

# Konec éry MIFARE

Rok 2008 byl pro celosvětově hojně rozšířené karty MIFARE Classic (často označované prostě jen MIFARE, viz ST 2/2007) doslova zlomovým. Několik výzkumných týmů téměř nezávisle na sobě totiž totálně zlomilo jejich bezpečnostní mechanismy. Vlastně se stalo to, co se stáť muselo – reverzním inženýrstvím byl zjištěn popis utajovaného algoritmu Crypto1, což následně strhlo lavinu útoků. Základním byl útok hrubou silou na krátkou délku klíče (48 bitů), avšak brzy se našla řada dalších zásadních trhlin. Namátkou uvedme slabý generátor náhodných čísel, nevhodný komunikační protokol a konečně i strukturální slabiny Crypto1 umožňující mnohem efektivnější útoky. Za všechny práce zde zmíníme [2], která popisuje zatím nejefektivnější druhy útoku, které jsou navíc jistou stavebnicí umožňující další rozšiřování. Autoři ukazují jak na základě interakce emulátoru karty se čtečkou či odposlechu oboustranné komunikace čtečky s řádnou kartou získat tajné kryptografické klíče, a to se složitostí zvládnutelnou na běžném kancelářském PC. V roli emulátoru lze přitom použít například náš PicNic s vhodně rozšířeným programem FakeUID (ST 1/2009). S ohledem na rozsah zde nebudeme výklad obsáhle provedené ve [2] opakovat. Pro manažery právě podané shrnutí stačí, zájemci o detaily si nejspíš stejně uvedenou literaturu rádi podrobně prostudují. Prostor našeho článku využijeme k tomu, abychom ukázali, že útoky z [2] lze ještě dále řádově zefektivnit. Využíváme fakt, že kód CRC a šifra jsou nevhodně použity. Tomuto tématu se totiž práce [2] ke své drobné škodě už nevěnuje.

## K využití CRC

Ukážeme využití linearitu proudové šifry (operátoru XOR) a cyklického kódu CRC [1] spolu s nevhodným pořadím jejich aplikace k získání rovnic o proudu hesla. Zdůrazníme, že nejen kvůli ochraně důvěrnosti se má šifrovat zásadně před aplikací CRC. Zde podaný rozbor má za cíl doplnit práci [2] o další způsob získání informace o výstupu z algoritmu Crypto1. Ten je následně vstupem procedur provádějících inverzi Crypto1 směrem k výchozímu tajnému klíči délky 48 bitů. Z důvodu jednoduchosti se budeme zabývat (s ohledem na sémantiku protokolu [2]) výhradně 4B datovým blokem tvořeným dvěma bajty příkazu následovanými dvěma bajty CRC. To celé je zašifrováno (přexorováno) aktuálním proudem hesla algoritmu Crypto1. Předvedený postup lze nicméně aplikovat i pro jiné délky bloku. Na okamžik nevěnujme pozornost konkrétním bitům a bajtům rádiově přenášených dat, ale soustředíme se na čistě algebraickou stránku vě-

ci. Zpracovávaná data (16 bitů příkazu) jsou pro účely CRC chápána jako polynom  $m(x)$  stupně max. 15 na  $GF(2)[x]$ . Ten je pak doplněn kódem CRC, čímž vznikne 32 bitů dat, tj. polynom stupně max. 31  $p(x) = m(x) * x^{16}$

$W_0 + W_{16} + W_{21} + W_{26} + W_{28} + W_{31}$	$= C_0 + C_{16} + C_{21} + C_{26} + C_{28} + C_{31} + 1$
$W_1 + W_{17} + W_{22} + W_{27} + W_{29}$	$= C_1 + C_{17} + C_{22} + C_{27} + C_{29} + 1$
$W_2 + W_{18} + W_{23} + W_{28} + W_{30}$	$= C_2 + C_{18} + C_{23} + C_{28} + C_{30}$
$W_3 + W_{19} + W_{24} + W_{29} + W_{31}$	$= C_3 + C_{19} + C_{24} + C_{29} + C_{31}$
$W_4 + W_{16} + W_{20} + W_{21} + W_{25} + W_{26} + W_{28} + W_{30} + W_{31}$	$= C_4 + C_{16} + C_{20} + C_{21} + C_{25} + C_{26} + C_{28} + C_{30} + C_{31}$
$W_5 + W_{17} + W_{21} + W_{22} + W_{26} + W_{27} + W_{29} + W_{31}$	$= C_5 + C_{17} + C_{21} + C_{22} + C_{26} + C_{27} + C_{29} + C_{31} + 1$
$W_6 + W_{18} + W_{22} + W_{23} + W_{27} + W_{28} + W_{30}$	$= C_6 + C_{18} + C_{22} + C_{23} + C_{27} + C_{28} + C_{30} + 1$
$W_7 + W_{19} + W_{23} + W_{24} + W_{28} + W_{29} + W_{31}$	$= C_7 + C_{19} + C_{23} + C_{24} + C_{28} + C_{29} + C_{31}$
$W_8 + W_{20} + W_{24} + W_{25} + W_{29} + W_{30}$	$= C_8 + C_{20} + C_{24} + C_{25} + C_{29} + C_{30} + 1$
$W_9 + W_{21} + W_{25} + W_{26} + W_{30} + W_{31}$	$= C_9 + C_{21} + C_{25} + C_{26} + C_{30} + C_{31} + 1$
$W_{10} + W_{22} + W_{26} + W_{27} + W_{31}$	$= C_{10} + C_{22} + C_{26} + C_{27} + C_{31}$
$W_{11} + W_{16} + W_{21} + W_{22} + W_{26} + W_{27} + W_{31}$	$= C_{11} + C_{16} + C_{21} + C_{22} + C_{26} + C_{27} + C_{31}$
$W_{12} + W_{17} + W_{22} + W_{24} + W_{27} + W_{28}$	$= C_{12} + C_{17} + C_{22} + C_{24} + C_{27} + C_{28}$
$W_{13} + W_{18} + W_{23} + W_{25} + W_{26} + W_{29}$	$= C_{13} + C_{18} + C_{23} + C_{25} + C_{26} + C_{29} + 1$
$W_{14} + W_{19} + W_{24} + W_{26} + W_{29} + W_{30}$	$= C_{14} + C_{19} + C_{24} + C_{26} + C_{29} + C_{30} + 1$
$W_{15} + W_{20} + W_{25} + W_{27} + W_{30} + W_{31}$	$= C_{15} + C_{20} + C_{25} + C_{27} + C_{30} + C_{31}$

Obr. 1 Soustava rovnic indukovaná nevhodným použitím CRC

–  $(m(x) * x^{16} \bmod g(x)) + s(x)$ , kde  $g(x) = x^{16} + x^{12} + x^5 + 1$  je generující polynom daného CRC a  $s(x) = x^{10} + x^8 + x^6 + x^5 + x^4 + x^3$  je, řekněme, známá pevná korekce syndromu (konstanta). Výpočet 32 bitů šifrovaného textu, tj. šifrovaného polynomu  $c(x)$  není nic jiného než binární načtení hesla na zdroj neboli načtení polynomu hesla na polynom dat:  $c(x) = p(x) + w(x)$ , kde  $w(x)$  je polynom aktuálního proudu hesla. Když odečteme korekci syndromu (konstantu) od šifrovaného textu, máme  $(c(x) - s(x)) \bmod g(x) = (p(x) - s(x) + w(x)) \bmod g(x) = w(x) \bmod g(x)$ . Na levé straně tak dostaneme známá data a na pravé straně informaci o hesle, konkrétně zbytek heslového polynomu po dělení generujícím polynomem (což je polynom stupně max. 15, tj. 16 bitů informace). Aniž bychom znali konkrétní hodnotu otevřeného textu, každý odposlechnutý zašifrovaný příkaz nám dává až 16 bitů informace o použitém hesle. Z hodnoty šifrovaného textu zasláného čtečkou do karty pak můžeme odvodit soustavu šestnácti nezávislých lineárních rovnic nad  $GF(2)$  o výstupu z Crypto1 na obr. 1. Indexování odpovídá pořadí rádiového přenosu jednotlivých bitů, počínaje pozicí 0. Vektor  $W$  je aktuální proud hesla, vektor  $C$  je proud šifrovaného textu s vynechanými paritními bity, jejichž využití se již práce [2] věnuje. Při spojení s metodami z [2] umožňuje jedna soustava z obr. 1 snížit entropii klíče až o 16 bitů. S ohledem na stavební charakter útoku tak můžeme buď zvýšit efektivitu stávajících přístupů, nebo útoky rozšiřovat směrem k získání tajného klíče na základě pouhého odposlechu terminálu. Připomeňme, že ten je běžně možný na desítky metrů daleko. Útočník tak může například ukrást klíče přístupového systému, aniž by překročil kamerový perimetr napadené budovy.

## Závěr

Z kryptografického hlediska jsou karty MIFARE Classic odepsané. Ačkoliv něco takového šlo minimálně s ohledem na délku klíče již delší dobu očekávat, jsou tou-

to skutečností téměř všichni dodavatelé aplikací RFID „zaskočení“. V důsledku potom nemají připravenou žádnou přesvědčivou alternativu. Karty s nepublikovanými algoritmy a protokoly už po této zkušenosti nelze za přesvědčivou alternativu považovat. Snad i proto jsou prolomené karty dál používány v nových systémech, jako by se nechumelilo. Zlepšení může přinést avizovaná platforma MIFARE Plus [3], která má být odpovědí na publikované útoky. Zatím ji však podle našich informací v praxi nikdo neviděl a výrobce (NXP) už údajně přiznal, že minimálně do poloviny tohoto roku se na tom nic nezmění. Útoky a odhalování slabin různých algoritmů nebo zařízení se publikují s cílem zabránit dalšímu používání slabých a nebezpečných metod k ochraně našich spotřebitelských aktivit (peněz, informací apod.). Dále proto, aby se vytvořil tlak na aktualizaci technologie na bezpečnější úroveň. Pokud v daném oboru neexistuje konkurence, která by náhradu nabídla, máme tu nepříjemný důsledek monopolu. Ovšem také známe jednoznačného primárního viníka, pokud někdo takové slabé aplikace napadne a zneužije například systémy vstupu do objektů, stravovací systémy, systém elektronických jízdenek...

Vlastimil Klíma, Tomáš Rosa,  
v.klima@volny.cz, tomas.rosa@rb.cz

## LITERATURA

- [1] Adámek, J.: *Kódování, SNTL Praha, 1989*
- [2] Garcia, F.-D., et al.: *Dismantling MIFARE Classic, ESORICS 2008, pp. 97-114, 2008*
- [3] [http://www.nxp.com/news/content/file\\_1418.html](http://www.nxp.com/news/content/file_1418.html)
- [4] E-archivy <http://cryptology.hyperlink.cz>, <http://crypto.hyperlink.cz>