $\mathbb{F}_2$. Pak

$$\mathbf{Pr}[f(y_0, \dots, y_{18}, y_{19}) \neq f(y_0, \dots, y_{18}, y_{19} \oplus 1)\,] = \frac{3}{32}.$$

Důkaz. Označme

$$z_0 \coloneqq f_a(y_0, \dots, y_3)$$
$$z_1 \coloneqq f_b(y_4, \dots, y_7)$$
$$z_2 \coloneqq f_b(y_8, \dots, y_{11})$$
$$z_3 \coloneqq f_a(y_{12}, \dots, y_{15})$$
$$z_4 \coloneqq f_b(y_{16}, \dots, y_{19})$$

a

$$z_4' \coloneqq f_b(y_{16}, \dots, y_{19} \oplus 1).$$

Hodnoty $z_0, \dots z_4$ jsou nezávislé a dle tvrzení 4.2-1 a 4.2-2 s rovnoměrným rozdělením z $\mathbb{F}_2$.

$$\begin{aligned}
&\Pr[f(y_0,\dots,y_{18},y_{19}) \neq f(y_0,\dots,y_{18},y_{19}\oplus 1)] = \\
&\quad = \Pr[f_c(z_0,z_1,z_2,z_3,z_4) \neq f_c(z_0,z_1,z_2,z_3,z_4')] = \\
&\quad = \Pr[f_c(z_0,z_1,z_2,z_3,z_4) \neq f_c(z_0,z_1,z_2,z_3,z_4') | z_4 \neq z_4'] \cdot \Pr[z_4 \neq z_4'] + \\
&\quad + \Pr[f_c(z_0,z_1,z_2,z_3,z_4) \neq f_c(z_0,z_1,z_2,z_3,z_4') | z_4 = z_4'] \cdot \Pr[z_4 = z_4'] = \\
&\quad = \Pr[f_c(z_0,z_1,z_2,z_3,0) \neq f_c(z_0,z_1,z_2,z_3,1)] \cdot \\
&\quad \cdot \Pr[f_b(y_{16},y_{17},y_{18},0) \neq f_b(y_{16},y_{17},y_{18},1)] + 0 \cdot 0 = \\
&\quad = (1 - \Pr[f_c(z_0,z_1,z_2,z_3,0) = f_c(z_0,z_1,z_2,z_3,1)]) \cdot \\
&\quad \cdot (1 - \Pr[f_b(y_{16},y_{17},y_{18},0) = f_b(y_{16},y_{17},y_{18},1)]) =: P_{G1}
\end{aligned}$$

Dle předchozích dvou tvrzení pak

$$P_{G1} = \left(1 - \frac{5}{8}\right) \cdot \left(1 - \frac{3}{4}\right) = \frac{3}{8} \cdot \frac{1}{4} = \frac{3}{32}.$$

■

Při tomto útoku nás zajímají ta n_R, která mají všechny čtyři vlastnosti F_0, F_1, F_2 a F_3, tedy ta, jejichž každý 8. bit ovlivňuje proud klíče.

Ukažme, jak je možné takové n_R nalézt.

1. Útočník zkouší postupně všech 256 možností pro první bajt $\{n_R\}$ a dle postupu výše hledá takové $\{n_R\}$, že n_R má vlastnost F_0, tj. že poslední bit prvního bajtu ovlivňuje proud klíče.
2. Pokud takové $\{n_R\}$ nalezne, zkouší postupně všech 256 možností pro druhý bajt $\{n_R\}$ a stejným způsobem hledá takové $\{n_R\}$, že n_R má vlasntost F_1. Obdobně postupuje i pro třetí a čtvrtý bajt a vlastnosti F_2, resp. F_3.
3. V případě, že v některém z kroků žádné takové $\{n_R\}$, pro které by n_R mělo vlastnost F_j, neexistuje, vrátí se útočník o krok zpět a hledá jiné $\{n_R\}$, pro které n_R splňuje F_{j-1} a postup opakuje.
4. V případě, že neexistuje žádné takové $\{n_R\}$, pro které by n_R mělo všechny čtyři vlastnosti F_j, útok selhal. Jelikož je počáteční stav šifry ovlivněn i hodnotou n_T (viz. Poznámka 5.3-7), kterou považujeme za konstantní, je v takovém případě třeba celý postup zopakovat s jinou hodnotou n_T.

Jelikož vlastnost F_j závisí pouze na j-tém bajtu, existuje $(2^8)^4 \cdot \left(\frac{3}{32}\right)^4 = (2^3 \cdot 3)^4 = 24^4 = 331\,776$ $\{n_R\}$, že n_R má všechny čtyři vlastnosti $F_0 \dots F_3$. Garcia v [**Gar09**] uvádí, že obvykle stačí kolem 28 500 dotazů na transpondér k nalezení takového $\{n_R\}$.

Stěžejní pro tento útok je pozorování, že při použití takového $\{n_R\}$ při autentizaci existuje výrazně méně než 2^{48} možných vnitřních stavů šifry:

Tvrzení 5.3-. Nechť n_R má vlastnosti F_0, F_1, F_2 a F_3. Pak existuje právě $436 \cdot 2^{24}$ možných vnitřních stavů šifry po nahrání $\{n_R\}$ do registru, tj pro bity $a_{64} \dots a_{111}$.

Důkaz. Z definice f víme, že nezávisí na sudých bitech registru. Pro ně tedy máme 2^{24} možností. Musíme ukázat, že pro liché bity zbývá jen 436 možností.

Označme l_i liché bity registru po nahrání $\{n_R\}$, tj. $l_0 = a_{65}, l_1 = a_{67}, \dots l_{23} = a_{111}$. Především, z předpokladů věty vyplývají následující vztahy:

$$F_3 : f(l_4, \dots, l_{23}) \neq f(l_4, \dots, l_{23} \oplus 1)$$
$$F_2 : f(l_0, \dots, l_{19}) \neq f(l_0, \dots, l_{19} \oplus 1)$$
$$F_1 : f(l_{-4}, \dots, l_{15}) \neq f(l_{-4}, \dots, l_{15} \oplus 1)$$
$$F_0 : f(l_{-8}, \dots, l_{11}) \neq f(l_{-8}, \dots, l_{11} \oplus 1)$$

Z předchozích tvrzení je zřejmé, že aby kterýkoliv ze vztahů platil, musí nutně

$$f_c(C_0, \dots, C_4) \neq f_c(C_0, \dots, C_4 \oplus 1) \tag{1a}$$

$$f_b(B_0, \dots, B_3) \neq f_b(B_0, \dots, B_3 \oplus 1). \tag{2a}$$

Z tabulky hodnot funkcí f_c a f_b (viz. příloha A) vidíme, že to znamená

$$C \coloneqq (C_0, C_1, C_2, C_3) = \begin{cases} (0,0,0,0) \\ (0,0,1,0) \\ (0,1,0,0) \\ (0,1,0,1) \\ (0,1,1,0) \\ (1,1,0,0) \end{cases} \tag{1b}$$

resp.

$$B \coloneqq (B_0, B_1, B_2) = \begin{cases} (0,1,0) \\ (1,0,0) \end{cases}. \tag{2b}$$

Všechny čtveřice $(l_{20}, \dots, l_{23})$, $(l_{16}, \dots, l_{19})$, $(l_{12}, \dots, l_{15})$ a $(l_8, \dots, l_{11})$ musí alespoň v jednom z uvedených vztahů vstupovat jako poslední čtyři bity do f_b, tedy všechny musí splňovat (2a) resp. (2b).

Dále, čtveřice $(l_{20}, \dots, l_{23})$ musí splňovat pouze jedinou podmínku (danou vlastností F_3), takže může nabývat všech čtyřech možností (0,1,0,0), (0,1,0,1), (1,0,0,0) a (1,0,0,1).

Pro větší názornost následujícího popisu rozepišme:

$${}_{F_3}C = \left({}_{F_3}C_0, {}_{F_3}C_1, {}_{F_3}C_2, {}_{F_3}C_3\right) = \left(f_a(l_4, \dots, l_7), f_b(l_8, \dots, l_{11}), f_b(l_{12}, \dots, l_{15}), f_a(l_{16}, \dots, l_{19})\right)$$
$${}_{F_2}C = \left({}_{F_2}C_0, {}_{F_2}C_1, {}_{F_2}C_2, {}_{F_2}C_3\right) = \left(f_a(l_0, \dots, l_3), f_b(l_4, \dots, l_7), f_b(l_8, \dots, l_{11}), f_a(l_{12}, \dots, l_{15})\right)$$
$${}_{F_1}C = \left({}_{F_1}C_0, {}_{F_1}C_1, {}_{F_1}C_2, {}_{F_1}C_3\right) = \left(f_a(l_{-4}, \dots, l_{-1}), f_b(l_0, \dots, l_3), f_b(l_4, \dots, l_7), f_a(l_8, \dots, l_{11})\right)$$
$${}_{F_0}C = \left({}_{F_0}C_0, {}_{F_0}C_1, {}_{F_0}C_2, {}_{F_0}C_3\right) = \left(f_a(l_{-8}, \dots, l_{-5}), f_b(l_{-4}, \dots, l_{-1}), f_b(l_0, \dots, l_3), f_a(l_4, \dots, l_7)\right)$$

Nechť ${}_{F_3}C_2 = f_b(l_{12}, \dots, l_{15}) = 0$, a z (2b) a tabulky máme $(l_{12}, \dots, l_{15}) = (0,1,0,0)$ nebo $(1,0,0,0)$. V obou případech je ${}_{F_2}C_3 = f_a(l_{12}, \dots l_{15}) = 1$. Taková podmínka existuje dle (1b) na ${}_{F_2}C$ pouze jedna, a to $(0,1,0,1)$. Tudíž musí $0 = {}_{F_2}C_2$ a obdobně dostáváme ${}_{F_1}C = (0,1,0,1)$ a tedy $0 = {}_{F_1}C_2$. Ovšem ${}_{F_1}C_2 = f_b(l_4, \dots, l_7) = {}_{F_2}C_1 = 1$, což je spor a nutně ${}_{F_3}C_2 = f_b(l_{12}, \dots, l_{15}) = 1$.

Nechť tedy ${}_{F_3}C_2 = f_b(l_{12}, \dots, l_{15}) = 1$, tj. $(l_{12}, \dots, l_{15}) = (0,1,0,1)$ nebo $(1,0,0,1)$. Tím jsme ale omezili podmínky na ${}_{F_3}C$

$${}_{F_3}C = \begin{cases} (0,0,1,0) \\ (0,1,1,0) \end{cases} \tag{3}$$

a dostáváme ${}_{F_3}C_0 = {}_{F_3}C_3 = 0$.

Protože ${}_{F_3}C_3 = 0 = f_a(l_{16}, \dots, l_{19})$, z tabulky hodnot f_a a podmínky (2b) plyne $(l_{16}, \dots, l_{19}) = (1,0,0,1)$.

Kdyby ${}_{F_2}C_3 = 1$, dostali bychom pro ${}_{F_0}C_2$ a ${}_{F_1}C_1$ stejný spor jako výše. Máme tedy ${}_{F_2}C_3 = 0 = f_a(l_{12}, \dots, l_{15})$, stejně jako v předchozím odstavci $(l_{12}, \dots, l_{15}) = (1,0,0,1)$ a také

$${}_{F_2}C = \begin{cases} (0,0,1,0) \\ (0,1,1,0) \end{cases} \tag{4}$$

Dále ${}_{F_3}C_0 = 0 = f_a(l_4, \dots, l_7)$ (3) a ${}_{F_2}C_0 = 0 = f_a(l_0, \dots, l_3)$ (4). Tyto bity však už nejsou omezeny podmínkou (2b).

Shrňme dosavadní výsledky:

	$(l_0, \dots, l_3)$	$(l_4, \dots, l_7)$	$(l_8, \dots, l_{11})$	$(l_{12}, \dots, l_{15})$	$(l_{16}, \dots, l_{19})$	$(l_{20}, \dots, l_{23})$
možnosti	$f_a(\dots) = 0$	$f_a(\dots) = 0$	(0,1,0,0) (0,1,0,1) (1,0,0,0) (1,0,0,1)	(1,0,0,1)	(1,0,0,1)	(0,1,0,0) (0,1,0,1) (1,0,0,0) (1,0,0,1)

Také případ ${}_{F_1}C_3 = 1$ už nevede k výše zmiňovanému sporu (jelikož nemáme žádné F_{-1}). Rozdělme si tedy zbylé případy na ${}_{F_1}C_3 = 1$ a ${}_{F_1}C_3 = 0$.

Opět z tabulky f_a a (2b) je zřejmé, že v případě ${}_{F_1}C_3 = 0$ je $(l_8, \dots, l_{11}) = (1,0,0,1)$. Protože

1. ${}_{F_1}C_2$ může mít hodnotu jak 1 tak 0 (1b);
2. v obou případech může ${}_{F_1}C_1$ také nabývat obou hodnot (1b);

3. $(l_{-4}, \dots, l_{-1})$ můžeme zvolit libovolně;

4. existují $y_0, \dots y_3$, že $f_a(y_0, \dots, y_3)$ nabývá obou hodnot 1 a 0 (tvrzení 4.2-1);

nejsou bity $l_0, \dots, l_7$ nijak dále omezeny.

Naopak v případě ${}_{F_1}C_3 = 1$ (zbylé možnosti pro $(l_8, \dots, l_{11})$) musí být ${}_{F_1}C_2 = 0 = f_b(l_4, \dots, l_7)$ a ${}_{F_1}C_1 = 1 = f_b(l_0, \dots, l_3)$ (1b). Stejně tak lze díky libovolnosti $l_{-8}, \dots l_{-1}$ a surjektivnosti f_a a f_b splnit zbylé podmínky ${}_{F_1}C$ a ${}_{F_0}C$.

Celkem tedy:

	$(l_0, \dots, l_3)$	$(l_4, \dots, l_7)$	$(l_8, \dots, l_{11})$	$(l_{12}, \dots, l_{15})$	$(l_{16}, \dots, l_{19})$	$(l_{20}, \dots, l_{23})$
možnosti	$f_a(\dots) = 0$	$f_a(\dots) = 0$	(1,0,0,1)	(1,0,0,1)	(1,0,0,1)	(0,1,0,0) (0,1,0,1) (1,0,0,0) (1,0,0,1)
	$f_a(\dots) = 0$ $f_b(\dots) = 1$	$f_a(\dots) = 0$ $f_b(\dots) = 0$	(0,1,0,0) (0,1,0,1) (1,0,0,0)	(1,0,0,1)	(1,0,0,1)	(0,1,0,0) (0,1,0,1) (1,0,0,0) (1,0,0,1)

S pomocí tabulek hodnot pak lze získat všechny možnosti pro liché bity registru:

	$(l_0, \dots, l_3)$	$(l_4, \dots, l_7)$	$(l_8, \dots, l_{11})$	$(l_{12}, \dots, l_{15})$	$(l_{16}, \dots, l_{19})$	$(l_{20}, \dots, l_{23})$
možnosti	(0,0,0,0) (0,0,0,1) (0,0,1,0) (0,1,1,0) (0,1,1,1) (1,0,0,1) (1,0,1,0) (1,1,0,1)	(0,0,0,0) (0,0,0,1) (0,0,1,0) (0,1,1,0) (0,1,1,1) (1,0,0,1) (1,0,1,0) (1,1,0,1)	(1,0,0,1)	(1,0,0,1)	(1,0,0,1)	(0,1,0,0) (0,1,0,1) (1,0,0,0) (1,0,0,1)
	(0,0,1,0) (1,0,0,1) (1,1,0,1)	(0,0,0,0) (0,0,0,1) (0,1,1,0) (0,1,1,1) (1,0,1,0)	(0,1,0,0) (0,1,0,1) (1,0,0,0)	(1,0,0,1)	(1,0,0,1)	(0,1,0,0) (0,1,0,1) (1,0,0,0) (1,0,0,1)

Na počet:

$$8 \cdot 8 \cdot 1 \cdot 1 \cdot 1 \cdot 4 + 3 \cdot 5 \cdot 3 \cdot 1 \cdot 1 \cdot 4 = 256 + 180 = 436.$$

■

Poznámka 5.3-. V obou případech posledního kroku jsme předpokládali libovolnost $l_{-8}, \dots l_{-1}$. Ve skutečnosti se ale jedná o předchozí hodnoty posuvného registru, speciálně, $l_{-8} = a_{49} \dots l_{-1} = a_{63}$, které závisí kromě klíče K a výrobního čísla uid na nonci n_T. Proto se může stát, že k danému n_T neexistuje žádné $\{n_R\}$, pro které by n_R mělo všechny vlastnosti $F_0, \dots, F_3$, a útok může selhat.

Jakmile tedy útočník najde $\{n_R\}$, že n_R má všechny vlastnosti $F_0, \dots, F_3$, stačí prohledat jen $436 \cdot 2^{24} = 109 \cdot 2^{26} \cong 7{,}315 \cdot 10^9$ vnitřních stavů šifry a zpětným posouváním registru vyzkoušet příslušné klíče k rozšifrování obdržených chybových zpráv.

5.3.2 S konstantním n_R

V předchozím útoku předpokládal útočník konstantní n_T a snažil se najít speciální $\{n_R\}$, které by prozradilo netriviální informaci o vnitřním stavu šifry. Nyní ponechme konstantní $\{n_R\}, \{a_R\}$ i paritní bity a pokusme se najít konkrétní n_T.

1. Vytvořme si tabulku T vnitřních stavů, které pro $\{n_R\} = \{a_R\} = 0$ dávají $\{p_4\} = \dots = \{p_{11}\} = 0$ a $b_{96} = \dots = b_{99} = 0$.
2. Buď $\{n_{R,i}\} = \{a_{R,i}\} = \{p_4\} = \dots = \{p_{11}\} = 0$.
3. Zkoušejme se s touto odpovědí autentizovat pro různá n_T tak dlouho, dokud transpondér neodpoví chybovou hláškou zašifrovanou bity $b_{96} = \dots = b_{99} = 0$. Získáme tak 12 bitů informace (8 paritních bitů a 4 bity proudu klíče).
4. Pak víme, že vnitřní stav šifry transpondéru po odeslání n_T je ve vytvořené tabulce, a můžeme každou položku projít, zpětným posunem získat kandidátní klíč pro dané n_T a ověřit jej.

Velikost takové tabulky jest $2^{48-12} = 2^{36}$ položek o velikosti 48 bitů, tedy celkem

$$2^{36} \cdot \frac{\frac{48}{8}}{1024^3} = 2^{36} \cdot 6 \cdot 2^{-30} = 2^6 \cdot 6 = 384 \text{ GB}.$$

Prohledávání lze urychlit rozdělením tabulky za cenu dalšího dotazování transpondéru:

4. Buď n_T nalezené n_T z kroku 3 a tentokrát $\{n_{R,i}\} = \{a_{R,i}\} = 1$.
5. Zkusme se autentizovat postupně se všemi 256 možnými kombinacemi paritních bitů $\{p_4\} \dots \{p_{11}\}$, až dokud transpondér neodpoví chybovou hláškou, tj. dokud nám nepotvrdí správnou kombinaci paritních bitů.
6. Označme nyní $\tau := (\{p_4\}, \{p_5\}, \{p_6\}, \{p_7\}, \{p_8\}, \{p_9\}, \{p_{10}\}, \{p_{11}\}, b_{96}, b_{97}, b_{98}, b_{99})$

Rozdělíme-li pak tabulku T tak, že vnitřní stavy v podtabulce T_τ pro $\{n_{R,i}\} = \{a_{R,i}\} = 1$ dávají $\{p_4\} = \tau_0 \dots \{p_{11}\} = \tau_7$ a $b_{96} = \tau_8 \dots b_{99} = \tau_{11}$, stačí k nalezení klíče prohledat jednu podtabulku T_τ velikosti $2^{36-12} = 2^{24}$ prvků, což představuje 96 MB.

[**Gar09**] uvadí, že průměrný počet nutných pokusů o autentizaci v kroku 3 je 4 096, plus dalších 128 v kroku 5.

5.4 Courtoisův útok

V roce 2009 publikoval Nicolas Tadeusz Courtois nový útok na transpondéry MIFARE [**Cou09**], vyžadující průměrně 300 autentizačních pokusů a předvýpočet zanedbatelného množství dat (v řádu desítek bajtů).

5.4.1 Popis útoku

Podstata tohoto útoku spočívá ve využití malé variability proudu klíče a následného použití diferenční kryptoanalýzy. Pravděpodobnost, že poslední tři bity proudu klíče po nahrání $\{n_R\}$ nezávisí na posledních třech bitech $\{n_R\}$ je téměř 72 %:

Tvrzení 5.4-. Nechť $\boldsymbol{u, v, w} \in \mathbb{F}_2$. Pak

$$\boldsymbol{P_n} \coloneqq \mathbf{Pr}[\boldsymbol{f}(\boldsymbol{x_0}, \dots, \boldsymbol{x_{44}}, \boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}) = \boldsymbol{f}(\boldsymbol{x_0}, \dots, \boldsymbol{x_{44}}, \boldsymbol{r}, \boldsymbol{s}, \boldsymbol{t})\ \forall \boldsymbol{r}, \boldsymbol{s}, \boldsymbol{t} \in \mathbb{F}_2] = \frac{\mathbf{23}}{\mathbf{32}}.$$

Důkaz. Z definice f je zřejmé, že f nezávisí na posledních třech bitech pouze pokud f_c nezávisí na posledním bitu, nebo pokud f_b nezávisí na posledních dvou bitech. Tedy

$$\begin{aligned} P_n = \Pr[f_c(y_0, y_1, y_2, y_3, y_4) = f_c(y_0, y_1, y_2, y_3, y_4 \oplus 1)] + \\ + \Pr[f_c(y_0, y_1, y_2, y_3, y_4) \neq f_c(y_0, y_1, y_2, y_3, y_4 \oplus 1)] \cdot \\ \cdot \Pr[f_b(x_{41}, x_{43}, v, w) = f_b(x_{41}, x_{43}, s, t)\ \forall s, t \in \mathbb{F}_2\ ]. \end{aligned}$$

Za pomoci tvrzení 5.3-5 pak dosadíme:

$$P_n = \frac{5}{8} + \frac{3}{8} \cdot \Pr[f_b(x_{41}, x_{43}, v, w) = f_b(x_{41}, x_{43}, s, t)\ \forall s, t \in \mathbb{F}_2\ ]$$

Připomeňme si funkci f_b:

$$f_b(y_0, y_1, y_2, y_3) \coloneqq \big((y_0 \wedge y_1) \vee y_2\big) \oplus \big((y_0 \oplus y_1) \wedge (y_2 \vee y_3)\big)$$

Potřebujeme určit, s jakou pravděpodobností f_b nezávisí na y_2 a y_3. Vidíme, že bude-li $b_B \coloneqq (y_0 \oplus y_1) = 0$, pak výraz $b_C \coloneqq (y_2 \vee y_3)$ nemá žádný význam. Obdobně, je-li $b_A \coloneqq (y_0 \wedge y_1) = 1$, pak se hodnota y_2 rovněž neprojeví. Tedy

$$\begin{aligned} \Pr[f_b(x_{41}, x_{43}, v, w) = f_b(x_{41}, x_{43}, s, t)\ \forall s, t \in \mathbb{F}_2\ ] = \Pr[b_B = 0 \wedge b_A = 1] = \\ = \Pr[b_A = 1] = \Pr[y_0 = y_1 = 1] = \frac{1}{4}. \end{aligned}$$

Celkem:

$$P_n = \frac{5}{8} + \frac{3}{8} \cdot \frac{1}{4} = \frac{20}{32} + \frac{3}{32} = \frac{23}{32}$$

■

Jinými slovy, pokud zafixujeme prvních 29 bitů $\{n_R\}$, s pravděpodobností $\frac{23}{32}$ bude $ks_1 = (b_{32}, \dots, b_{63})$ stejné bez ohledu na hodnotu $\{n_R\}$. Další důležité pozorování je, že po nahrání $\{n_R\}$ do registru šifry se její další stav vyvíjí již jen na základě lineární zpětné vazby. Speciálně, rozdíl ve dvou vnitřních stavech pocházejících z $\{n_R\}$ s různými posledními třemi bity záleží pouze na rozdílu mezi těmito bity. Útočník pak může postupovat následovně:

1. Buď n_T pevné. Zkoušejme se autentizovat s náhodným $(\{n_R\}, \{a_R\})$ (a pevnými nebo náhodnými hodnotami paritních bitů $\{p_4\}, \dots, \{p_{11}\}$) tak dlouho, dokud nám transpondér neodpoví chybovou hláškou (zašifrovanou čtyřmi bity proudu klíče $b_{96}, b_{97}, b_{98}, b_{99}$).
2. Zafixujme nyní prvních 29 z 32 bitů $\{n_R\}$, odpovídající hodnoty paritních bitů $(\{p_4\}, \{p_5\}$ a $\{p_6\})$ a celé $\{a_R\}$. Celkem tedy máme proměnné 3 poslední bity $\{n_R\}$ a 5 hodnot paritních bitů.
3. Pro všech $2^3 = 8$ možností $\{n_R\}$ hledejme dalšími autentizačními pokusy správnou kombinaci paritních bitů (odpovídajícím poslednímu bajtu $\{n_R\}$ a pevnému $\{a_R\}$).

Po těchto třech krocích tedy máme:

- 8 dotazů $(\{n_R\}, \{a_R\})_{1\dots8}$ lišících se jen ve třech bitech $\{n_R\}$;
- ke každému z nich správné hodnoty paritních bitů $\{p_4\}_{1\dots8}$ až $\{p_{11}\}_{1\dots8}$;
- 8 odpovědí NACK $e_{1\dots8}$ o délce 4 bitů, zašifrovaných bity proudu klíče.

Dále z definice víme, že výpočet proudu klíče, tedy nelineární funkce f, závisí pouze na 20 hodnotách registru šifry. Navíc, jedná se o všechny liché bity x_9 až x_{47} a při výpočtu $b_{96}, \dots, b_{99}$ se registr vyvíjí už jen pomocí zpětné vazby. To ovšem znamená, že pokud $b_{96} = f(a_{121}, a_{123}, \dots, a_{159})$, pak $b_{98} = f(a_{123}, a_{125}, \dots, a_{161})$, tj. právě 21 bitů vnitřního stavu určuje hodnoty b_{96} a b_{98}.

4. Prohledáme prostor všech možností pro bity $a_{121}, a_{123}, \dots, a_{161}$. Pro každý dotaz spočtěme rozdíl

$$\Delta^{\mathrm{i}} = (\{n_R\}, \{a_R\})_1 \oplus (\{n_R\}, \{a_R\})_i, i \in [1, 8].$$

Necháme-li vnitřní stav šifry vyvinout s dotazem Δ^{i} (a všemi předchozími bity rovnými 0), získáme (s pravděpodobností $23/32$) rozdíl vnitřních stavů během šifrování odpovědi

$$\delta^i = \Big(\big(a_{121_1} \oplus a_{121_i}\big), \big(a_{123_1} \oplus a_{123_i}\big), \dots, \big(a_{161_1} \oplus a_{161_i}\big) \Big).$$

Pro každou z 2^{21} možností musíme ověřit, že

$$f\left(a_{121} \oplus \delta_0^i, a_{123} \oplus \delta_1^i, \dots, a_{159} \oplus \delta_{19}^i\right) = b_{96}$$

a

$$f\left(a_{123} \oplus \delta_1^i, a_{125} \oplus \delta_1^i, \dots, a_{161} \oplus \delta_{20}^i\right) = b_{98}.$$

5. Stejný postup opakujeme pro sudé bity $a_{122}, a_{124}, \dots a_{162}$ a b_{97}, b_{99}.

Protože je f_c balancovaná (tvrzení 4.2-3), bude těmto podmínkám vyhovovat zhruba polovina, tj. 2^{10} možností. Získáme tak možnosti pro 42 bitů vnitřního stavu šifry, který má 48 bitů celkem. Zbylých 6 bitů musíme prohledat hrubou silou.

6. Pro každou z takto získaných možností a pro všech 2^6 hodnot pro zbývající bity vnitřního stavu provedeme zpětný posun registru až do počátečního stavu, a získáme tak kandidátní klíč. Pro tento klíč zbývá ověřit, že sedí všechny paritní bity $\{p_4\}_{1\dots8}, \dots \{p_{11}\}_{1\dots8}$.

Pokud žádný z kandidátních klíčů nevyhovuje paritním bitům, znamená to, že poslední bity proudu klíče závisí na měněných bitech $\{n_R\}$, útok selhal a je třeba jej znovu opakovat.

5.4.2 Složitost útoku

V prvním kroku máme $2^8 = 256$ možností pro paritní bity, v průměru tedy budeme potřebovat 128 dotazů na transpondér.

Ve třetím kroku pak máme $2^5 = 32$ možností pro hledané paritní bity, v průměru budeme potřebovat 16 dotazů na transpondér pro každý ze sedmi dotazů (první už máme z kroku 1).

A konečně, předpokládaný počet opakování celého postupu je $\frac{32}{23}$. Celkem máme průměrně

$$(128 + 16 \cdot 7) \cdot \frac{32}{23} \doteq 240 \cdot 1{,}391 = 334$$

dotazů na transpondér. V [**Cou09**] lze pak najít ještě malé vylepšení prvního kroku při opakování pokusu.

Druhá část útoku spočívá zejména ve dvou prohledáváních prostoru velikosti 2^{21}.

Rozdíl vnitřních stavů v kroku 4 lze předpočítat pro různá Δ^i dopředu (pro všechny 3 bity necháme vyvinout vnitřní stav od a_{112} do a_{163}, což představuje tabulku velikosti $8 \cdot 52$ bitů).

6 Závěr

Předmětem práce bylo nastudovat útok N. Courtoise, který značně vychází z práce kolektivu F. D. Garciy. V rámci této práce byly představeny myšlenky a základní principy RFID a shrnuta možná rizika pro tuto technologii. Byly popsány dostupné transpondéry MIFARE a uvedeny bezpečnostní vlastnosti nejpoužívanějšího transpondéru, kterého se útok týká. Byl představen algoritmus šifry Crypto1, analyzován jeho návrh a zkoumány vlastnosti použitých booleovských funkcí. Dále bylo poukázáno na slabiny v realizaci šifry, zejména na způsob výpočtu paritních bitů a vlastnosti náhodného generátoru. Byly prozkoumány možnosti útoku hrubou silou a jeho zlepšení při využití těchto slabin. V rámci práce byly dále analyzovány dva Garciovy útoky, které byly doplněny o chybějící důkazy uvedených tvrzení. V závěru se práce zabývá útokem Courtoisovým, který je doposud nejrychlejším známým útokem na transpondéry MIFARE Classic, a jehož implementace je k dispozici na přiloženém CD.

Bezpečnost těchto transpondérů byla uvedenými útoky zcela prolomena. K naklonování transpondéru stačí i s běžně dostupným hardwarem jen několik málo minut v jeho blízkosti. Hardwarová realizace je mimo obor této práce, nicméně jedno ze zařízení vhodných k takovému účelu by mohl být tzv. *P*ic*N*ic Tomáše Rosy, viz. **[Ros091]** nebo http://crypto.hyperlink.cz/PicNic.htm.

Implementace, která je součástí této práce – pravděpodobně poprvé vytvořena v jazyce C# – pak v kombinaci s prostředím .NET Micro Framework usnadňuje konstrukci takového hardwaru i pro méně odbornou veřejnost. Tato realizace bude také k dispozici na http://www.microframework.cz/Go.axd?Projects-Mifare.

Již v roce 1883 vyslovil Dr. Auguste Kerckhoffs jeden ze základních principů kryptografie [**Ker83**], a to, že bezpečnost jakéhokoliv systému by neměla být ovlivněna prozrazením jeho návrhu nepříteli. Nedodržení tohoto principu při návrhu algoritmu Crypto1 pak postavilo výrobce s tolika prodanými kusy do těžké situace.

Nezbývá než věřit, že tato technologie nebude použita k ochraně citlivých údajů (jak se téměř stalo v případě OpenCard) a k nápravě ve stávajících systémech dojde dříve, než bude způsobena závažná škoda.

7 Použitá literatura

[Boo08] "NXP vs. Radboud," 171900 / KG ZA 08-415 BD7578. Voorzieningenrechter Rechtbank Arnhem., Arnhem, 2008.

[Cou09] Nicolas T. Courtois, "The Dark Side of Security by Obscurity and Cloning MiFare Classic Rail and Building Passes Anywhere, Anytime," Cryptology ePrint Archive, Report 137, 2009. [Online]. http://eprint.iacr.org/2009/137

[Gar09] Flavio D. Garcia, Peter van Rossum, Roel Verdult, and Ronny Wichers Schreur, "Wirelessly Pickpocketing a Mifare Classic Card," in *30th IEEE Symposium on Security and Privacy*, Oakland, 2009, pp. 3-15.

[Har10] Matthew J. Harmon and Natascha E. Shawver, "A new challenge — Plugging security gaps," *ISO Focus+*, vol. 7, no. 4, pp. 18-20, April 2010.

[Har52] Donald B. Harris, "Radio Transmission Systems with Modulatable Passive Responder," U.S. Patent 2,927,321, August 16, 1952.

[ISO012] ISO/IEC JTC1/SC17/WG8, "Identification Cards – Contactless integrated circuit(s) cards – Proximity cards – Part 2:Radio frequency power and signal interface," ISO/IEC FCD 14443-2, 2001.

[ISO013] ISO/IEC JTC1/SC17/WG8, "Identification Cards – Contactless integrated circuit(s) cards – Proximity cards – Part 3:Initialization and anti-collision," Final Comitee Draft ISO/IEC FCD 14443-3, January 2001.

[Ker83] Auguste Kerckhoffs, "La cryptographie militaire," *Journal des sciences militaires*, vol. IX, pp. 5-83, 161-191, January, February 1883.

[KHG08] Gerhard de Koning Gans, Jaap-Henk Hoepman, and Flavio D. Garcia, "A Practical Attack on the MIFARE Classic," in *Lecture Notes in Computer Science*, vol. 5189, London, 2008, pp. 267-282.

[Lan05] Jeremy Landt, "The history of RFID," *IEEE potentials*, vol. 24, no. 4, pp. 8-11, October/November 2005.

[Noh08] Karsten Nohl, David Evans, Starbug, and Henryk Plötz, "Reverse-Engineering a Cryptographic RFID Tag," in *Proceedings of the 17th USENIX Security Symposium*, San Jose, 2008, pp. 185-193.

[Nxp04] NXP Semiconductors. (2004, November) Application Note: mifare® Interface Platform Type Identification Procedure. [Online]. http://www.nxp.com/acrobat_download2/other/identification/m018413.pdf

[Nxp09] NXP Semiconductors. (2009, July) AN10833: MIFARE Type Identification Procedure. [Online]. http://www.nxp.com/documents/application_note/AN10833.pdf

[Ros09] Peter van Rossum. (2009, May) Mifare Classic Troubles. [Online]. http://www.ict-forward.eu/workshop2/program/

[Ros091] Tomáš Rosa, "PicNic pro RFID-KV," *Sdělovací technika*, pp. 1-2, January 2009.

[Sto48] Harry Stockmann, "Communication by Means of Reflected Power," *PROCEEDINGS OF THE I.R.E.*, vol. 36, no. 10, pp. 1196-1204, October 1948.

[Ver09] Roela Verdulta. (2009, August) Classic Mistakes. [Online]. https://har2009.org/program/attachments/123_[HAR2009]-Roel.Verdult-Classic.Mistakes.pdf

8 Doporučená literatura

Z následující literatury nebylo v této práci citováno, přesto mi byla významným pomocníkem a domnívám se, že by případnému zájemci o dané téma neměla uniknout.

- Práce popisující rekonstrukci šifry Crypto1

  Flavio D. Garcia, Peter van Rossum, Roel Verdult, Ronny Wichers Schreur and Bart Jacobs, "Dismantling MIFARE Classic", in *13th European Symposium on Research in Computer Security (ESORICS 2008)*, October 2008.

- Český článek Tomáše Rosy zaměřený na Courtoisův útok

  Tomáš Rosa, "Svědectví o definitivním konci MIFARE Classic," *Sdělovací Technika*, pp. 16-19, August 2008.

- Diplomová práce Wee Hon Tana

  Wee Hon Tan, "Practical Attacks on the MIFARE Classic," MSc thesis on Imperial College London, September 2009.

- Dokumentace k integrovaným obvodům výrobce

  NXP Semiconductors. (January 2008) MF1ICS70: Functional Specification. [Online]. http://www.nxp.com/acrobat_download2/other/identification/M043541_MF1ICS70_Fspec_rev4_1.pdf

- Prezentace s mikroskopickými snímky transpondérů

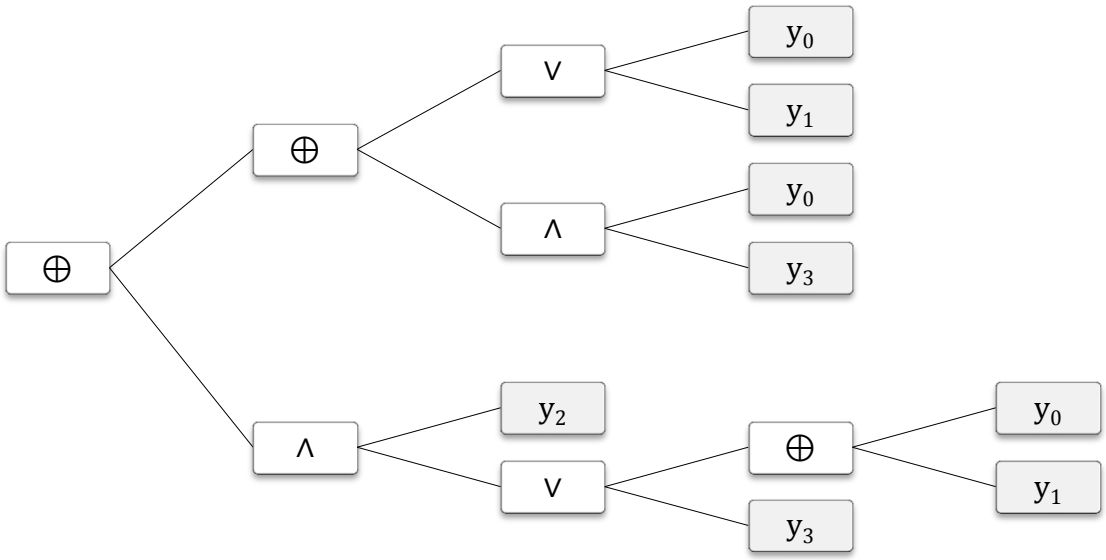
  Karsten Nohl, "MIFARE—Little Security, despite Obscurity" in *Black Hat*, Las Vegas, 2008. [Online]. https://www.blackhat.com/presentations/bh-usa-08/Nohl/BH_US_08_Nohl_Mifare.pdf

Citované dubnové vydání časopisu ISO Focus+ z roku 2010 je celé věnováno technologii RFID.

Mnohé další informace, ukázky a milníky v prolomení MIFARE jsou k dispozici na stránkách hlavních autorů:

Karsten Nohl	http://www.cs.virginia.edu/~kn5f/
Flavio D. Garcia	http://www.cs.ru.nl/F.Garcia/
Nicolas T. Courtois	http://www.cryptosystem.net/~courtois/

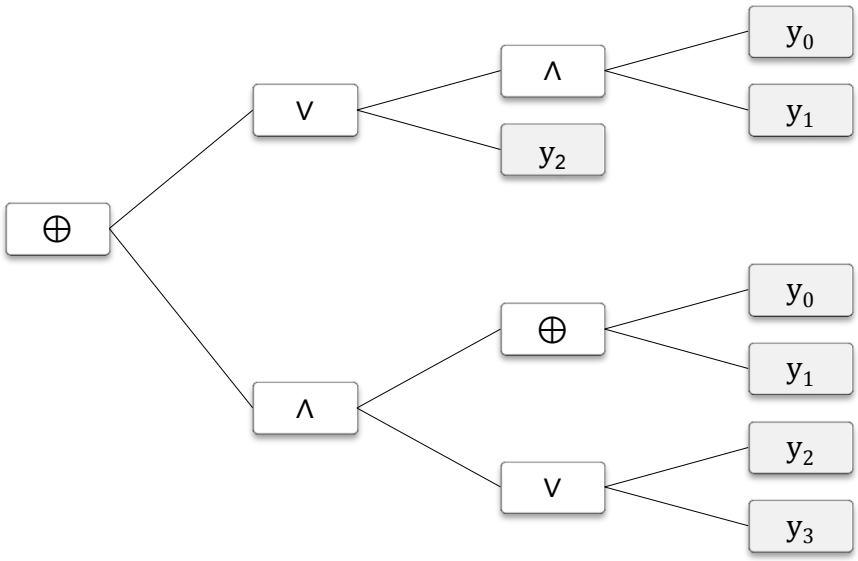
Příloha A. Funkce Crypto1



Obrázek 5.4- Strom funkce f_a

				A	B	C	D	E	F	E ⊕ F
y_0	**y_1**	**y_2**	**y_3**	$y_0 \vee y_1$	$y_0 \wedge y_3$	$y_0 \oplus y_1$	$C \vee y_3$	$A \oplus B$	$y_2 \wedge D$	**f_a**
0	**0**	**0**	**0**	0	0	0	0	0	0	**0**
0	**0**	**0**	**1**	0	0	0	1	0	0	**0**
0	**0**	**1**	**0**	0	0	0	0	0	0	**0**
0	**0**	**1**	**1**	0	0	0	1	0	1	**1**
0	**1**	**0**	**0**	1	0	1	1	1	0	**1**
0	**1**	**0**	**1**	1	0	1	1	1	0	**1**
0	**1**	**1**	**0**	1	0	1	1	1	1	**0**
0	**1**	**1**	**1**	1	0	1	1	1	1	**0**
1	**0**	**0**	**0**	1	0	1	1	1	0	**1**
1	**0**	**0**	**1**	1	1	1	1	0	0	**0**
1	**0**	**1**	**0**	1	0	1	1	1	1	**0**
1	**0**	**1**	**1**	1	1	1	1	0	1	**1**
1	**1**	**0**	**0**	1	0	0	0	1	0	**1**
1	**1**	**0**	**1**	1	1	0	1	0	0	**0**
1	**1**	**1**	**0**	1	0	0	0	1	0	**1**
1	**1**	**1**	**1**	1	1	0	1	0	1	**1**
P[... = 1]				$\frac{3}{4}$	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{3}{4}$	$\frac{1}{2}$	$\frac{3}{8}$	$\mathbf{\frac{1}{2}}$

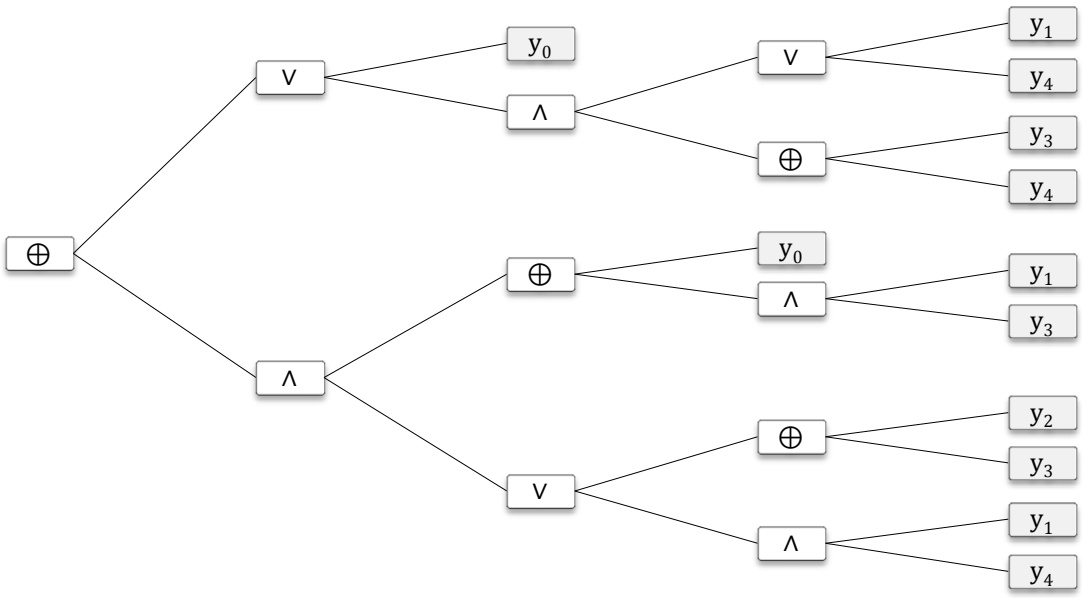
Tabulka 5.4- Hodnoty funkce f_a



Obrázek 5.4- Strom fukce f_b

				A	B	C	E	F	E ⊕ F
y_0	**y_1**	**y_2**	**y_3**	$y_0 \wedge y_1$	$y_0 \oplus y_1$	$y_2 \vee y_3$	$A \vee y_2$	$B \wedge C$	**f_b**
0	**0**	**0**	**0**	0	0	0	0	0	**0**
0	**0**	**0**	**1**	0	0	1	0	0	**0**
0	**0**	**1**	**0**	0	0	1	1	0	**1**
0	**0**	**1**	**1**	0	0	1	1	0	**1**
0	**1**	**0**	**0**	0	1	0	0	0	**0**
0	**1**	**0**	**1**	0	1	1	0	1	**1**
0	**1**	**1**	**0**	0	1	1	1	1	**0**
0	**1**	**1**	**1**	0	1	1	1	1	**0**
1	**0**	**0**	**0**	0	1	0	0	0	**0**
1	**0**	**0**	**1**	0	1	1	0	1	**1**
1	**0**	**1**	**0**	0	1	1	1	1	**0**
1	**0**	**1**	**1**	0	1	1	1	1	**0**
1	**1**	**0**	**0**	1	0	0	1	0	**1**
1	**1**	**0**	**1**	1	0	1	1	0	**1**
1	**1**	**1**	**0**	1	0	1	1	0	**1**
1	**1**	**1**	**1**	1	0	1	1	0	**1**
P[... = 1]				$\frac{1}{4}$	$\frac{1}{2}$	$\frac{3}{4}$	$\frac{5}{8}$	$\frac{3}{8}$	**$\frac{1}{2}$**

Tabulka 5.4- Hodnoty funkce f_b



Obrázek 5.4- Strom funkce f_c

					A	B	C	D	E	F	G	H	I	J	I ⊕ J
y_0	y_1	y_2	y_3	y_4	$y_1 \vee y_4$	$y_3 \oplus y_4$	$y_1 \wedge y_3$	$y_2 \oplus y_3$	$y_1 \wedge y_4$	$A \wedge B$	$y_0 \oplus C$	$D \vee E$	$y_0 \vee F$	$G \wedge H$	f_c
0	**0**	**0**	**0**	**0**	0	0	0	0	0	0	0	0	0	0	**0**
0	**0**	**0**	**0**	**1**	1	1	0	0	0	1	0	0	1	0	**1**
0	**0**	**0**	**1**	**0**	0	1	0	1	0	0	0	1	0	0	**0**
0	**0**	**0**	**1**	**1**	1	0	0	1	0	0	0	1	0	0	**0**
0	**0**	**1**	**0**	**0**	0	0	0	1	0	0	0	1	0	0	**0**
0	**0**	**1**	**0**	**1**	1	1	0	1	0	1	0	1	1	0	**1**
0	**0**	**1**	**1**	**0**	0	1	0	0	0	0	0	0	0	0	**0**
0	**0**	**1**	**1**	**1**	1	0	0	0	0	0	0	0	0	0	**0**
0	**1**	**0**	**0**	**0**	1	0	0	0	0	0	0	0	0	0	**0**
0	**1**	**0**	**0**	**1**	1	1	0	0	1	1	0	1	1	0	**1**
0	**1**	**0**	**1**	**0**	1	1	1	1	0	1	1	1	1	1	**0**
0	**1**	**0**	**1**	**1**	1	0	1	1	1	0	1	1	0	1	**1**
0	**1**	**1**	**0**	**0**	1	0	0	1	0	0	0	1	0	0	**0**
0	**1**	**1**	**0**	**1**	1	1	0	1	1	1	0	1	1	0	**1**
0	**1**	**1**	**1**	**0**	1	1	1	0	0	1	1	0	1	0	**1**
0	**1**	**1**	**1**	**1**	1	0	1	0	1	0	1	1	0	1	**1**
1	**0**	**0**	**0**	**0**	0	0	0	0	0	0	1	0	1	0	**1**
1	**0**	**0**	**0**	**1**	1	1	0	0	0	1	1	0	1	0	**1**
1	**0**	**0**	**1**	**0**	0	1	0	1	0	0	1	1	1	1	**0**
1	**0**	**0**	**1**	**1**	1	0	0	1	0	0	1	1	1	1	**0**
1	**0**	**1**	**0**	**0**	0	0	0	1	0	0	1	1	1	1	**0**
1	**0**	**1**	**0**	**1**	1	1	0	1	0	1	1	1	1	1	**0**
1	**0**	**1**	**1**	**0**	0	1	0	0	0	0	1	0	1	0	**1**
1	**0**	**1**	**1**	**1**	1	0	0	0	0	0	1	0	1	0	**1**
1	**1**	**0**	**0**	**0**	1	0	0	0	0	0	1	0	1	0	**1**
1	**1**	**0**	**0**	**1**	1	1	0	0	1	1	1	1	1	1	**0**
1	**1**	**0**	**1**	**0**	1	1	1	1	0	1	0	1	1	0	**1**
1	**1**	**0**	**1**	**1**	1	0	1	1	1	0	0	1	1	0	**1**
1	**1**	**1**	**0**	**0**	1	0	0	1	0	0	1	1	1	1	**0**
1	**1**	**1**	**0**	**1**	1	1	0	1	1	1	1	1	1	1	**0**
1	**1**	**1**	**1**	**0**	1	1	1	0	0	1	0	0	1	0	**1**
1	**1**	**1**	**1**	**1**	1	0	1	0	1	0	0	1	1	0	**1**
P[... = 1]					$\frac{3}{4}$	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{3}{8}$	$\frac{1}{2}$	$\frac{5}{8}$	$\frac{11}{16}$	$\frac{5}{16}$	$\mathbf{\frac{1}{2}}$

Tabulka 5.4- Hodnoty funkce f_c

Výpočet algebraických normálních forem funkcí:

Pro f_a:

$$(w_0, \dots, w_{1234}) = [f_a] \cdot A_4 =$$
$$= (0 \quad 0 \quad 0 \quad 1 \quad 1 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1) \cdot A_4 =$$
$$= (0 \quad 0 \quad 0 \quad 1 \quad 1 \quad 2 \quad 1 \quad 3 \quad 1 \quad 1 \quad 1 \quad 3 \quad 3 \quad 4 \quad 4 \quad 8) \equiv$$
$$\equiv (0 \quad 0 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 0 \quad 0 \quad 0)$$

Tedy

$$f_a = y_2y_3 \oplus y_1 \oplus y_1y_2 \oplus y_1y_2y_3 \oplus y_0 \oplus y_0y_3 \oplus y_0y_2 \oplus y_0y_2y_3 \oplus y_0y_1$$
$$= y_0 \oplus y_1 \oplus y_0y_1 \oplus y_0y_2 \oplus y_0y_3 \oplus y_1y_2 \oplus y_2y_3 \oplus y_0y_2y_3 \oplus y_1y_2y_3$$

Pro f_b:

$$(w_0, \dots, w_{1234}) = [f_b] \cdot A_4 =$$
$$= (0 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 1 \quad 1 \quad 1) \cdot A_4 =$$
$$= (0 \quad 0 \quad 1 \quad 2 \quad 0 \quad 1 \quad 1 \quad 3 \quad 1 \quad 1 \quad 1 \quad 3 \quad 1 \quad 4 \quad 3 \quad 8) \equiv$$
$$\equiv (0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1 \quad 0)$$

Tedy

$$f_b = y_2 \oplus y_1y_3 \oplus y_1y_2 \oplus y_1y_2y_3 \oplus y_0y_3 \oplus y_0y_2 \oplus y_0y_2y_3 \oplus y_0y_1 \oplus y_0y_1y_2$$
$$= y_2 \oplus y_0y_1 \oplus y_0y_2 \oplus y_0y_3 \oplus y_1y_2 \oplus y_1y_3 \oplus y_0y_1y_2 \oplus y_0y_2y_3 \oplus y_1y_2y_3$$

Pro f_c:

$$(w_0, \dots, w_{12345}) = [f_c] \cdot A_5 =$$
$$= (0 \ 1 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ 1 \ 1 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 1 \ 1 \ 0 \ 0 \ 1 \ 1) \cdot A_5 =$$
$$= (0 \ 1 \ 0 \ 1 \ 0 \ 2 \ 0 \ 2 \ 0 \ 2 \ 0 \ 3 \ 0 \ 4 \ 1 \ 7 \ 1 \ 3 \ 1 \ 3 \ 1 \ 4 \ 2 \ 6 \ 2 \ 5 \ 3 \ 8 \ 2 \ 7 \ 6 \ 16) \equiv$$
$$\equiv (0 \ 1 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 0)$$

Tedy

$$f_c = y_4 \oplus y_3y_4 \oplus y_1y_3y_4 \oplus y_1y_2y_3 \oplus y_1y_2y_3y_4 \oplus y_0 \oplus y_0y_4 \oplus y_0y_3 \oplus y_0y_3y_4 \oplus$$
$$\oplus y_0y_2 \oplus y_0y_1y_4 \oplus y_0y_1y_3 \oplus y_0y_1y_2y_4$$
$$= y_0 \oplus y_4 \oplus y_0y_2 \oplus y_0y_3 \oplus y_0y_4 \oplus y_3y_4 \oplus y_0y_1y_3 \oplus y_0y_1y_4 \oplus y_0y_3y_4 \oplus$$
$$\oplus y_1y_2y_3 \oplus y_1y_3y_4 \oplus y_0y_1y_2y_4 \oplus y_1y_2y_3y_4$$

Příloha B. Identifikace transpondérů MIFARE

Následující informace byly vybrány z [**Nxp09**] a [**Nxp04**].

Odpověď ATQA:

transpondér **bit ATQA**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
MIFARE Standard Mini	0	0	0	0	0	1	0	0
MIFARE Standard 1K	0	0	0	0	0	1	0	0
MIFARE Standard 4K	0	0	0	0	0	0	1	0
MIFARE Plus	0	0	0	0	0	1	0	0
	0	0	0	0	0	0	1	0
MIFARE ProX	0	0	0	0	1	0	0	0
	0	0	0	0	0	1	0	0
	0	0	0	0	0	0	1	0

Odpověď SAK:

transpondér **bit SAK**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
MIFARE Standard Mini	0	0	0	0	1	0	0	1
MIFARE Standard 1K	0	0	0	0	1	0	0	0
MIFARE Standard 4K	0	0	0	1	1	0	0	0
MIFARE Plus 2K	0	0	0	0	1	0	0	0
MIFARE Plus 4K	0	0	0	1	1	0	0	0
MIFARE Plus 2K (security level 2)	0	0	0	1	0	0	0	0
MIFARE Plus 4K (security level 2)	0	0	0	1	0	0	0	0
MIFARE Plus 2K (security level 3)	0	0	1	0	0	0	0	0
MIFARE Plus 4K (security level 3)	0	0	1	0	0	0	0	0
MIFARE ProX	0	0	0	0	1	0	0	0
	0	0	0	1	1	0	0	0

Uvedeny jsou pouze bezkontaktní typy transpondérů s 32bitovým UID.

Transpondéry MIFARE Plus nepodporují šifrovací schéma uvedené v této práci, jsou-li v režimu security level 2 nebo 3.

Příloha C. Výrobní klíče transpondérů

Při nákupu nového transpondéru jsou jeho sektory chráněny klíči, které nastavil jejich výrobce. Je až k nevíře, kolik aplikací je buď využívá beze změny, nebo používá některé z příkladů uváděných v dokumentaci.

výrobce	**Klíč A**	**Klíč B**
Philips (NXP)	A0 A1 A2 A3 A4 A5	B0 B1 B2 B3 B4 B5
Infineon	FF FF FF FF FF FF	FF FF FF FF FF FF

Další používané klíče:

00 00 00 00 00 00

AA BB CC DD EE FF

D3 F7 D3 F7 D3 F7 (Advanced Card Systems)

1A 98 2C 7E 45 9A

4D 3A 99 C3 51 DD

Zdroj: [**Ver09**] a přednáška Tomáše Rosy na MFF UK.