elektrorevue
ISSN 1213 - 1539

ČIPOVÉ KARTY MIFARE A JEJICH BEZPEČNOST

Ing. Radim Pust
Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií,
Ústav telekomunikací, Purkyňova 118, 612 00 Brno, Česká republika
Email: pust@feec.vutbr.cz

Autentizace pomocí kontaktních a bezkontaktních čipových karet je jeden z nejrozšířenějších způsobů dokazování identity. Článek je zaměřen na čipové karty Mifare a jejich bezpečnost.

1. ÚVOD

Čipy Mifare jsou na trhu poměrně hojně rozšířeny. Svoje využití našly především při placení služeb hromadné dopravy v řadě evropských měst. Také při řízení fyzického přístupu do budov či kanceláří. Čipy s označením Mifare vyrábí výhradně společnost NXP Semiconductor, která byla založena společností Philips. Čipy Mifare používají mimo jiné vlastní proprietární autentizační a šifrovací algoritmus Crypto, který je výrobcem utajován.

2. ČIPY MIFARE

Čipy Mifare jsou k dispozici na trhu v několika provedeních, porovnání jejich vlastností je uvedeno v tabulce 1, která vychází z [1].

Všechny uvedené čipy disponují bezkontaktním rozhraním dle ISO 14443-A.

U vyšších modelů se můžeme setkat s kontaktním rozhraním dle ISO-7816. Čipy typu Ultralight a Standard používají výhradně proprietární protokol a šifrování. Čip typu DESFire je již vybaven vlastním operačním systémem DESFire operating systém a podporou algoritmu DES a 3DES. Čipy typu ProX a SmartMX platform jsou vybaveny Java Card operating system (JCOP). Nižší modely používají na transportní vrstvě pouze vlastní proprietární řešení Mifare naproti tomu u vyšších modelů je již integrována podpora standardizovaného transportního protokolu dle ISO-14443-4.

Typ	Mifare Ultralight	Mifare Standard 1K/4K	Mifare DESFire	Mifare ProX Platform	Mifare SmartMX Platform
CPU	-	-	Tangram 80C51	Tangram 80C51	Tangram 80C51
HW podpora šifrování	-	Mifare Crypto	DES/3DES	Mifare Crypto, DES/3DES, PKI	Mifare Crypto, DES/3DES, AES, PKI
Velikost paměti EEPROM	512bit	1/4 KByte	4KByte	4/8/16 KByte	4-72 KByte
kontaktní rozhraní	ne	ne	ne	ISO-7816	ISO-7816
bezkontaktní rozhraní	ISO 14443-A	ISO 14443-A	ISO 14443-A	ISO 14443-A	ISO 14443-A
Podpora ISO-14443-3 (inicializace, antikolozní mechanismus)	ano	ano	ano	ano	ano
Podpora ISO-14443-4 (transportní vrstva)	ne	ne	ano, T=CL	ano, T=CL	ano, T=CL

TABULKA 1: PŘEHLED ČIPŮ MIFARE A JEJICH VLASTNOSTÍ.

elektrorevue
ISSN 1213 - 1539

3. Čip Mifare Standard

Čipy typu Mifare Standard nelze zařadit do kategorie smart card, jedná se spíše o „chytřejší“ paměťová uložiště. Komunikační rozhraní tohoto čipu je řešeno dle standardu ISO 14443-A. Samotná inicializace karty a antikolizní procedura probíhá dle standardu ISO-14443-3. Dále již probíhá komunikace pomocí proprietárního protokolu. Nejprve proběhne třícestná autentizace mezi terminálem a kartou během níž je ustanoven dočasný klíč.

Tento klíč slouží k šifrování následné komunikace proprietárním symetrickým šifrovacím algoritmem Crypto. Po úspěšně provedené oboustranné autentizaci lze vykonávat potřebné příkazy.

Organizace paměti na čipu je řešena rozdělením do bloků po 16 Bytech. Určitý počet bloků je sdružován do tzv. sektoru. Počet přidružených bloků do jednoho sektoru je odlišný pro jednotlivé typy karet. U karty Standard o velikosti 1K, je paměť rozdělena na 16 sektorů po 4 datových blocích. U karty Standard o velikosti 4K, je prvních 32 sektorů po 4 datových blocích a následujících 8 sektorů po 16 datových blocích. Na začátku každého sektoru je jeden blok vyhrazen jako tzv. zavaděč sektoru viz tabulka 2.

…	…	…
…	…	…
0x07	Klíč A, přístupová práva, Klíč B	Sektor 0x01
0x06	Datový blok	
0x05	Datový blok	
0x04	Datový blok	
0x03	Klíč A, přístupová práva, Klíč B	Sektor 0x00
0x02	Datový blok	
0x01	Datový blok	
0x00	UID,BCC, atd.	

Tabulka 2: Organizace paměti [2].

Zavaděč sektoru slouží pro uložení dvou klíčů A a B a nastavení práv k bloku. Práva k bloku jsou nastavena pomocí indexu, který určuje typ požadovaného klíče pro provedení daného příkazu, viz tabulka 3. Zvláštní význam má blok 0 ve kterém jsou pevně uloženy údaje od výrobce jako je například identifikátor karty UID (Unique IDentifier) a jeho kontrolní součet BCC (Bit Count Check), který vzniká provedením operace XOR nad všemi bity UID.

Index	Příkazy			
	čtení	zápis	inkrementace	dekrementace, přenos, obnovení
0	A / B	A / B	A / B	A / B
1	A / B	nelze	nelze	nelze
2	A / B	B	nelze	nelze
3	A / B	B	B	A / B
4	A / B	nelze	nelze	A / B
5	B	B	nelze	nelze
6	B	nelze	nelze	nelze
7	nelze	nelze	nelze	nelze

Tabulka 3: Možnosti nastavení přístupových práv bloku [3].

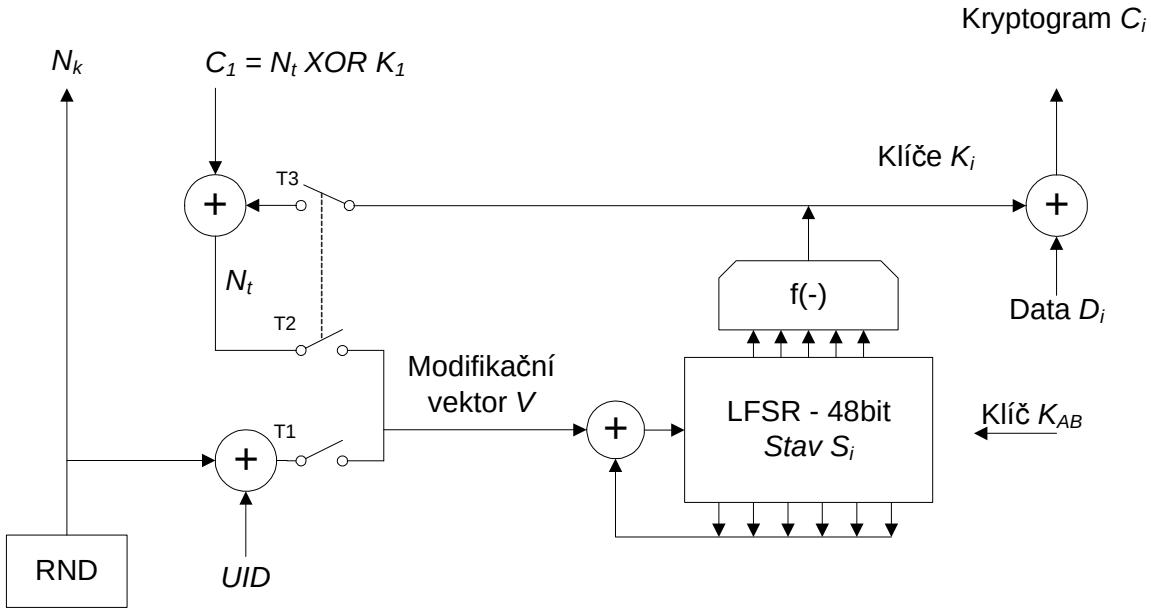
Klíče A a B mají shodnou délku 48 bitů. Pro každý sektor mohou být voleny odlišné klíče i odlišné nastavení způsobu řízení bloku, tím je zajištěno oddělení jednotlivých aplikací. Volné datové bloky mohou být datového nebo číselného typu. U datového bloku lze uložit řetězec o délce 16 Bytů. U číselného typu je dané číslo uloženo v bloku celkem třikrát, dvakrát v neinvertované podobě (2x4B) a jednou v invertované podobě 4B a poslední 4B slouží k uložení adresy, která může být použita jako ukazatel [2]. Továrně je klíč A nastaven na hodnotu A0A1A2A3A4A5 a klíč B na B0B1B2B3B4B5 [3]. Z bezpečnostních důvodů je vhodné továrně přednastavené klíče změnit.

Počet příkazů u Mifare je poměrně strohý jedná se o příkazy umožňující čtení a zapisování do bloků inkrementaci a dekrementaci číselné hodnoty, obnovení původní hodnoty a přesun do jiného bloku [2], [4].

4. Autentizace a šifrování Mifare

Vědci Karsten Nohl a Henryk Plotz za použití mikroskopu pořídili snímky jednotlivých vrstev čipu a pomocí specializovaného softwaru odhalili podobu neveřejného algoritmu Crypto viz [5], [6]. Vědec F.D.Garcia a jeho kolektiv analýzou komunikace mezi kartou a čtečkou přesně popsali šifrovací a autentizační algoritmus Crypto viz [2], [7]. Následující poznatky vycházejí z výše uvedených dokumentů.

elektrorevue
ISSN 1213 - 1539



Obrázek 1: Algoritmus Crypto na straně karty ([5],[6]).

Na obrázku 1 je vyobrazen algoritmus Crypto, který je kromě hradel XOR složen z generátoru náhodných čísel RND, 48-bitového registru LFSR (Linear Feedback Shift Register - posuvný registr s lineární zpětnou vazbou), filtru f a tří spínačů (T1, T2 a T3). Vědcům se podařilo na základě získaných znalostí a pokusů odvodit polynom u 48-bitového registru LFSR, polynom náhodného generátoru RND a funkci filtru f.

Náhodný generátor RND je tvořen 16-bitovým registrem LFSR, který generuje náhodná čísla o délce 32 bitů. Pomocí generátorů RND jsou generována náhodná čísla N_k a N_t. Generátor také slouží pro výpočet tzv. následníků N_k' a N_k". Následníci jsou vypočítáváni v průběhu autentizace z náhodného čísla karty N_k, tak že generátor RND je iniciován náhodným číslem N_k a následně je provedeno 32 posuvů k získání N_k' respektive 64 posuvů pro získání N_k".

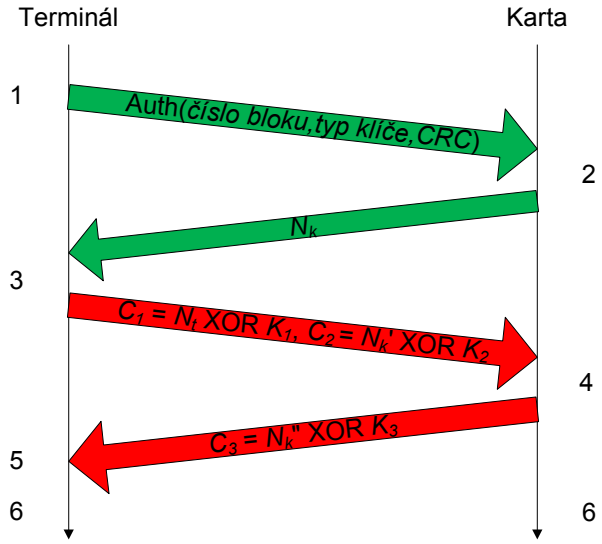
Algoritmus Crypto šifruje v průběhu autentizace po datových blocích o délce 32 bitů. Šifrování provádíme, tak že kryptogram $C_i = D_i$ XOR K_i, kde D_i je i-tý blok dat a K_i je klíč pro šifrování i-tého bloku dat. Klíč K_i je získán z výstupu filtrační funkce f po provedení 32 posuvů v LFSR. Z každého stavu LFSR je získán funkcí f právě jeden bit klíče, čímž vznikne po 32 posuvech registru LFSR klíč o délce 32 bitů. Formálně vygenerování klíče zapisujeme K_i = F (S_i, V), kde funkci F tvoří filtr f a registr LFSR, přičemž S_i je počáteční stav registru LFSR pro výpočet i-tého klíče a V je modifikační vektor se kterým bude provedeno následujících 32 posuvů registru vedoucích k získání klíče K_i. Pomocí modifikačního vektoru V dochází k ovlivňování vazby registru LFSR funkcí XOR V. Ve formálním zápisu hodnota 0 na místě modifikačního vektoru V znamená, že vazba registru nebude modifikována. Registr LFSR je na začátku autentizace iniciován příslušným klíčem K_{AB} (tj. buď A nebo B pro daný blok). Terminál má příslušný klíč K_{AB} uložen ve své paměti a karta klíč získá přečtením zavaděče požadovaného sektoru. Funkci LFSR formálně zapisujeme pomocí funkce G výrazem S_{i+1} = G (S_i, V), kde S_i je počáteční stav registru LFSR a S_{i+1} konečný stav registru LFSR po provedení 32 posuvů.

Modifikační vektor V je vkládán na obrázku 1 pomocí spínačů T1-T3. Výchozím stavem uvedených spínačů je stav rozepnuto. Spínač T1 je v průběhu autentizace sepnut po dobu generování náhodného čísla N_k. Vazba registru LFSR je pomocí spínače T1 modifikována náhodným číslem karty N_k a identifikátorem karty UID. Spínače T2 a T3 jsou v průběhu autentizace sepnuty po dobu přijímání náhodného čísla N_t. Vazba registru LFSR je pomocí spínačů T2 a T3 modifikována náhodným číslem terminálu N_t, který je získán z kryptogramu C_1.

Inicializace čipu a antikolizní mechanismus proběhne dle standardu ISO-14443-3 typ A. Na konci antikolizního mechanismu, karta definitivně potvrdí svůj identifikátor a zároveň oznámí, že není kompatibilní s ISO-14443-4. Další komunikace s kartou již probíhá pomocí proprietárního protokolu Crypto. Následuje třícestný proces autentizace při němž dojde k autentizaci obou stran a inicializaci LFSR. Po ukončení autentizace jsou klíče pro následnou komunikaci generovány dle potřeby bez modifikace vektorem V. Synchronizace klíčů je na obou stranách zajištěna synchroním posouváním registru LFSR, ke kterému dochází vždy při příjmu respektive odesílání jednotlivých bitů.

elektrorevue
ISSN 1213 - 1539

Algoritmus Crypto řadíme mezi symetrické šifrovací systémy, jednotlivé výpočty na obou stranách (karta a terminál) jsou realizovány shodným postupem. Průběh autentizace je podrobně zobrazen na obrázku 2, kde zelené barva značí nešifrované spojení zatímco červenou barvou je značeno spojení již šifrované.



Obrázek 2: Průběh autentizace [7].

Popis protokolu:

1. Terminál zahájí autentizaci vysláním příkazu Auth jehož součástí je číslo požadovaného bloku, typ klíče (A nebo B) a kontrolní součet *CRC*. Terminál inicializuje LFSR klíčem K_{AB} tj. $S_0 = K_{AB}$.
2. Karta přijme příkaz Auth a také inicializuje svůj LFSR klíčem K_{AB}. Dále karta vygeneruje pomocí svého RND náhodné číslo N_k a odešle je terminálu. V průběhu vysílání N_k jím modifikuje stav svého LFSR. Po odeslání N_k se LFSR karty nachází ve stavu S_1 = G (S_0, N_k XOR *UID*) a zároveň se vygeneruje klíč K_1 = F (S_0, N_k XOR *UID*).
3. Terminál v průběhu příjmu N_k od karty jím modifikuje stav svého LFSR. Po přijetí N_k se LFSR terminálu nachází také ve stavu S_1 = G (S_0, N_k XOR *UID*) a zároveň byl vygenerován klíč K_1 = F (S_0, N_k XOR *UID*). Terminál dále pomocí svého generátoru RND vygeneruje svoje náhodné číslo N_t a z čísla N_k odvodí jeho následovníky N_k' a N_k''. Kartě poté odešle kryptogramy $C_1 = N_t$ XOR K_1 a $C_2 = N_k'$ XOR K_2, kde K_2 = F (S_1, N_t). Po jejich odeslání se LFSR terminálu nachází ve stavu S_2 = G (S_1, N_t).
4. Karta přijme C_1 a C_2. Podle K_1 dešifruje C_1 a tak získá N_t. Pomocí získaného N_t karta modifikuje stav svého LFSR do stavu S_2 = G (S_1, N_t) přičemž zároveň získá klíč K_2 = F(S_1, N_t). Podle K_2 dešifruje C_2 a pomocí svého generátoru RND ověří, zda dešifrovaná hodnota je rovna N_k'. V kladné případě vygeneruje N_k'' pomocí svého generátoru RND a zašifruje jej do kryptogramu $C_3 = N_k''$ XOR K_3, kde K_3 = F (S_2, *0*). Po odeslání se LFSR terminálu nachází ve stavu S_3 = G (S_2, *0*).
5. Terminál přijme kryptogram C_3, vygeneruje si klíč K_3 = F (S_2, *0*) a ověří, zda dešifrovaná hodnota je rovna N_k''.
6. Pokud je vše v pořádku jsou nyní obě strany navzájem autentizovány a generátory LFSR jsou na obou stranách shodně nastaveny. S jejich pomocí se dále generují sekvence bitů klíče pro vzájemnou komunikaci o délce podle aktuální potřeby.

5. Slabiny algoritmu Crypto

Součásti již zmíněných dokumentů [2], [7] nizozemských vědců je popis slabin za pomoci níž může být klíč K_{AB} odhalen. Jelikož je hodnota N_k přenášena nešifrovaně lze z kryptogramu C_2 zpětně odvodit funkcí XOR a výpočtem N_k' hodnotu klíče K_2 aniž bychom znali klíč K_{AB}.

Experimentálně bylo zjištěno, že většina terminálů při neodeslání odpovědi v bodě 4 po vypršení časového limitu reaguje kryptogramem C_3 = HALT XOR K_3. Binární podoba příkazu HALT je známa a lze tedy zpětně odvodit klíč K_3. Některé čtečky však místo příkazu HALT použijí příkaz čtení READ. U příkazu READ je však již nutné odhadnout i jeho parametry jako je číslo bloku ke zpětnému odvození klíče K_3.

Výše uvedeným způsobem lze získat klíče K_2 a K_3 bez znalosti klíče K_{AB} a bez karty s klíčem K_{AB}. V případě zachycení celé komunikace mezi terminálem a kartou s klíčem K_{AB} lze odvodit zpětně klíč K_3 výpočtem N_k'' i bez příkazů HALT nebo READ.

V dokumentu [7] jsou popsány dva typy útoku. První z popsaných útoků využije slabiny LFSR opakovanou autentizací pro různá N_k získává K_2 a K_3. Díky předem vypočtené tabulce stavů LFSR a k nim klíčů K_2, K_3 invertuje funkci f a nalezne stav LFSR a následně odvodí jeho počáteční stav čili tajný klíč K_{AB}. Druhý z popsaných útoků na základě znalosti klíče K_2 inverzní funkcí f určí množinu 65536 možných klíčů K_{AB} z nichž lze opakováním autentizace určit správný klíč K_{AB}. Získání klíče K_{AB} z terminálu umožní přístup útočníkovi k obsahu příslušných bloků u karet s klíčem K_{AB}.

6. Závěr

Na základě volně dostupných materiálů od vědců Karsten Nohl, Henryk Plotz a F.D. Garcia et al. je v článku popsán properietární protokol a šifra užívaná u karet Mifare. Na základě posledních poznatků jsou zde popsány principy možných útoků na karty typu Mifare. Popsané útoky je možné realizovat na základní typy karet Mifare, které jsou však i dnes stále velmi hojně užívány.

Potvrzuje se tak že proprietární protokoly mohou snadněji v sobě ukrývat bezpečnostní slabiny, které mohou být později odhaleny a zneužity. Domnívám se že pokud by byl takto navržený protokol od svého počátku veřejný pravděpodobně byla by jeho slabina odhalena mnohem dříve a nedošlo by k tak masovému rozšíření. Do

elektrorevue
ISSN 1213 - 1539

budoucna si snad lze jen přát, abychom se s podobně zabezpečenými protokoly setkávali pokud možno co nejméně.

LITERATURA

[1] Philips Semiconductors. Mifare interface platform [online]. 2004 [cit. 2009-02-21]. Dostupný z WWW: <http://www.nxp.com/acrobat_download/other/identification/m018413.pdf>.

[2] Gerhard de Koning Gans, Jaap-Henk Hoepman, Flavio D. Garcia. A Practical Attack on the MIFARE Classic [online]. [2008] [cit. 2009-02-21]. Dostupný z WWW: <http://www.proxmark.org/documents/Mifare_weakness.pdf>.

[3] Klíma Vlastimil, Rosa Tomáš. Bezkontaktní karty MIFARE [online]. 2007 [cit. 2009-02-21]. Dostupný z WWW: <http://crypto.hyperlink.cz/files/ST_2007_02_x_x.pdf>.

[4] Gemalto. Gemalto Mifare 4K Datasheet [online]. 2007 [cit. 2009-02-21]. Dostupný z WWW: <http://www.gemalto.com/products/hybrid_card_body/download/Mifare4K_datasheet.pdf>.

[5] Nohl Karsten, Starbug, Plötz Henryk. MIFARE SECURITY [online]. 2007 [cit. 2009-02-21]. Dostupný z WWW: <http://events.ccc.de/congress/2007/Fahrplan/attachments/1049_CCC-07-Mifare-v2.pdf>.

[6] Nohl, Karsten. Cryptanalysis of Crypto-1 [online]. [2008] [cit. 2009-02-21]. Dostupný z WWW: <http://www.cs.virginia.edu/~kn5f/pdf/Mifare.Cryptanalysis.pdf>.

[7] F.D. Garcia, et al. Dismantling MIFARE Classic [online]. [2008] [cit. 2009-02-21]. Dostupný z WWW: <http://www.sos.cs.ru.nl/applications/rfid/2008-esorics.pdf>.