VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV INTELIGENTNÍCH SYSTÉMŮ
DEPARTMENT OF INTELLIGENT SYSTEMS

BEZPEČNOSTNÍ ANALÝZA KARET MIFARE CLASSIC

SECURITY ANALYSIS OF MIFARE CLASSIC SMART CARDS

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE — **MARTIN BOBČÍK**
AUTHOR

VEDOUCÍ PRÁCE — **Ing. ONDŘEJ HUJŇÁK,**
SUPERVISOR

BRNO 2019

Vysoké učení technické v Brně
Fakulta informačních technologií

Ústav inteligentních systémů (UITS) Akademický rok 2018/2019

Zadání bakalářské práce

Student: **Bobčík Martin**
Program: Informační technologie
Název: **Bezpečnostní analýza karet Mifare Classic**
Security Analysis of Mifare Classic Smart Cards
Kategorie: Bezpečnost

Zadání:

1. Seznamte se s technologií RFID se zaměřením na chytré karty Mifare Classic.
2. Prostudujte dostupnou literaturu týkající se známých zranitelností karet Mifare Classic.
3. Nastudujte možnosti nástroje Chameleon Mini.
4. Demonstrujte zneužitelnost vybraných zranitelností a analyzujte rizika jejich využití v reálném prostředí.
5. Zhodnoťe dosažené výsledky a diskutujte jejich další využití pro bezpečnostní analýzy chytrých karet.

Literatura:

- YANG, Qing a Lin HUANG. *Inside radio: an attack and defense guide*. New York, NY: Springer Berlin Heidelberg, 2018. ISBN 978-981-10-8446-1.
- RANKL, Wolfgang a Wolfgang EFFING. *Smart Card Handbook*. Chichester, UK: John Wiley & Sons, 2010. DOI: 10.1002/9780470660911. ISBN 9780470743676.
- TUNSTALL, Michael. Smart Card Security. *Smart Cards, Tokens, Security and Applications*. Boston, MA: Springer US, 2008, s. 195-228. DOI: 10.1007/978-0-387-72198-9_9. ISBN 9780387721972.
- PARET, Dominique a Roderick RIESCO. *RFID and contactless smart card applications*. Chichester: John Wiley, 2005, 330 s. ISBN 0-470-01195-5.

Podrobné závazné pokyny pro vypracování práce viz http://www.fit.vutbr.cz/info/szz/

Vedoucí práce: **Hujňák Ondřej, Ing.**
Vedoucí ústavu: Hanáček Petr, doc. Dr. Ing.
Datum zadání: 1. listopadu 2018
Datum odevzdání: 15. května 2019
Datum schválení: 1. listopadu 2018

Abstrakt

Cílem této bakalářské práce je studium bezpečnosti chytrých, bezkontaktních karet MIFARE Classic a analýza rizik spojených s jejich využíváním. Popisuje jednotlivé známé zranitelnosti v návrhu těchto karet a jejich šifrovacího algoritmu CRYPTO1. V této práci je dále experimentováno se zařízením Chameleon Mini, s jehož pomocí jsou provedeny dva útoky a jedna kryptoanalýza karty. Konkrétně emulace karty, relay útok a analýza nedostatečné náhodnosti generátoru pseudonáhodných čísel karty. Z těchto se úspěšně podařila pouze emulace karet.

Abstract

Goal of this bachelor thesis is a security study of MIFARE Classic contactless smart cards and risk analysis of their usage. There are described individual vunerabilities in the design and CRYPTO1 cipher of such cards. In this thesis is also experimented with Chameleon Mini device, which is used to perform two attacks and one cryptoanalysis of the cards. Namely, card emulation, relay attack, and analysis of insufficient randomness of cards' pseudorandom number generator. From those, only card emulation was fully successful.

Klíčová slova

RFID technologie, chytré karty, bezkontaktní karty, MIFARE Classic, šifra CRYPTO1, zranitelnosti, Chameleon Mini, emulace karet, relay útok, kryptoanalýza, C# .Net

Keywords

RFID technology, smart cards, contactless cards, MIFARE Classic, CRYPTO1 cipher, vunerabilities, Chameleon Mini, card emulation, relay attack, cryptoanalysis, C# .Net

Citace

BOBČÍK, Martin. *Bezpečnostní analýza karet Mifare Classic*. Brno, 2019. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Ondřej Hujňák,

Bezpečnostní analýza karet Mifare Classic

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Ondřeje Hujňáka. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Martin Bobčík
10. května 2019

Poděkování

Tímto bych velmi rád poděkoval Ing. Ondřeji Hujňákovi za poskytnutý čas, cenné rady a odborné vedení při řešení této práce.

Obsah

Kapitola 1

Úvod

Tato bakalářská práce se zabývá chytrými kartami Mifare Classic. Tyto karty jsou nejrozšířenější ve stále se zvyšujícím počtu bezkontaktních karet. Většina typů karet je navržena s požadavkem na co nejnižší cenu. To má za následek snížení zabezpečení těchto karet. Vyjímkou nejsou ani karty Mifare Classic. Navzdory tomu, že se jedná o jednu z nejrozšířenějších karet na světě, obsahuje velké množství zranitelností umožňující například kopírování obsahu karty. Tyto karty byly představeny v roce 1996. Netrvalo však dlouho a byly představeny první útoky, pomocí nichž je možné měnit obsah karty s pomocí jednoduchých čtecích zařizení. Hromadné využívání těchto útoků vedlo nákladným změnám bezpečnostních systémů mnoha firem. Vzhledem k tomu, že karty Mifare Classic se hojně používají dodnes jako způsob identifikace osob, je toto téma stále zajímavé a aktuální. V této práci jsou vybrány tři různé zranitelnosti a jsou demonstrovány za pomoci nástroje Chameleon Mini. Kompaktní rozměry spolu s dostupností tohoto zařízení umožňují nejen penetrační testování přístupových systémů, ale také zneužitelnost osobami s nekalými úmysly.

Cílem práce je nejprve nastudovat technologi RFID, tedy technologii umožňující bezkontaktní komunikaci. Dále karty Mifare Classic se zaměřením na zabezpečení a zranitelnosti těchto karet. Praktickou částí je demonstrace vybraných zranitelností těchto karet. K demonstraci slouží zařízení Chameleon Mini, které bylo také nutné nastudovat. Po praktické části následuje analýza rizik využití provedených demonstrací v reálném prostředí.

Členění práce

Druhá kapitola se zabývá principy technologie RFID, která je využívána jako komunikační rozhraní těchto karet. Je nastíněna stručná historie této technologie, využívané komponenty jako RFID čipy a čtecí zařízení a jiné využití v praxi.

V další kapitole je nejprve ujasněno, co to jsou chytré karty, následuje popis produktové značky Mifare a nakonec jsou podrobně popsány samotné karty Mifare Classic, jejich struktura paměti, šifrovací algoritmus a autentizační a komunikační protokol.

Čtvrtá kapitola popisuje vybrané zranitelnosti v návrhu karet Mifare Classic a jejich šifrovacím algoritmu CRYPTO1.

Pátá kapitola představuje zařízení Chameleon Mini, jeho schopnosti a podporované příkazy. V následující kapitole je dokumentováno testovací prostředí, ve kterém budou útoky provedeny. Kromě toho také nástroje vytvořené k usnadnění práce se zařízením Chameleon Mini.

Sedmá kapitola se zaobírá jednotlivými vybranými útoky. Je rozdělena na popis kryptografické analýzy, útok pomocí emulace karet a relay útok. Každá podkapitola obsahuje teoretický rozbor útoku a jeho průběh, nastavení zařízení Chameleon Mini a případnou implementaci útoku. Následuje samotné provedení útoku s popisem jeho průběhu a vyhodnocení. Vyhodnocení se také věnuje analýze provedení dané demonstrace v reálném prostředí.

Kapitola 2

Radio frekvenční identifikace

V této kapitole jsou popsány principy komunikace pomocí radio frekvenční identifikace (dále jen RFID), její historie, jednotlivé komponenty a využití v průmyslu a každodenním životě.

2.1 Úvod do RFID

RFID je zkratka pro Radio-Frequency IDentification, tedy identifikace pomocí rádiové frekvence (dále jen RF). Tato technologie umožňuje bezdrátovou komunikaci na relativně krátkou vzdálenost pomocí elektromagnetického pole. Jako RFID jsou popsány různé přístupy k identifikaci objektů, které pracují na různých frekvencích. Nejedná se tak o jedinou technologii, nýbrž o celý soubor identifikačních technologií. Různé alternativy jsou popsány mezinárodními standardy jako například ISO 14443, ISO 15693 nebo ISO18092. Technologie využívána k automatické identifikaci objektů, které jsou spojeny s RFID značkami, neboli tagy. Tyto tagy po přiblížení odpovídají RFID čtecímu zařízení. Vzdálenost na jakou mohou tag a čtecí zařízení komunikovat se liší na základě použité frekvence. Tagy ani čtecí zařízení většinou neobsahují žádnou logiku. Zprávu o načtení tagu předá čtecí zařízení back-end systému, který informaci zpracuje.[4].

2.2 Historie

Principy RFID byly poprvé použity v systému IFF (Identity: Friend or Foe) za 2. světové války Britskou armádou. Tento systém měl za úkol rozlišit vlastní letadla od nepřátelských. Proto byla vybavena nastavitelným radio-majákem, který byl schopen vysílat 6 identifikačních kódů. Systém pracoval v principu tak, že vysílač (radar) vyslal dotaz směrem k letadlu. Po dosažení a zpracování signálu odpovídač (transpondér) na letadle vyslal signál zpět, čímž došlo k zjištění příslušnosti stroje. Transpondér mohl odpovídat dvěma způsoby. Pasivní systém využil odražený původní signál a upravil ho tak aby obsahoval potřebné informace. Tento princip je dnes nejvíce využívaným způsobem identifikace pomocí RFID. Naopak aktivní systém přijal signál a sám okamžitě odeslal odpověď, přičemž ta mohla být odeslána i na jiné nosné frekvenci. V padesátých letech minulého století se radio identifikace rozmohla z armády do celého letectví a používá se dodnes.

RFID vznikla jako alternativa k čárovým kódům. I když je výroba čárových kódů levnější, mají proti RFID mnoho nevýhod. Pro čtení musí čtecí zařízení přímo vidět na štítek s kódem. Nesmí být snížená jeho vizuální čitelnost například špínou popisovačem nebo po-

Pásmo	Frekvence	Dosah	Přenosová rychlost
LF	125-135 kHz	1-2 m	100 bps
HF	13,56 MHz	2 m	2 kbps
UHF-Evropa	865-868 MHz	12-20 m	40-640 kbps
UHF-Severní Amerika	902-928 MHz	12-20 m	40-640 kbps

Tabulka 2.1: Porovnání vlastností operačních pásem RFID[28][4]

kroucením. Zápis vícero informací se dá řešit pouze zvětšením plochy štítku nebo použitím jemnějšího značení, které je ale viditelné z menší vzdálenosti. Modifikace dat uložených pomocí čárových kódů se dá prakticky řešit pouze tiskem nového kódu [4][12].

2.3 Tagy

Primární využití RFID tagů spočívá v identifikaci objektů. Cena takových objektů je nesrovnatelně vyšší oproti ceně tagu. Pokud je značený objekt levný, tag musí být ještě levnější. Na rozdíl od RFID čteček jsou tagy prakticky pořád v pohybu, ať už jako chytré karty, nebo identifikátory zboží, vlaků kontejnerů apod. Tagy tedy musí být velmi levné za vysoké odolnosti proti fyzickému poškození[4]. RFID tag je systém skládající se minimálně z mikročipu, antény a pouzdra. Mikročip obsahuje pamět a logické obvody pro přijímání a odesílání dat čtecímu zařízení. Anténa přijímá signál z čtečky a poté jej zpětně rozptýlí (dále jen backscatter modulace) odesílanými daty. Pouzdro je potřeba pro udržení integrity tagu a jako ochrana proti vnějšímu poškození samotného čipu a antény[28]. RFID tagy se dělí na aktivní, pasivní a částečně pasivní podle toho, zda je jsou napájeny z externího nebo interního zdroje[4]. Rozdíl je také v tom, kdo iniciuje komunikaci. Aktivní tag komunikaci zahajuje sám, zatímco komunikaci s pasivními tagy musí zahájit sama čtečka[10].

2.3.1 Pasivní tagy

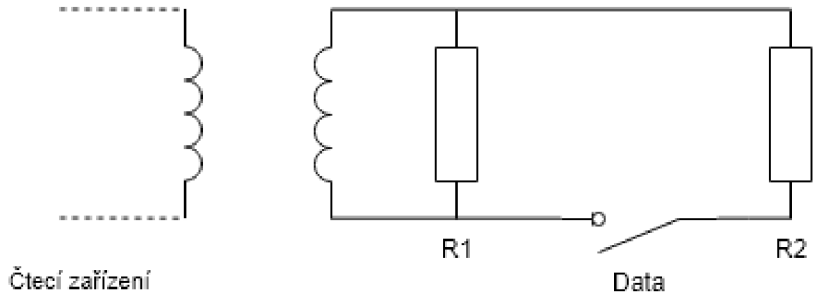
Pasivní tagy identifikují levné objekty. Interní zdroj, i ve formě malé baterie, je pro ně příliš velký a drahý. Stejně jako transmittery a přijímače používané v klasických radiových zařízeních. Bez konvenčního zdroje energie jsou prakticky použitelné pouze jednoduché obvody, které je možno napájet bezdrátově i na vzdálenost několika metrů od čtecího zařízení.

Aby mohl integrovaný obvod tagu pracovat, potřebuje zdroj stejnosměrného proudu několika desítek mikroampérů o napětí jeden až tři volty v závislosti na typu použitých tranzistorů. Toto napětí musí tag získat z RF signálu[4]. Frekvence tohoto signálu se podle použití liší. Od použité frekvence se také odvíjí dosah čtení a rychlost přenosu dat. Používané frekvence spadají do několika pásem, a to nízká frekvence (Low Frequency - LF), vysoká frekvence (High Frequency - HF) a ultra vysoká frekvence (Ultra High Frequency - UHF). Evropská a americká specifikace pásma UHF se liší ve frekvenci (viz. tabulka 2.1). Chytré karty používají HF pásmo[18].

Energie se přenáší na principu takzvaných volně vázaných transformátorů. Cívka ve čtecím zařízení generuje vysokofrekvenční elektromagnetické pole s vlnovou délkou 22m až 2400m. Vlnová délka nosné vlny je tedy mnohem delší než vzdálenost karty. To umožňuje volné vázání. Magnetické pole cívky v okolí čtecího zařízení indukuje napětí v cívce karty. Toto napětí je použito k napájení cívky. Účinnost tohoto modelu je velmi malá.[24]

K přenosu dat čtecího zařízení kartě se využívá klíčování s amplitudovým, frekvenčním i fázovým posuvem. Tedy reprezentace digitálních dat pomocí příslušné modulace nosné

vlny. K přenosu dat opačným směrem od karty ke čtecímu zařízení se využívá amplitudová modulace. Karta v poli čtecího zařízení využívá jeho energie. Změnou zatížení cívky v kartě lze měnit napětí na cívce čtecího zařízení. Tyto změny lze detekovat a interpretovat jako data. Změna zátěže v kartě je generována připojováním nebo odpojováním rezistoru z obvodu karty na základě datového signálu.[24]



Obrázek 2.1: Princip přenosu dat karty

2.3.2 Aktivní tagy

Aktivní tagy jsou napájeny z vlastního zdroje. Ten může být buď baterie, nebo připojení do elektrické infrastruktury. Zdroj napájí nejenom přenos dat, ale i ostatní elektronické komponenty. Těmito komponenty mohou být různé senzory nebo uživatelské rozhraní[28]. Vzhledem k technologii logických obvodů postupuje vývoj baterií velmi pomalu. Jedním z hlavních problémů návrhu těchto tagů je tedy zkrácení doby aktivity a snížení energie potřebné jak pro aktivní, tak pro klidové období tagu. Technologické skloubení těchto požadavků není jednoduché a výroba tagů se do jisté míry podobá výrobě běžných rádiových zařízení. Diskrétní komponenty a integrované obvody připájené k tištěným spojům, to celé připojené k anténě a uložené v plastovém krytu[4]. Částečně pasivní tagy obsahují vnitřní zdroj pouze k napájení pomocných komponent. Data jsou přenášena pomocí backscatter modulace jako u pasivních tagů[16].

2.4 Čtecí zařízení

Pro rádiovou komunikaci s tagy se používá čtecí zařízení, které funguje jako vysílač i přijímač dohromady. Taková zařízení komunikují buď plným, nebo poloviční duplexem. Poloduplexní spojení znamená, že zařízení nemůže přijímat a vysílat zároveň. Současný obousměrný přenos podporuje plný duplex. Pro komunikaci s pasivními tagy se používá právě plný duplex. Čtecí zařízení musí vysílat RF signál pro napájení tagu a zároveň přijímat odpověď. Nedílnou součástí čtecího zařízení je anténa. Vysílač i přijímač mohou mít každý svou anténu. Tato konfigurace je známá jako bistatická. Monostatický systém používá jednu anténu pro vysílání a přijímání signálu zároveň. V tomto případě je příjímač vystaven signálu z vysílače. Přijímač tedy musí být navržený tak, aby rozeznal signál z tagu[4].

2.5 Použití

Technologie RFID nabízí mnohá využití v různých oblastech lidské činnost. Nejčastější je prevence chyb. Například ve zdravotnictví každý pacient dostane náramek obsahující

RFID čip s pamětí. Do paměti jsou pak ukládány informace o pacientovi, jeho patologie, předepsané a podáné léky a jiné skutečnosti. Jedním z nejznámějších využití je sledování zboží a ochrana proti jeho krádeži. Zboží je opatřeno identifikátorem, který obsahuje data o přijetí do skladu či obchodu. Vstupy a výstupy budovy jsou opatřeny čtecími zařízeními, které registrují procházející zboží. Stejné značky lze využít v pokladnách při prodeji zboží. Značka obsahuje cenu, kterou odečte čtecí zařízení pokladny a přičte ji na účet zákazníka. Zároveň může být na zboži deaktivována ochrana, která by způsobila poplach při průchodu branou vybavenou čtečkou bez zaplacení.[8]

Kapitola 3

Chytré karty Mifare Classic®

V této kapitole jsou popsány obecně chytré karty a jejich typy. Následuje popis komunikačních standardů chytrých karet a představení značky Mifare. Následuje podrobný popis karet Mifare Classic, jejich struktura paměti, využitý algoritmus CRYPTO1, autentizační fáze karty a její komunikační protokol.

3.1 Co jsou chytré karty

Chytré karty, jak už název napovídá, jsou karty, které mohou být použity chytrým způsobem. Za chytrou kartu považujeme plastovou kartu s integrovaným obvodem. Takové karty lze obecně považovat za čipové karty a ty mohou být v kontaktní nebo bezkontaktní variantě. Kontaktní karty se vkládají do čtecích zařízení, které naváže fyzický kontakt se zlatě zbarveným rozhraním karty. Bezkontaktní karty stačí dostat do blízkosti čtecího zařízení. Napájení a komunikace je poté zajištěna pomocí RFID. Pod pojmem "chytrá karta" si obyčejně člověk představí platební kartu. Chytré karty mohou ale být použity i ve formě elektronických pasů, občanských průkazů, přístupových karet, jsouvhodné jako zabezpečení proti krádeži apod. Ne každá karta vyžaduje stejné zabezpečení. Některé ho nevyžadují vůbec, například jednoduché tagy vysílající své ID jako inventarizační systém ve skladech. Jiné vyžadují složité kryptografické mechanismy zajišťující soukromí přenášených dat.

Jednoduchý tag je z těchto produktů nejméně zabezpečený. Je nastaven tak, že své unikátní identifikační číslo (dále jen UID), které vysílá, je pouze pro čtení (Read-only). Kromě vysílání svého UID nemá žádný jiný protokol, je tedy jednoduché odposlouchávat komunikaci a replikovat ji pomocí emulátoru. Místo speciálního emulátoru lze použít podobný tag, který umožnuje změnu svého UID.

Paměťové tagy, stejně jako jednoduché, mají UID, navíc ale obsahují paměť. Tato paměť je jak pro čtení, tak pro zápis (Read/Write). Její přenos není nijak šifrován, což může vést k neoprávněnému čtení nebo emulaci. Výrobce karet může ovšem data opatřit ochranou integrity, například pomocí MAC (z anglického Message Authentication Code). Tato metoda zabrání útočníkovi ve vytváření nových a modifikaci stávajících dat. Je ale nutné opatřit čtecí zařízení funkcionalitou a klíči k ověření MAC[17].

MAC je krátká informace odesílána s daty, která dokáže zajistit integritu a autenticitu dat. To znamená ověřit, že data byla odeslána důvěryhodným odesílatelem a nebyla nijak modifikována. Jedná se o metodu výpočtu kontrolního součtu s pomocí symetrické kryptografie. Tato metoda se skládá ze tří algoritmů. První algoritmus generuje náhodné

klíče z množiny klíčů. Druhý algoritmus na základě vstupních dat a sdíleného tajného klíče vygeneruje MAC. A třetí algoritmus pomocí klíče a MAC verifikuje přijatá data[7].

Tag se zabezpečenou pamětí implementuje nějaký kryptografický protokol pro správu přístupu k paměti. Tag a čtecí zařízení se nejprve vzájemně autentizují a teprve poté je povolen přístup k paměti. Data z paměti jsou před odesláním většinou nejprve zašifrována relačním klíčem, aby byla zajištěna jejich bezpečnost. Paměť některých tagů je rozdělena do menších oddílů, z nichž každý má vlastní klíč. Toto uspořádání umožňuje použít jeden tag k více aplikacím, přičemž každé aplikaci náleží jiný klíč.

Nejpokročilejšími jsou zabezpečené tagy s mikrokontrolerem, který umožňuje nahrát různou funkcionalitu. Tyto tagy obvykle implementují bezpečnostní standardy jako například Global Platform. Šifrování, verifikace a další symetrické i asymetrické kryptografické metody jsou zajištěny kryptografickými koprocesory přímo na tagu[17]. Správný chod mikrokontroleru je zabezpečen například senzory vysokého a nízkého napětí, teploty nebo frekvence, filtrem vstupu hodinového signálu nebo aktivním stíněním[22].

3.2 Standardy komunikace chytrých karet

Standard ISO 14443 je určena zejména pro identifikační a platební karty. Skládá se ze čtyř částí. První část s označením ISO 14443-1 popisuje zejména fyzikální charakteristiky karty, její velikost a odolnost proti mechanickému namáhání a působení elektrických a magnetických polí. Druhá část (ISO 14443-2) udává charakteristiky elektromagnetického pole, které zajišťuje napájení a obousměrnou komunikaci mezi čtecím zařízením a kartou. Čtecí zařízení je zde označováno jako PCD, tedy Proximity Coupling Device, a karta jako PICC, což je zkratka pro Proximity Integrated Circuit Card. Zde jsou také definovány dvě metody přenosu dat, typ A a typ B. Ty se liší v kódování a v modulaci frekvencí. Třetí část (ISO 14443-3) uvádí, jak má čtecí zařízení postupovat při inicializaci komunikace s kartou, tedy formát bytů, časování a obsah příkazů REQ a ATQ. Dále popisuje, jak detekovat a komunikovat pouze s jednou kartou z mnoha v dosahu zařízení, antikolizní metody. Poslední část (ISO 14443-4) popisuje protokoly a příkazy používané na vyšších vrstvách komunikace po inicializaci[11].

Některé zprávy obsahují takzvaný cyklický redundantní součet (dále jen CRC, z anglického Cyclic redundancy check). Je to metoda používaná k detekci chyb, nemůže být ovšem použita k jejich korekci. Součet generuje cyklický posuvný registr se zpětnou vazbou. Parametry registru (a výpočtu) se mění v závislosti na typu této metody. Standard ISO 14443 používá CRC typu A. Registr má 16 bitů s počáteční hodnotou $0x6363$ a generačním polynomem $C(x) = x^{16} + x^{12} + x^5 + 1$. Výsledek metody se přidá na konec zprávy. Po přijetí zprávy je kontrolní součet znovu, nezávisle spočítán stejnou metodou. Pokud se vypočítaný součet liší od přijatého, nastala při přenosu chyba[24][11].

3.3 Mifare®

Mifare® je obchodní značka rakouské společnosti NXP Semiconductors, dříve známé pod jménem Philips Semiconductors. Tato značka zahrnuje množství proprietárních, bezdrátových řešení splňující mezinárodní standard ISO/IEC 14443. Produktová rodina zahrnuje čtyři typy karet. Zpětná kompatibilita zajišťuje bezproblémové přecházení na lépe zabezpečené produkty s více funkcemi. S více než 10 miliardami prodaných karet ovládá Mifare zhruba 80% světového trhu s bezdrátovými chytrými kartami. [20][5].

Produkt	Rychlost přenosu	Velikost paměti	Šifrování	Životnost
Classic®	106 kbps	1-4 KB	CRYPTO1	10 let
Plus®	106-848 kbps	2-4 KB	CRYPTO1, 128 bit AES	10 let
DESFire®	106-848 kbps	256 B - 8 KB	128 bit AES, 168 bit DES	10 let
UltraLight®	106 kbps	40-144 B	112 bit DES	2 roky

Tabulka 3.1: Porovnání jednotlivých Mifare karet[23]

3.3.1 Varianty karet Mifare

Karty Mifare Classic® byly vyvinuty již v roce 1994 a brzy se staly úspěšným produktem. Původně byly navrženy jako karty se zabezpečenou pamětí. Byly proto používány v různých odvětvích, například v hromadné dopravě, ve školních kampusech nebo jako zaměstnanecké karty. Data a autentizace jsou šifrovány proprietární šifrou CRYPTO1. Zabezpečení ale není nejsilnější stránkou těchto karet. Pro porovnání i mnohem starší DES šifrování odolá o několik řádů déle útokům hrubou silou (brute force). Toho si je firma vědoma a nadále tyto karty nedoporučuje pro aplikace s důrazem na zabezpečení. Požadavky na bezpečnost vedly k vývoji dvou nových, lépe zabezpečených typů karet Mifare Plus a Mifare DESFire[17][21].

Chytré karty Mifare Plus® jsou nástupcem karet Classic. Zpětná kompatibilita je zajištěna podporou starší a méně bezpečné šifry CRYPTO1. Je tedy možné postupně modernizovat již zavedené systémy, i když tento způsob neodstraňuje všechna rizika karet Classic. Navíc je implementována mnohem bezpečnější 128bitová AES šifra. Karty Mifare DESFire® už šifrování CRYPTO1 nepodporují vůbec. Podporují ovšem šifrování DES a AES, komunikaci pomocí NFC a až 28 různých aplikací na jedné kartě[21]. V České Republice je používají například České dráhy ve svém produktu In Karta[29]. Naopak Ultralight® jsou velmi levné karty s krátkou životností a malou pamětí. Jsou vhodné pro jednorázové použití jako celodenní jízdenky nebo vstupenky na velké události[21]. Každá z těchto karet se dělí na další dva až čtyři podtypy, které se liší v různých parametrech nebo nabízených vlastnostech karty.

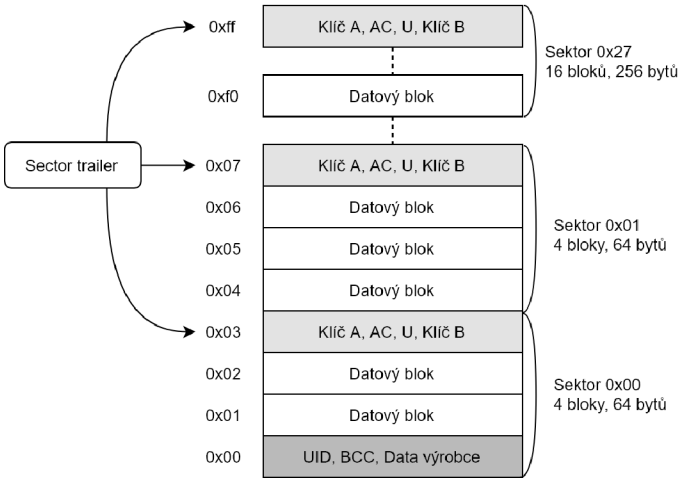
3.4 Karty Mifare Classic

Jak již bylo řečeno, Classic jsou karty se zabezpečenou pamětí. Data udržuje čip s pamětí typu EEPROM. To je zkratka pro Electrically Erasable Programmable Read-Only Memory, tedy elektronicky mazatelná a programovatelná paměť pouze pro čtení. Důležitou vlastností je, že EEPROM je nevolatilní paměť, svůj stav si udrží i bez zdroje napájení[24]. Nad pamětí karty Classic lze provádět základní operace jako čtení, zápis, přičtení a odečtení. Paměť karty je rozdělena do sektorů a sektory jsou dále rozděleny do bloků po 16 bytech. Na prvním bloku prvního sektoru je zapsáno UID karty, kontrolní součet a data výrobce. Poslední blok každého sektoru obsahuje dva 48bitové klíče a podmínky přístupu pro daný sektor. Klíče jsou sdíleny s legitimním čtecím zařízením. To se musí autentizovat alespoň jedním z nich. Podmínky přístupu říkají, se kterým klíčem lze provádět jaké operace[5][17].

3.4.1 Struktura paměti

Paměť karet je rozdělena do sektorů a ty jsou dále děleny do bloků po 16 bytech. Karta Mifare Classic 1k disponuje 16 sektory, a každý z nich má 4 datové bloky. Struktura karet Mifare Classic 4k je více heterogenní. Prvních 32 sektorů se skládá ze 4 datových bloků a

zbývajících 8 sektorů obsahuje bloků 16. První blok prvního sektoru obsahuje speciální data. Na prvních 4 bytech je zapsán unikátní identifikátor (UID) karty. Následuje jednobytová bitová kontrola počtu (dále jen BCC z anglického bit count check). Ta se vypočítá postupným provedením operace XOR nad všemi byty UID. Na zbývajících bytech jsou uložena data výrobce. Celý tento blok je pouze pro čtení.

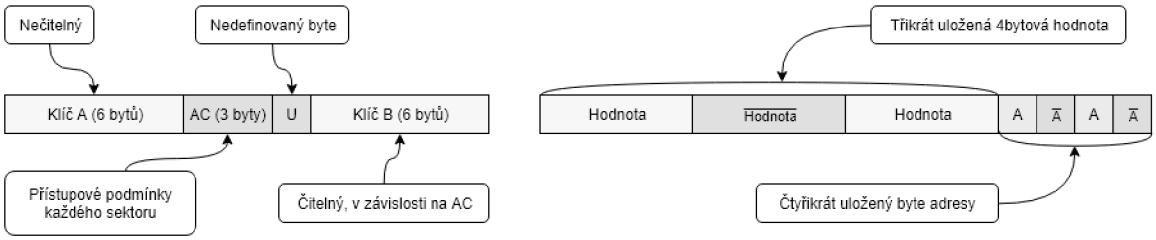


Obrázek 3.1: Struktura paměti karty Classic[3]

Před jakoukoliv operací nad pamětí karty se čtecí zařízení musí nejprve autentizovat proti sektoru, se kterým chce pracovat. Každý sektor uzavírá takzvaný sector trailer, speciální datový blok. Obsahuje tajné klíče A a B, které jsou použity při autentizaci. Operace, které je možné nad sektorem provádět, jsou v podmínkách přístupu AC (z anglického Access Condition). Poslední částí sector traileru je jeden datový byte U, který nemá definovaný účel. Může však být použit pro uložení dat. Sector trailer má zvláštní podmínky přístupu. Zatímco klíč A není čitelný nikdy, klíči B se čitelnost nastavit může. V takovém případě se jím nedá autentizovat a je považován za uživatelská data. Čitelností klíče je myšlen přístup čtecího zařízení k tomuto datovému prostoru s právy pro čtení, karta samotná je může číst bez problémů[3].

Nastavení klíčů a podmínek přístupu se provede jednoduchým zápisem dat do sector traileru. Nejmenší jednotka přístupu je však celý blok. Načítání nebo zápis tedy přečte, respektive přepíše blok celý. Změna jediného bytu vyžaduje načtení a přepsání celého bloku. Pro sector trailer je tato operace o něco složitější. Čtení bytů na pozicích, kde jsou uloženy klíče, vrací nuly. Tedy změna konfigurace beze změny klíčů vyžaduje znalost těchto klíčů. Takže například nelze zjistit neznámý klíč B změnou konfigurace podmínek přístupu AC a jeho následným přečtením, protože změna AC vyžaduje přepsání celého bloku i s klíčem[27].

Na datových blocích jsou uložena libovolná data, nebo jsou konfigurovány jako blok s hodnotou (value block). Při použití hodnotového bloku je 4bytová podepsaná hodnota uložena dohromady třikrát. Dvakrát normálně a jednou invertovaně, tedy se všemi bity negovanými. Tyto 4 byty mají uložen nejvýznamnější byte vpravo a nejméně významný vlevo (little-endian). Na posledních čtyřech bytech je uložena jednobytová adresa bloku, která může být použita jako ukazatel. Adresa je uložena čtyřikrát po sobě, z toho druhý a čtvrtý byte jsou opět negovány[3].



Obrázek 3.2: Struktura paměti sector traileru a bloku s hodnotou[3]

3.4.2 CRYPTO1

Po autentizaci je veškerá komunikace mezi čtecím zařízením a kartou šifrována. Pro šifrování se používá proprietární proudová šifra CRYPTO1 navržena přímo firmou NXP[17]. Proudové šifry jsou symetrické šifry kde je důvěrný text neznámé délky (anglicky plaintext) bit po bitu kombinován s proudem pseudonáhodných šifrovacích bitů (dále jen anglicky keystream). Kombinace se nejčastěji provádí pomocí funkce XOR a jejím výsledkem je zašifrovaný text (cipher text). Šifrovaný text je dešifrován stejnou funkcí a keystreamem. Keystream i důvěrný text musí být stejně dlouhé. Aby bylo možné tohoto dosáhnout s konečnou pamětí, je potřeba keystream generovat. To se děje na základě klíče (anglicky secret key) v generátoru[1]. Šifra CRYPTO1 jako generátor používá 48bitový posuvný registr s lineární zpětnou vazbou (dále jen LFSR, z anglického Linear Feedback Shift-register) s generačním polynomem 3.1. Každou periodu hodinového signálu se z dvaceti určitých bitů pomocí filtrační funkce 3.3 vypočítá bit keystreamu a potom se všechny bity registru posunou doleva. Bit nejvíce vlevo je zahozen a nový, pravý bit je jako zpětná vazba vypočítán funkcí 3.2, kde x je aktuální stav registru. V průběhu inicializace se bere v úvahu také vstupní bit, který je kombinován funkcí XOR s 3.2. [5].

$$g(x) = x^{48} + x^{43} + x^{39} + x^{38} + x^{36} + x^{34} + x^{33} + x^{31} + x^{29} + x^{24} + x^{23} + x^{21} + x^{19} + x^{13} + x^{9} + x^{7} + x^{6} + x^{5} + 1 \quad (3.1)$$

$$L(x_0, x_1 \dots x_{47}) = x_0 \oplus x_5 \oplus x_9 \oplus x_{10} \oplus x_{12} \oplus x_{14} \oplus x_{15} \oplus x_{17} \oplus x_{19} \oplus x_{24} \oplus x_{25} \oplus x_{27} \oplus x_{29} \oplus x_{35} \oplus x_{39} \oplus x_{41} \oplus x_{42} \oplus x_{43} \quad (3.2)$$

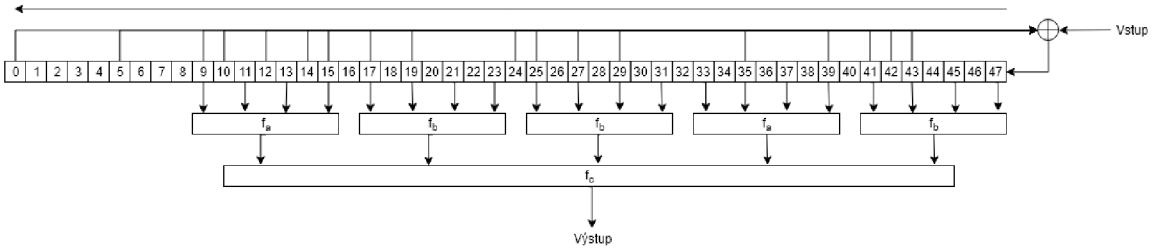
$$f(x_0, x_1 \dots x_{47}) = f_c(f_a(x_9, x_{11}, x_{13}, x_{15}), f_b(x_{17}, x_{19}, x_{21}, x_{23}), f_b(x_{25}, x_{27}, x_{29}, x_{31}), f_a(x_{33}, x_{35}, x_{37}, x_{39}), f_b(x_{41}, x_{43}, x_{45}, x_{47})) \quad (3.3)$$

$$f_c(y_0, y_1, y_2, y_3, y_4) = (y_0 \vee ((y_1 \vee y_4) \wedge (y_3 \oplus y_4))) \oplus ((y_0 \oplus (y_1 \wedge y_3)) \wedge ((y_2 \oplus y_3) \vee (y_1 \wedge y_4))) \quad (3.4)$$

$$f_a(y_0, y_1, y_2, y_3) = ((y_0 \vee y_1) \oplus (y_0 \wedge y_3)) \oplus (y_2 \wedge ((y_0 \oplus y_1) \vee y_3)) \quad (3.5)$$

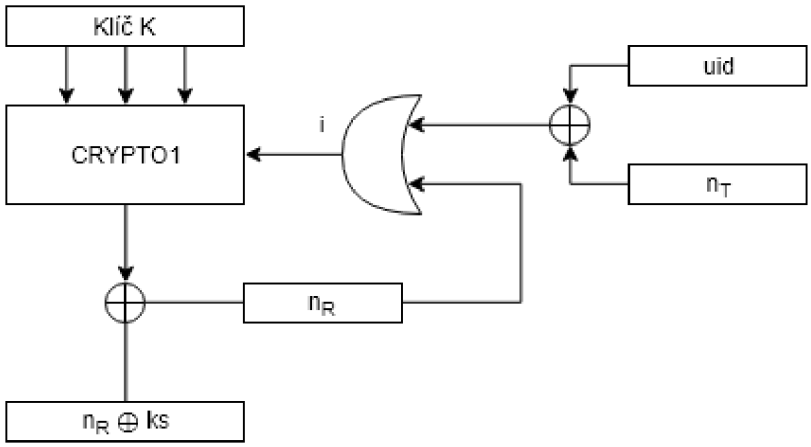
$$f_b(y_0, y_1, y_2, y_3) = ((y_0 \wedge y_1) \vee y_2) \oplus ((y_0 \oplus y_1) \wedge (y_2 \vee y_3)) \quad (3.6)$$

Inicializace šifry probíhá při autentizaci karty a čtecího zařízení. Poté, co karta odešle výzvu n_T (viz kapitola 3.4.3) je do registru šifry obou zařízení načten sdílený tajný klíč K.



Obrázek 3.3: Struktura šifry CRYPTO1[6]

Zde přichází na řadu vstupní bit, který se v tomto momentu vypočítá jako $i = n_T \oplus uid$. Všech 32 bitů tohoto vztahu se naplní do registru spolu s bity zpětné vazby LFSR. Následně se vstupní bity změní na bity výzvy čtecího zařízení n_R a jsou aplikovány stejným způsobem, tedy $g(x) \oplus i$. Protože šifrování komunikace začíná při odesílání n_R, dřívější bity n_R ovlivňují šifrování pozdějších bitů n_R. Na diagramu 3.4 je znázorněna inicializace šifry v obou zařízeních. Jediný rozdíl je v tom, že čtecí zařízení nejprve vygeneruje n_R a poté spočítá $\{n_R\} = n_R \oplus ks$, zatímco karta přijme $\{n_R\}$ a teprve spočítá n_R. Od této chvíle je inicializace hotová a vstupní bit šifry není nadále potřeba[5].



Obrázek 3.4: Diagram inicializace šifry CRYPTO1, kde ks je šifrovací proud (keystream)[5]

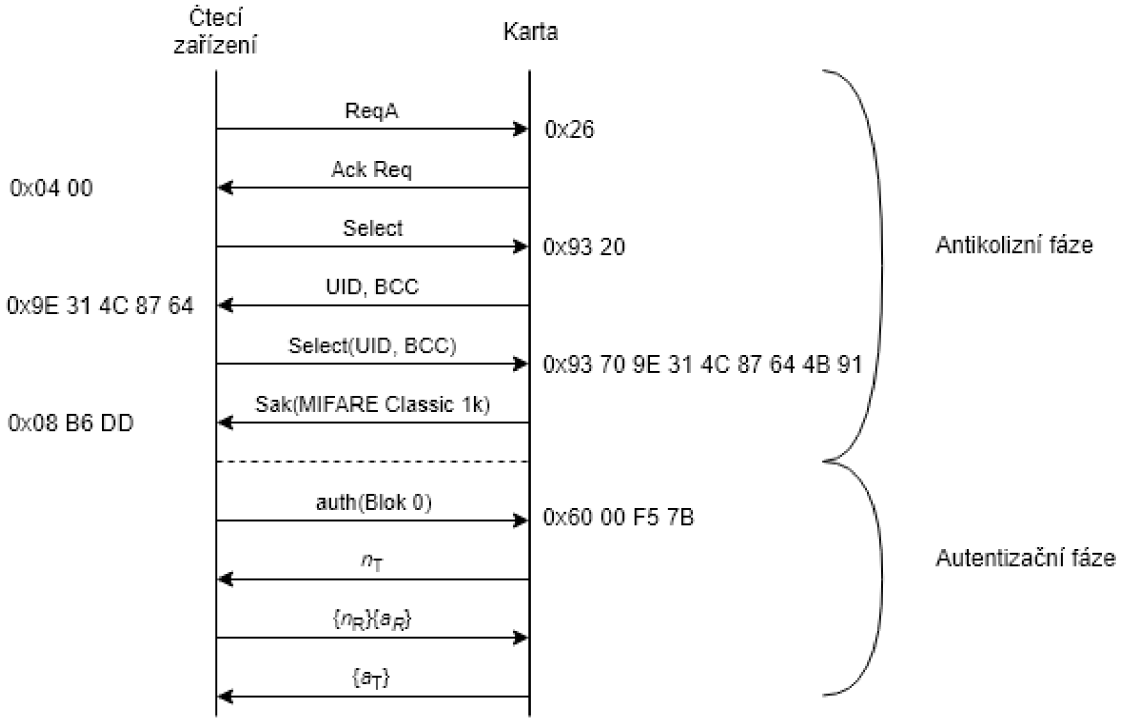
3.4.3 Autentizace

Komunikace karet Mifare Classic implementuje standard ISO 14443. Ve čtvrté části se implementace od standardu liší a Mifare zde používá svůj vlastní, neveřejný protokol. Po vstupu do blízkosti čtecího zařízení a nabití karty nastává antikolizní fáze (viz 3.5). Komunikaci zahajuje čtecí zařízení příkazem Select. Odešle *0x93 20*, na což karta odpoví svým UID *u*. Čtecí zařízení pošle *0x93 70* následováno UID vybrané karty a dvěma byty CRC. Antikolizní fáze je ukončena odpovědí karty SAK (z anglického Select acknowledge). Nyní je karta v aktivním stavu a připravena přijímat příkazy vyšší vrstvy[3].

Čtecí zařízení požádá o autentizaci pro specifický blok odesláním požadavku *0x 60 00 F5 7B*. První byte *0x60* znamená, že se má autentizace provést s klíčem A. Pro autentizaci s klíčem B musí být byte *0x61*. Druhým bytem se vybere blok, pro který se chce čtecí zařízení autentizovat. Blok 0 je v sektoru 0, autentizace tedy proběhne pro celý sektor 0. Pokud by

byl požadován například blok 5, autentizace proběhne pro sektor 1. Zbývající dva byty jsou opět CRC[3].

Na tento požadavek karta pošle výzvu (anglicky challenge) n_T[1] ve formě takzvané nonce[6]. Nonce je anglický výraz pro slovo na jedno použití. V kryptografii označuje náhodné číslo použité při komunikaci pouze jednou[26]. Od této chvíle je komunikace šifrována, tedy XORována s pseudonáhodným proudem bitů (anglicky keystream). Čtecí zařízení odpoví se svou vlastní výzvou n_R a odpovědí na výzvu karty $a_R = suc^{64}(n_T)$. Autentizace je dokončena odpovědí karty $a_T = suc^{96}(n_R)$. Po této odpovědi jsou karta i čtecí zařízení vůči sobě autentizovány. Autentizace je platná pouze pro sektor, o který bylo požádáno[6].



Obrázek 3.5: Antikolizní a autentizační fáze komunikace. Šifrované zprávy jsou ve složených závorkách.[5]

3.4.4 Komunikační protokol

Pro manipulaci s daty mají karty Classic malou sadu příkazů. Aby mohly být provedeny nad datovým blokem, musí být čtecí zařízení autentizováno pro sektor, který tento blok obsahuje. Před každým použitím jakéhokoliv příkazu se kontrolují přístupové podmínky. Ne všechny příkazy mohou být povoleny. Například blok může být nastaven pouze pro čtení nebo jiný blok s hodnotou může být pouze inkrementován[3].

Po autentizaci následuje šifrovaná komunikace s kartou. Příkazy Read a Write čtou či zapisují jeden blok. Ten může být jak datový, tak hodnotový. Příkaz Write může být použit k formátování datového bloku na blok s hodnotou nebo na zapsání libovolných dat do bloku. Další příkazy jsou povoleny pouze na hodnotových blocích. Jsou to Decrement, Increment, Restore a Transfer. Příkazy Increment a Decrement inkrementují nebo dekrementují hodnotu hodnotového bloku a výsledek vloží do paměťového registru. Příkaz Restore načte do

[1] Notace je zachována stejná jako v [6], tedy T jako tag, R jako reader, n jako nonce a a jako answer

registru hodnotu nezměněnou a příkaz Transfer nahraje hodnotu z registru zpět do stejného nebo jiného bloku[3].

Kapitola 4

Známé zranitelnosti

Zabezpečení není silnou stránkou karet Mifare Classic. Tato podkapitola obsahuje popis jednotlivých nedostatků. Nedostatečná délka šifrovacích klíčů je první z nich. Následuje předvídatelnost výzev posílaných při autentizaci. Popsáno je také nedostatečné řešení výpočtu paritních bitů. Další zranitelnosti jsou více kryptografického rázu, s jejichž pomocí lze získat aktuální stav šifry a vrátit ho do stavu, kdy obsahoval tajný klíč sektoru. Nakonec je popsána zranitelnost při autentizaci více sektorů, která byla pojmenována jako vnořená autentizace.

4.1 Krátké šifrovací klíče

Pro šifrování karet se používají 48bitové klíče. Tato délka klíčů je ale příliš malá na to, aby zabránila úspěšnému brute force útoku v dosažitelném čase. Proto bylo zavedeno zpoždění v komunikaci a v autentizaci. Každý pokus by trval 6 milisekund. Díky této kompenzaci by online brute force útok na jeden sektor, prohledávající všech 2^{48} možných klíčů, trval více než 44 tisíc let. Odhalení algoritmu šifry CRYPTO1 umožnilo provést offline brute force útok. V takovém případě útočník nemusí komunikovat se zařízením pod útokem. Stačí mu pouze záznam komunikace, tím se eliminuje zavedené zpoždění. V prosinci 2007 Nohl a Plötz uvedli, že zařízení za 100$ dokáže najít klíč přibližně za týden. Tato doba lze dál zkrátit přidáním paměti[19].

4.2 Předvídatelné výzvy

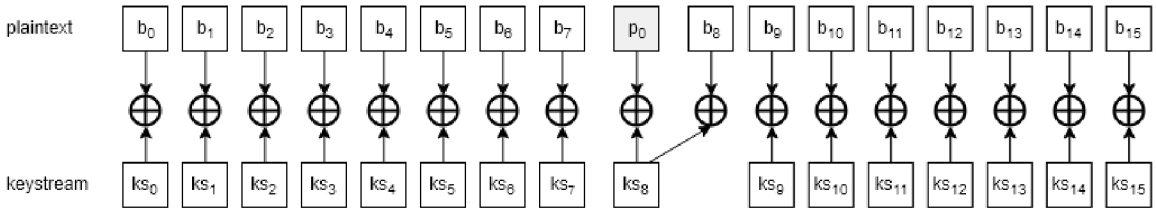
Aby mohly kryptografické protokoly poskytovat správné zabezpečení, je pro ně zásadní dostatečný generátor pseudonáhodných čísel. Čísla pro Mifare Classic generuje 16bitový LFSR. Výzvy n_T (nonce, viz. 3.4.4) použité při autentizaci jsou ale 32bitové. To znamená, že první polovina n_T určuje její zbytek. Sekvence všech výzev se opakuje každých $2^{16} - 1$ cyklů[19]. Cyklus generátoru karty je rozdílný od čtecího zařízení. Zatímco karta mění stav každou periodu hodinového signálu, čtecí zařízení aktualizuje stav jen při volání generátoru[5]. Generátor v kartě se resetuje do původního stavu při každém nastartování karty. Výzva poslaná kartou je tedy podmíněna pouze časem mezi zapnutím elektromagnetického pole k nabití karty a momentem odeslání žádosti o autentizaci. Autentizace je tedy zbavena jakékoliv náhodnosti.

Útočník s fyzickým přístupem ke kartě ji může ”přinutit” k odeslání vždy stejné výzvy. K tomu je potřeba po každém pokusu vypnout elektromagnetické pole (zhruba na

30μs), aby se vybily všechny kondenzátory, znovu zapnout pole a počkat konstantní čas před odesláním požadavku o autentizaci. Druhý způsob spočívá v čekání mezi opětovným odesláním požadavku přesně stanovený čas t. Stav generátoru pseudonáhodných čísel se mění každých 9,44μs. Na vystřídání všech stavů tak stačí pouze $(2^{16} - 1) * 9,44\mu s = 618,650ms$. Velikost výzvy je dvojnásobná oproti LFSR, výsledný čas t je tedy poloviční $t = 618,650ms/2 = 309,325ms$ [6]. V novější verzi karet Classic EV1 je tento nedostatek odstraněn nahrazením generátoru pseudonáhodných čísel generátorem náhodných čísel[21].

4.3 Paritní bity

Další slabinou karet Mifare Classic je jejich řešení paritních bitů. Podle standardu ISO 14443 musí každý odeslaný byte následovat lichý paritní bit. Karty Classic sice s každým odeslaným bytem posílají paritní bit, ten je ale vypočítán už z důvěrného textu (plaintext) a ne z textu šifrovaného (ciphertext), který je skutečně posílán komunikační vrstvou. Kromě toho odeslané paritní bity jsou šifrovány stejným šifrovacím bitem (keystream), který je použit pro šifrování dalšího bitu plaintextu (viz 4.1)[19]. Kvůli tomu je s každým odeslaným bytem vyzrazena jednobitová informace o důvěrném textu[5].



Obrázek 4.1: Šifrování paritních bitů [19]

V souvislosti s paritními bity existuje ještě jedna slabina. V průběhu autentizační fáze karta vždy nejprve kontroluje právě paritní bity. Když karta přijme $\{n_R\}\{a_R\}$ (viz. 3.4.3) a nějaký z osmi paritních bitů je špatný, karta neodešle žádnou odpověď. V případě že jsou všechny paritní bity správné, ale a_R je špatně, karta odešle 4bitovou chybovou zprávu $0x5$ indikující selhání autentizace. Jak bylo řečeno v 3.4.3, komunikace je v tuto chvíli šifrována. I když se čtecí zařízení úspěšně neautentizovalo, dostane chybovou zprávu zašifrovanou. Nelze předpokládat, že si ji správně rozšifruje a dozví se, že došlo k selhání. Jelikož je kód této chyby známý, lze z šifrované zprávy získat 4 bity keystreamu. Přestože se takovýto únik nemusí zdát jako velký problém, je nepostradatelnou částí mnoha útoků, které cílí na klíče těchto karet.

Jestliže se čtecí zařízení nemá jak dozvědět o selhání při autentizaci, pak nezáleží na tom, jestli dostane zprávu šifrovanou nebo žádnou. Zmíněná slabina tedy lze odstranit vydáním nových karet, které při autentizaci neposílají žádné chybové kódy. Tato metoda je sice nákladná pro již zavedené systémy s kartami v oběhu, ale je pořád kompatibilní s Mifare Classic protokolem[19].

4.4 Navrácení stavu posuvného registru

Stav posuvného registru na začátku inicializace obsahuje tajný klíč sektoru. V průběhu autentizace a šifrování se stav tohoto registru mění. Pokud by se útočníkovi podařil nějakým způsobem získat stav registru v určitém čase a záznam komunikace, lze jednotlivé změny

stavu obrátit a deterministicky spočítat jakýkoliv předchozí stav registru včetně tajného klíče.[5]

4.5 Získání stavu šifry

Pro získání aktuálního stavu registru v CRYPTO1 existují dvě metody. První metoda využívá výzev manipulujících se stavem šifry při autentizaci. Nejprve je potřeba vygenerovat tabulku dvojic (*lfsr*, *ks*), kde *lfsr* jsou stavy registru určitého formátu (*0x000WWWWWWWW*) a *ks* je prvních 64 bitů keystreamu. Předpočítaná tabulka je znovupoužitelná na jakýkoliv klíč/čtecí zařízení. Výsledek obsahuje 2^{36} dvojic zabírající přibližně 1TB. Následuje útok na samotné čtecí zařízení. Pro každý stav 0xXXX je zahájena autentizace se zafixovaným UID karty a výzvou karty ve tvaru $n_T = 0x0000XXX0$. Po odpovědi čtecího zařízení ve tvaru $n_R \oplus ks_1, suc(n_T) \oplus ks_2$ se v komunikaci nepokračuje. Na to většina čtecích zařízení odešle $halt \oplus ks_3$(tento útok lze provést i nad čtecími zařízeními, které příkaz halt neodešlou). Protože $suc(n_T)$ a formát příkazu halt jsou známé, je možné získat i ks_2 a ks_3. Existuje právě jedna hodnota 0xXXX, která změní stav šifry do jednoho ze stavů v předpočítané tabulce. Tento stav odhalíme vyhledáním ks_2, ks_3 v tabulce. V konfiguraci této metody lze vyměnit paměť za čas, tedy místo 12 bitů měnit bitů 13, snížit velikost tabulky na polovinu, ale zdvojnásobit počet potřebných autentizací.[5]

Druhá metoda využívá chyby v návrhu filtrační funkce 3.3. Konkrétně toho, že její vstupy jsou pouze na lichých pozicích registru. Vstupním požadavkem je část keystreamu. Bity LFSR použité při výpočtu sudých a lichých bitů keystreamu lze generovat odděleně. Rozdělením také zpětné vazby na dvě části lze zkombinovat tyto sudé a liché části, čímž vzniknou právě ty stavy LFSR, které generují daný keystream.[5]

4.6 Vnořená autentizace

Za předpokladu, že útočník zná některý z tajných klíčů a je jím ověřen vůči příslušnému bloku karty, autentizace vůči dalším blokům probíhá trochu jinak než při první autentizaci. Další autentizaci zahájí útočník odesláním požadavku o autentizaci. Tento požadavek je šifrovanán původním klíčem předchozího bloku, narozdíl od první autentizace, kdy požadavek nebyl šifrován vůbec. Po zpracování požadavku je vnitřní stav šifry CRYPTO1 nastaven na klíč nového sektoru a autentizační protokol popsaný v 3.4.3 začíná od začátku (ovšem bez antikolizní fáze). Změna nastává při odesílání výzvy n_T. Ta je totiž nyní odeslána zašifrovaná s tím, že pro šifrování byl použit již nový klíč. Se znalostí šifrování paritních bitů z 4.3 je možné zredukovat 2^{16} výzev na pouhých 64. Kvůli implementaci slabého pseudonáhodného generátoru čísel lze téměř s jistotou určit, která výzva je ta správná. Odeslaná výzva závisí na takzvané vzdálenosti od předchozího pokusu o autentizaci. Tato vzdálenost byla popsána v [5] jako počet posunutí registru generátoru čísel. Pomocí odhadnuté výzvy lze odhalit 32 bitů šifrovacích bitů (keystream)[5]. Tuto slabinu využívají například útoky popsané v [19] a v [6] kde je útok nazván vnořeným útokem (nested attack).

Kapitola 5

Chameleon Mini

V této kapitole bude představeno zařízení Chameleon Mini, jeho původ, využití, hardware a výběr podporovaných příkazů.

Chameleon Mini je zařízení velikosti běžné kreditní karty německé firmy Kasper & Oswald. Zařízení je navrženo jako univerzální nástroj pro praktickou bezpečnostní analýzu NFC[1]a RFID systémů, penetrační testování nebo různé jiné aplikace koncových uživatelů. Tato volně programovatelná platforma umožňuje emulovat existující komerční chytré karty, a to včetně kryptografických funkcí a unikátních identifikátorů (UID). Dále může být využito v různých scénářích útoků, na odposlech NFC a RFID komunikace nebo funkční testování RFID vybavení. O bezstarostné nahrávání firmware se stará USB zavaděč. Pomocí praktické a lehce zapamatovatelné příkazové sady lze měnit nastavení a chování zařízení. Zároveň dokáže do vnitřní paměti uložit a virtualizovat až osm bezkontaktních karet.[15]

Projekt Chameleon Mini byl založen na Porúrské univerzitě v Bochumi a je financován pomocí crowdfundingu na platformě KickStarter. V současné době se veškerý oficiální firmware, schémata tištěných spojů nebo podpůrné nástroje nachází v open-source repozitáři služby GitHub[2].[14]

5.1 Hardware

Zařízení je prodáváno jako deska plošných spojů. Nejnápadnější částí je pak anténa schopná generovat RFID pole o frekvenci 13,56MHz. Chameleon Mini může emulovat karty standardů ISO 14443, ISO 15693, NFC tagy a díky RFID poli dokáže vystupovat také jako aktivní čtecí zařízení.[15] Dosah antény je závislý na módu zařízení. Při emulaci karet má na dosah vliv také typ čtecího zařízení. V tomto případě se tak dosah pohybuje od 0 do 8,5cm. Dosah v módu čtecího zařízení závisí na tvaru a typu čteného tagu. Standardní bezkontaktní karty lze číst na vzdálenost 1,5cm. Malé tagy (například Mifare UltraLight) je potřeba položit přímo na anténu.[13] Pro praktické a bezdrátové použití lze pomocí USB nabít lithium-iontovou baterii přímo na desce. Po plném nabití je možné zařízení používat zhruba hodinu. Interakci v bezdrátovém módu umožňují dvě integrovaná, nastavitelná tlačítka a LED diody. Baterii je možné odpojit pomocí vypínače a šetřit tak její energii. Pomocnými piny lze připojit jakékoliv externí vybavení a zakomponovat tak možnosti tohoto nástroje například do zámku dveří nebo jiného zařízení IoT (z anglického Internet of Things, internet věcí).[15]

[1] z anglického Near-field communication

[2] https://github.com/emsec/ChameleonMini

Vnitřní logiku po hardware stránce zajišťuje mikrokontroler Atmel ATXMega128A4U. Ten poskytuje funkce pro radio-frekvenční kódování a dekódování, rozhraní USB a je v něm nahraný stavový automat, kterým je řízen chod zařízení. Mikrokontroler obsahuje podporu AES a DES na hardware úrovni. Výpočty těchto kryptografických algoritmů jsou tak podle výrobce až třikrát rychlejší než na originálních kartách (Mifare DESfire). Jako operační paměť slouží 128kB Flash paměť, 2kB EEPROM paměť a 8kB SRAM paměť. Do volatilní paměti SRAM je ukládána odposlouchaná komunikace. Tu je možné zapsat do nevolatilní 128kb FRAM paměti, která slouží také jako uložiště pro virtuální karty.[15]

5.2 Podporované příkazy

Ke komunikaci se zařízením Chameleon Mini lze použít příkazovou řádku, emulátor terminálu nebo vlastní skripty a aplikace. Struktura firmware umožňuje jednoduchou rozšiřitelnost o nové, nepodporované standardy a karty. Tlačítkům a LED diodám je možné přiřadit různé funkce. Přenos dat mezi zařízením a počítačem přes USB zajišťuje protokol X-MODEM. [15]

Chameleon Mini připojený pomocí USB do počítače je zobrazeno jako sériové zařízení. Zařízení se nastavuje a ovládá příkazy posílanými přes rozhraní příkazové řádky. K tomuto rozhraní jsem přistupoval pomocí programu Putty. Příkazy je možné použít se čtyřmi různými syntaxemi.

- <Příkaz>=<Hodnota> Nastaví parametr zařízení
- <Příkaz>=? Vypíše seznam všech možných hodnot parametru
- <Příkaz>? Vrátí aktuální hodnotu parametru
- <Příkaz> Provede funkci a vypíše možnou odpověď

Chameleon poskytuje osm slotů, které je možno nakonfigurovat jako aktivní čtecí zařízení, pasivní zařízení pro odposlech komunikace (sniffing) nebo pro uložení různých virtualizovaných karet. V každém slotu je uložena jeho konfigurace a případně obsah karty. Číslo právě aktivního slotu zjistí příkaz `Setting?`. Jiný slot aktivujeme pomocí `Setting=X`, kde X je číslo od 1 do 8. Příkaz `Config?` vrátí konfiguraci aktivního slotu. Všechny možné konfigurace vyjmenuje `Config=?` a opět změnu konfigurace například na emulaci karty Mifare Classic provede příkaz `Config=MF_CLASSIC_1K`(viz. část 5.1).

Každý odeslaný příkaz následuje odpověď zařízení s číslem a zprávou stavu. První cifra třímístného čísla stavu ukazuje vážnost odpovědi. Zprávy s číslem začínající cifrou ”1” jsou spíše informačního charakteru, zatímco cifra ”2” značí chybu.[14]

Slotu s konfigurací virtualizace karty můžeme nastavit libovolné UID příkazem `UID=<UID>`. Stejně jako v předchozím případě neznámé UID zjistíme voláním `UID?`. V některých případech je potřeba nastavit náhodné UID virtualizované karty, například jako opatření proti ukládání škodlivých UID na černou listinu. Toho lze docílit nastavením jednoho z tlačítek `Rbutton=UID_Random` jeho zmáčknutím. V rozhraní příkazové řádky tato funkce (zatím) zveřejněná není, ale pomocí tlačítka lze UID měnit přímo v terénu bez připojení k počítači.

Při komunikaci může dojít ke změně paměti virtualizované karty, například snížení kreditu na kartě. Obnovením paměti karty by se kredit dostal do původního stavu a šel by znovu použít. Voláním příkazu `Store` se aktuální stav karty uloží z FRAM do paměti nevolatilní paměti Flash. Po domnělé změně stavu karty `Recall` načte kartu zpět do FRAM,

připravenou k použití v původním stavu. Tato funkcionalita lze provázat s dvěma tlačítky na zařízení příkazy `Rbutton=Store_Mem`, respektive `Rbutton=Recall_Mem`.

```
Setting?
101:OK WITH TEXT
2
Setting=4
100:OK
Config?
101:OK WITH TEXT
NONE
Config=?
101:OK WITH TEXT
NONE,MF_ULTRALIGHT,MF_ULTRALIGHT_EV1_80B,MF_ULTRALIGHT_EV1_164B,
MF_CLASSIC_1K, MF_CLASSIC_1K_7B,MF_CLASSIC_4K,MF_CLASSIC_4K_7B,
ISO14443A_SNIFF, ISO14443A_READER
Config=MF_CLASSIC_1K
100:OK
```

Výpis 5.1: Záznam komunikace se zařízením Chameleon Mini

Voláním příkazu `Download` je možné stáhnout obraz karty ze zařízení. Samotný přenos začne po navázání spojení protokolem X-Modem. Připojení tímto protokolem podporuje například program Tera Term. Mimo to se obraz karty dá stáhnout pomocí programu Chameleon Mini GUI[3], který poskytuje přehledné grafické rozhraní pro manipulaci se zařízením. Stažený obraz karty do aktivního slotu zařízení načte příkaz `Upload`, který opět počká na připojení X-Modem. Příkazem `Clear` se smaže celý obsah karty aktivního slotu.

Chameleon Mini v módu čtecího zařízení nabízí trochu jinou funkcionalitu, a tedy i jiné příkazy než v módu virtualizace karet. Mód čtecího zařízení aktivuje příkaz `Config=ISO14443A_READER`. Tento mód je aktivován nad aktivním slotem. Jakákoli data v tomto slotu jsou smazána. Tato konfigurace slotu neovlivňuje jiné sloty. Je tedy možné mít například první slot čtecí a ostatní emulující různé karty.

Základní příkaz této konfigurace je `Send` případně `Send_Raw`, které slouží k odesílání dat kartě. Příkaz `Send`, pokud je parametr delší než jeden byte, přidá paritní bity a odešle zprávu. Naopak příkaz `Send_Raw` předpokládá, že paritní bity jsou v parametru zprávy už obsaženy a odešle ji rovnou. Oba příkazy mohou mít jeden nebo dva parametry. Parametry jsou odděleny mezerou a nejsou uzavřeny v žádných závorkách. Jedním parametrem je předána zpráva k odeslání ve formě hexadecimálního čísla. V případě dvou parametrů je zpráva až druhý parametr. Prvním je předán počet bitů k odeslání opět ve tvaru hexadecimálního čísla. Po odeslání zprávy zařízení chvíli počká na odpověď a pokud ji přijme, zbaví ji paritních bitů a na příkazovou řádku vytiskne přijatou zprávu, její délku v bitech a buď *PARITY OK*, nebo *PARITY ERR* v závislosti na výsledku kontroly parity. Pokud žádnou odpověď neobdrží, vrátí *NO DATA*.

Jelikož čtecí zařízení je aktivní prvek, poskytuje Chameleon Mini možnost ovládat jeho elektromagnetické pole. Za pomoci příkazu `Field=1` lze toto pole zapnout. Naopak příkaz `Field=0` pole vypne. Aktuální stav zjistí `Field?`.

Jak již bylo zmíněno, pro interakci s uživatelem jsou na zařízení dvě konfigurovatelná tlačítka, levé a pravé. Kromě toho Chameleon rozlišuje mezi dlouhým a krátkým stiskem

[3] https://github.com/iceman1001/ChameleonMini-rebootedGUI

tlačítek, kdy stisk tlačítka delší než $1.28ms$ značí dlouhý stisk. Dohromady jsou tedy čtyři konfigurační příkazy těchto tlačítek, a to:

- Lbutton
- Rbutton
- Lbutton_Long
- Rbutton_Long

Každý z těchto příkazů podporuje nastavení ("="), získání aktuálního stavu ("?") a nápovědu ("=?").

Stav zařízení v terénu lze zjišťovat dvěma LED diodami, které se nastavují příkazy `Ledred` a `Ledgreen`. Diody mohou indikovat například změnu a stav paměti, komunikaci s terminálem nebo přítomnost elektromagnetického pole čtecího zařízení. Je jen na uživateli, jak diody nastaví.[14]

Kapitola 6

Testovací prostředí a podpůrné nástroje

Tato kapitola se věnuje testovacímu prostředí, ve kterém budou demonstrovany vybrané zranitelnosti. Některé demonstrace vyžadují implementační část. Pro usnadnění práce byla implementována knihovna tříd. Dokumentace její implementace je také součástí této kapitoly.

6.1 Testovací prostředí

Testování vybraných zranitelností probíhalo v reálném prostředí na čtecím zařízení HID RP10[1]. Po přiložení osobní karty studenta, či učitele je možné otevřít dveře a vstoupit. Karty používané studenty jsou mezinárodní studentské ISIC karty, případně VUT karty. Tyto karty jsou typu Mifare Classic, a jsou tedy vhodné k těmto demonstracím. Testování implementovaných demonstrací probíhalo na operačním systému Windows 10 Pro. Při testování byla použita zařízení Chameleon Mini, popsaná v kapitole 5.

6.2 Implementace modulů

První a poslední demonstrace vyžadují programovou implementaci. Nejdříve byly implementovány takzvané moduly, tedy třídy abstrahující komunikaci se zařízením Chameleon Mini nebo poskytující pomocné funkce. Všechny moduly jsou zapouzdřeny v knihovně tříd `MifareModules`. Některé z modulů mají vlastnost `Verbose` typu `Boolean`. Jejím nastavením na `true` začnou moduly zapisovat aktuální stav na standardní výstup k ulehčení ladění. Celá programová část této práce byla realizována v jazyce C# .Net Framework verze 4.6.

SerialModule

Třída `SerialModule` pozměňuje funkcionalitu `System.IO.Ports.SerialPort` (dále jen `SerialPort`) tak, aby vyhovoval požadavkům modulů vyšší vrstvy. V parametru konstruktoru je předávána třída `SerialModuleConfig`, která obsahuje veškeré možné nastavení komunikace přes sériový port. Třída je předpřipravená tak, že beze změn zajišťuje komunikaci se zařízením Chameleon Mini na portu "COM3". Konstruktor vytvoří komponentu `SerialPort` se zadanou konfigurací a rovnou jej otevře pro komunikaci.

[1] https://www.sourcesecurity.com/hid-rp10-access-control-reader-technical-details.html

```
public SerialModule(SerialModuleConfig config)
public void Open()
public void Close()
public void WriteLine(string message)
public string ReadLine()
public List<string> ReadLines(int numberOfLines)
public List<string> WriteAndGetResult(string message, int numberOfLines = 1)
public void WriteAndTrashResult(string message, int numberOfLines = 1)
```

Výpis 6.1: Metody třídy *SerialModule*

Práce metod `Open()`, `Close()`, `WriteLine(string)` a `ReadLine()` je vcelku jednoznačná. První dvě se starají o otevírání, respektive zavírání portu pro komunikaci. Pomocí třetí metody lze zapsat řetězec na otevřený port a poslední metoda řetězec z portu přečte. Tyto dvě metody mohou zapisovat řetězce komunikace na standardní výstup. Metoda `ReadLines(int)` přečte z portu několik řádků, v závislosti na jejím parametru a vrátí je jako seznam řetězců. Poslední dvě metody zjednodušují zápis a následné zpracování odpovědi. Metoda `WriteAndGetResult(string, int)` odešle zprávu a vrátí řádky odpovědi opět jako seznam řetězců. Není možné předpovědět, kolik řádků bude odpověď mít, proto je nutné zadat počet řádků k přečtení. Při zadání většího čísla než je počet řádků, metoda vrátí všechny řádky, které přečetla, a po zachycení vyjímky `TimeoutException` skončí. Zadáním menšího čísla metoda přečte a vrátí daný počet řádků. Problém nastane při dalším čtení, kdy nepřečtené řádky zůstanou ve vyrovnávací paměti portu a budou přečteny místo nových. Poslední metoda `WriteAndTrashResult(string, int)` funguje stejně jako předchozí metoda s tím rozdílem, že přečte odpověď, ale už ji nevrátí.

ChameleonModule

Tento modul abstrahuje komunikaci se zařízením Chameleon Mini tak, aby se uživatel nemusel starat o obsluhu sériového portu. Konstruktory jsou dva. Prvním je v parametru předána pouze instance třídy `SerialModule`. Tímto způsobem je předána informace, na kterém portu zařízení komunikuje. Druhým konstruktorem lze navíc specifikovat roli zařízení (při relay útoku) kvůli zápisu informací do konzole.

```
public ChameleonModule(SerialModule serial)
public ChameleonModule(SerialModule serial, string role)
public void GetCommands()
public void Send(string message)
public List<string> ReadToEnd()
public string SendWithAnswer(string message)
public string GetUid()
public void TurnElectromagnetic(Field value)
private string CalculateMessageLength(string message)
```

Výpis 6.2: Metody třídy *ChameleonModule*

Metoda `GetCommand()` je spíše informativního charakteru a vrátí všechny příkazy, které Chameleon podporuje. Pomocí metody `Send(string)` se odešle zpráva ve formátu hexadecimálního čísla v řetězci. Metoda sama vypočítá délku zprávy a připojí ji k příkazu `Send` pro Chameleon Mini. Metoda `ReadToEnd()` přečte všechny řádky odpovědi po příkazu `Send`. V této části už je známý počet řádků, jelikož všechny odpovědi končí buď řádkem ”PARITY

OK", "PARITY ERR", nebo "NO DATA". Funkcionalitu obou předchozích metod kombinuje `SendWithAnswer(string)`. Tato metoda odešle zadanou zprávu a vrátí relevantní část odpovědi. Pokud se v blízkosti zařízení nachází kompatibilní karta, metodou `GetUid()` získáme její UID. Elektromagnetické pole zařízení ovládá poslední veřejná metoda `TurnElectromagnetic(Field)`. Ta má jako parametr výčtový typ `Field` s hodnotami "Off" a "On". Poslední významná metoda je privátní `CalculateMessageLength(string)`, jenž vypočítá již zmíněnou délku zprávy k odeslání.

CardModule

Poslední z komunikačních modulů je zaměřen na komunikaci přímo s kartou a obsahuje metody usnadňující práci s protokolem ISO 14443. Informaci o tom, u kterého zařízení se karta nachází, je jedinému konstruktoru předána parametrem typu `ChameleonModule`. Tato třída dále závisí na `CRCModule`, ten se ale parametrem nepředává. Jeho instance je vytvořena v konstruktoru. Třída však pokrývá pouze antikolizní fázi a začátek autentizační fáze. Pro další by byly potřebné knihovny s šifrou CRYPTO1.

```
public CardModule(ChameleonModule cm)
public string ReqA()
public string Select()
public string SelectUid(string uid)
public string AuthenticateForBlock(string blockNumber)
private string Get(ISOCode code)
```

Výpis 6.3: Metody třídy *CardModule*

Antikolizní fáze protokolu ISO 14443 je inicializována metodou `ReqA()`. Metoda `Select()` odešle příkaz *Select* podle protokolu a pomocí metody `SelectUid(string)` je vybrána karta na základě jejího UID. Po této sekvenci je antikolizní fáze. Metoda `AuthenticateForBlock(string)` začne s autentizační fází vůči paměťovému bloku na kartě. Vybraný blok je zadán ve formě řetězce hexadecimálního čísla. Všechny metody ke zprávám připojují kódy protokolu ISO 14443 nebo případně kontrolní součty CRC. Každá z těchto metod vrací odpovědi karty. Nedílnou součástí tohoto modulu je výčtový typ `ISOCode` s hodnotami kódů standardu. Čísla kódů z výčtového typu získává privátní metoda `Get(ISOCode)`.

CRCModule

Třída tohoto modulu zapouzdřuje a zjednodušuje použití externí knihovny `Nito.KitchenSink.CRC`[2], která slouží pro výpočet kontrolních součtů CRC. Konstruktor vytvoří instanci třídy `CRC16` se správným nastavením pro použití v předchozím modulu. Jediná metoda třídy `Hash(string)` vypočítá a vrátí kontrolní součet zadané hodnoty.

[2] https://www.nuget.org/packages/Nito.KitchenSink.CRC/

Kapitola 7

Demonstrace vybraných zranitelností

Tato kapitola se věnuje návrhu, implementaci, provedení a vyhodnocení tří vybraných zranitelností karet Mifare Classic. Nejprve se zaměřím na analýzu předvídatelných výzev karty Mifare Classic popsaných v části 4.2. Následují možnosti a úskalí emulace karet pomocí zařízení Chameleon Mini. Nakonec nastíním princip a provedení relay útoku.

7.1 Časová krypto-analýza

Jako první demonstraci jsem vybral zranitelnost nedostatečné náhodnosti generování výzev karty ve fázi autentizace, popsané v části 4.2. Tato analýza se týká pouze karet, protože generátor pseudonáhodný čísel ve čtecích zařízeních funguje jiným způsobem. Karty Mifare Classic generují 32bitové výzvy 16bitovým posuvným registrem s lineární zpětnou vazbou, jehož stav se mění každou periodu hodinového signálu karty a při startu karty se jeho stav resetuje vždy do stejného stavu. Perioda hodinového signálu je 9,44μs, zatímco perioda generátoru pseudonáhodných čísel je 65535. Generátor tedy vyčerpá všechny stavy během přibližně 0,618s. Odesláním požadavku o autentizaci vždy ve správný čas docílíme téměř konstantní výzvy karty.

S těmito znalostmi můžeme navrhnout dva způsoby, jak si kartou nechat vygenerovat vždy stejnou výzvu. První možností je mezi každým požadavkem o autentizaci čekat stanovený čas. Tímto počkáme až proběhne perioda generátoru pseudonáhodných čísel a jeho stav se bude rovnat stavu v době předchozího pokusu. Druhou možností je odesílat pokusy o autentizaci vždy ve stejný čas od startu karty. Startem karty se generátor pseudonáhodných čísel resetuje do konstantního stavu. Vyčkáním generátor vygeneruje vždy stejnou výzvu.

První možnost je z hlediska času náročnější. Aby bylo dosaženo požadovaného výsledku, musejí být žádosti odesílány pouze každých 0,618s. Naopak pokusy v druhé možnosti lze opakovat téměř ihned po získání výzvy. Za stejnou dobu tak lze získat větší množství výzev k porovnání. Využitím příkazu `Field` zařízení Chameleon Mini můžeme ovládat stav elektromagnetického pole zařízení a tím start karty. K analýze této zranitelnosti tak bude využita druhá možnost.

Cílem této demonstrace je ověřit možnost získání stejných výzev s vysokou pravděpodobností za pomoci zařízení Chameleon Mini. Výzva karty je použita jako jedna ze složek vstupního bitu použitého při inicializaci šifry CRYPTO1 (viz část 3.4.2). Ovládáním výzvy

karty útočník ovládne celý inicializační proces šifry karty, a tím i šifrovací bity použité k šifrování důvěrného textu.

7.1.1 Implementace

Tato demonstrace vyžaduje implementaci skriptu či programu, který automatizuje obsluhu elektromagnetického pole a průběh antikolizní a autentizační fáze. Chameleon v tomto případě napodobuje čtecí zařízení, bude tedy nastaven do konfigurace `ISO14443A_READER`. Při demonstraci bude Chameleon Mini opakovaně zapínat a vypínat své elektromagnetické pole a mezi tím vždy proběhne požadavek o autentizaci s kartou. Karta odešle výzvu, ta se uloží do seznamu a cyklus začíná od znova. Nakonec bude seznam výzev prohledán na duplikáty.

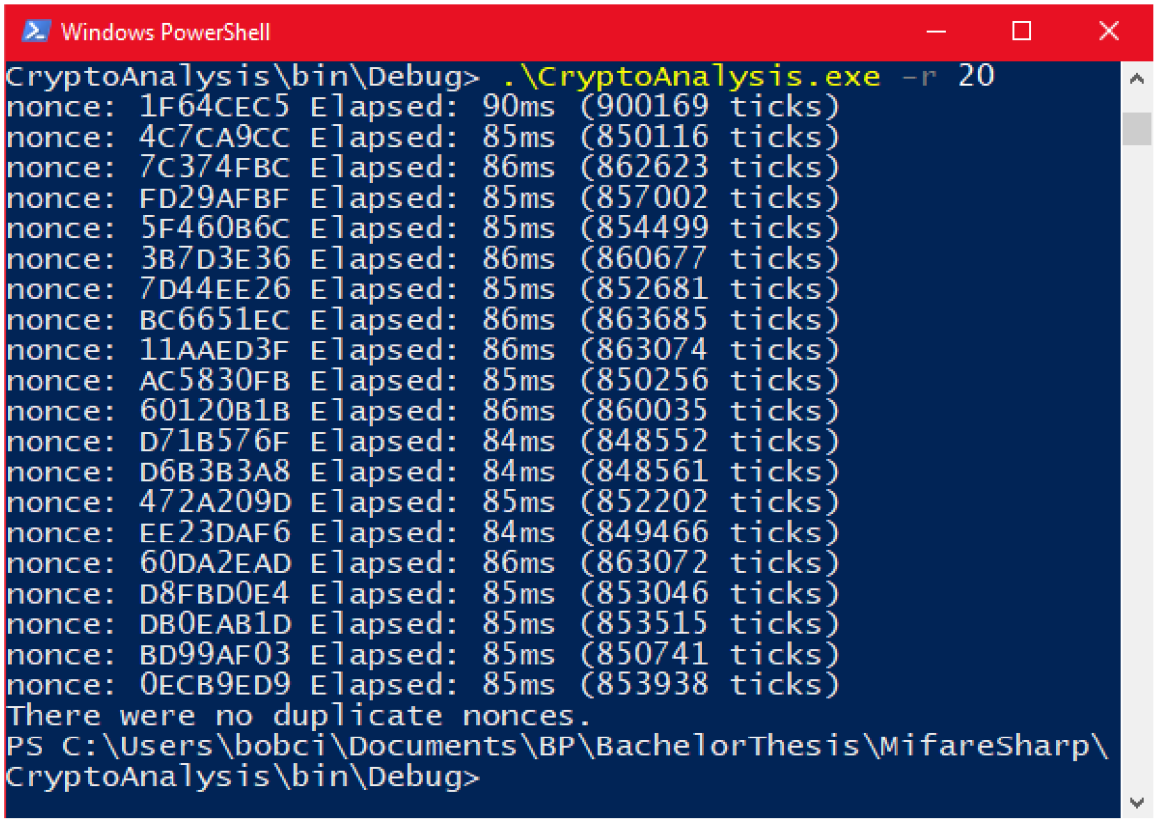
Program *CryptoAnalysis.exe* využívá knihovny `MifareModules` (viz část 6.2), jejíž moduly jsou inicializovány hned po startu. Dále je inicializován prázdný seznam výzev typu `List<string>` a třída `Stopwatch` pro stopování času jednotlivých pokusů. Inicializační část následuje jádro programu - cyklus opakující pokusy o autentizaci. Na začátku cyklu jsou spuštěny stopky a elektromagnetické pole zařízení. Po krátkém čekání je zahájena antikolizní fáze. Pokud při ní nastane chyba (například karta nestihne odpovědět), je algoritmus vrácen zpět na začátek cyklu. Na antikolizní fázi navazuje pokus o autentizaci. Ten je realizován odesláním příkazu *0x60 00*, tedy autentizace vůči prvnímu sektoru. Voláním metody `CardModule.AuthenticateForBlock("00")` získáme přijatou výzvu karty. V tuto chvíli je vypnuto elektromagnetické pole a výzva je uložena do seznamu výzev. Před návratem na začátek cyklu se ještě vytiskne informace o přijaté výzvě a čase cyklu. Na konci programu jsou v seznamu přijatých výzev vyhledány duplikáty a vypsán výsledek.

Chování programu lze měnit vstupními parametry. Nejdůležitější je parametr "-p", kterým lze nastavit jméno portu, na kterém poslouchá Chameleon Mini. Při nezadání parametru je nastaven port "COM3". Počet pokusů o autentizaci nastavuje parametr "-r"(standardně 100 pokusů). Mezi každým odesláním zprávy při antikolizní a autentizační fázi lze nastavit čekání pomocí "-w" v milisekundách. Při odesílání zpráv těsně při sobě karta nemusí odpovědět včas, standardní čekací doba je 2ms.

7.1.2 Provedení

Program postupně provádí pokusy o autentizace a zaznamenává si výsledné výzvy karty. Po dokončení zadaného množství pokusů jsou v seznamu výzev nalezeny duplikáty. Jak je vidět na obrázku 7.1, žádné duplikáty nalezeny nebyly. Toto chování se opakovalo i při zvýšení počtu pokusů. Kapitola 4.2 představuje řešení slabiny karet vydáním novější verze Mifare Classic EV1. Na první pohled by se tak mohlo zdát, že cílová karta je tato novější verze, ale identifikace Chameleonem, příkazem `Identify`, říká, že karta je původní Mifare Classic. Kromě použití příkazu `Identify` si lze také všimnout UID karty. Zatímco délka UID testované karty jsou 4B, nové karty mají UID 7B dlouhé[21]. Z toho vyplývá, že karta samotná není důvod neúspěchu analýzy.

Důvodu si můžeme všimnout na obrázku výsledku. Můžeme na něm vidět výsledek dvaceti pokusů o autentizaci. Každý řádek vyjadřuje jeden pokus. Na řádku jsou vždy tři veličiny - přijatá výzva, uplynulý čas daného cyklu v milisekundách a v počtu cyklů procesoru. Předpokladem pro úspěch analýzy je precizní časování v řádu mikrosekund. Jednotlivé pokusy se neliší jen v počtu cyklů procesoru, ale také v milisekundách. Tyto nerovnosti jsou způsobeny možným přepínáním kontextu procesů plánovačem CPU a komunikace sériového



Obrázek 7.1: Výsledek časové kryptoanalýzy (pro názornost pouze 20 pokusů)

portu, který je připojen přes sběrnici USB. Všechna data přenášená tímto způsobem jsou nahrávána do vyrovnávací paměti a jejich doba odeslání se může lišit[2].

Oba důvody by mohly být vyřešeny použitím specializovaného hardware s možností preciznějšího časování pokusů. Tím může být například už zmíněný Proxmark 3, který obsahuje programovatelná hradlová pole FPGA[25]. Analýza by tak mohla být prováděna ze zařízení, čímž by se eliminoval přenos přes USB a logika by byla naprogramována přímo v hardware, bez jiných procesů vyžadujících pozornost procesoru.

Tato demonstrace se do reálného prostředí příliš nehodí a je vhodná spíše do laboratorních podmínek. Je využitelná při kryptografické analýze karet Mifare Classic a jejich šifry CRYPTO1.

7.2 Emulace karet

Předmětem další demonstrace je schopnost zařízení Chameleon Mini emulovat karty. Emulací rozumíme schopnost zařízení imitovat kartu při komunikaci s cizím čtecím zařízením. Aby mohla být emulace dostatečně důvěryhodná, musí zařízení pracovat stejně jako původní karta a mít k tomu dostatečné množství dat karty. Chameleon v současné době dokáže emulovat až 5 různých typů karet. Množství potřebných dat závisí na složitosti systému. Některé systémy mohou pracovat přímo s datovými či hodnotovými bloky karty, jiným stačí pouze autentizace karty. Samotným zařízením Chameleon Mini lze získat pouze UID karty. Pro získání celé paměti karty jsou potřeba sofistikovanější zařízení, například ProxMark 3 nebo standardní čtecí zařízení ve spojení s kryptoanalytickými skripty Mi-

fare Classic Universal toolKit (MFCUK) a Mifare Classic Offline Cracker (MFOC)[2]. Tyto skripty využívají slabin popsaných v kapitole 4. Můžeme ale emulovat jen prázdnou kartu s nastaveným UID. Toto řešení nebude fungovat na všechny systémy, pouze na ty, kterým k ověření stačí úspěšná antikolizní fáze s platným UID.

7.2.1 Nastavení zařízení Chameleon Mini

Pro emulaci karty je nutné nastavit Chameleon Mini do správné konfigurace. Nejprve zjistíme typ emulované karty příkazem `Identify`.

```
Config=ISO14443A_READER
100:OK
Identify
101:OK WITH TEXT
Mifare Classic 1k
ATQA: 0400
UID: 9E314C87
SAK: 08
```

Výpis 7.1: Záznam postupu identifikace karty

Správně nastavíme konfiguraci a UID zařízení a je možné přistoupit k demonstraci.

```
Config=MF_CLASSIC_1K
100:OK
Uid=9E314C87
100:OK
```

Výpis 7.2: Záznam nastavení emulace karty

V lednu 2018 byla do firmware Chameleon Mini implementována funkce zjednodušující emulaci karet. Příkaz `Clone` přepne zařízení do konfigurace čtečky, identifikuje přiloženou kartu a správně nakonfiguruje zařízení. Tato funkce lze také navázat na tlačítko (`Rbutton=clone`) a klonovat karty přímo v terénu, bez použití nápadného počítače.

7.2.2 Provedení útoku

Po zapnutí se Chameleon chová stejně jako nakonfigurovaná karta a přiložením ke čtecímu zařízení odpovídá na jeho dotazy. Tento test skončil úspěšně. Po přiložení se světlo na čtecím zařízení rozsvítilo zeleně (patrné i na obrázku 7.2), signalizující úspěšnou identifikaci uživatele. To potvrdilo i následné odemčení vstupních dveří. K identifikaci a vstupu na fakultu tak stačí pouze úspěšná antikolizní fáze. Systém tak získá pouze UID uživatelovy karty a případná další potřebná data získá z vnitřní databáze. To můžeme vidět i na záznamu komunikace čtecího zařízení (viz výpis 7.3). Tento výpis zobrazuje část z logů zařízení Chameleon Mini. Každý záznam (řádek) obsahuje čtyři informace. První dvě se týkají času přijetí, přesněji čas přijetí od startu zařízení a rozdíl časů aktuálního a předchozího přijetí. Druhé dvě se týkají velikosti a obsahu přijatých zpráv. Za povšimnutí stojí, že neproběhne pouze jedna identifikace uživatele. Čtecí zařízení provádí antikolizní algoritmus zhruba každých 100ms.

```
04467 ms < +102 ms>: (1 bytes) [26 ]
04468 ms < +1 ms>: (2 bytes) [9320 ]
04470 ms < +2 ms>: (9 bytes) [93709e314c87644b91 ]
```

Obrázek 7.2: Testování emulace karty

```
04471 ms < +1 ms>: (4 bytes) [500057cd ]
04573 ms < +102 ms>: (1 bytes) [26 ]
04573 ms < +0 ms>: (2 bytes) [9320 ]
04575 ms < +2 ms>: (9 bytes) [93709e314c87644b91 ]
04576 ms < +1 ms>: (4 bytes) [500057cd ]
```

Výpis 7.3: Část komunikace čtecího zařízení

Po prolomení CRYPTO1 se systémy využívající karty s tímto šifrováním spoléhají pouze na UID, které je uloženo jen pro čtení. Jak je ale vidět z provedeného útoku, není to nejbezpečnější metoda. Chameleon Mini lze za předchozího správného nastavení použít úplně samostatně (bez počítače k ovládání) a provádět tento typ útoku přímo v terénu. Útočník tak pomocí tohoto zařízení může, s trochou nenápadnosti, napadnout jakýkoliv systém využívající podporované karty. Kromě dedikovaných zařízení na emulaci karet, jako

je zde použité Chameleon Mini, se dají sehnat čínské kopie karet Mifare Classic s měnitelným UID.

Ke zvýšení zabezpečení systému z pohledu chytrých karet doporučuje i sama firma NXP Semiconductors využít novější typy karet se silnějším šifrováním. Je ale nutné si uvědomit, že bezpečnost se nedá zařídit pouze moderními kartami. Důležitá je celková architektura zabezpečení systému. Není nutné opatřovat si složité zařízení schopné jakéhokoliv útoku, pokud si stačí pro vstup vybrat například jiné dveře, které nevyžadují identifikaci. Dalšími způsoby podobného prolamování bezpečnosti se zabývá sociální inženýrství.

7.3 Relay útok

Poslední demonstrací je takzvaný relay útok. Jedná se o man-in-the-middle útok. Většina typů autentizací je založena na ujištění, že oba účastníci znají sdílený tajný klíč. Po autentizaci je veškerá komunikace šifrována tímto tajným klíčem a pokud by útočník odposlechl takovou komunikaci, bez znalosti klíče jsou přenášená data stále zabezpečena. Bezkontaktní karty pracují na dálku (do 10cm) a jsou aktivovány vstupem do blízkosti čtecího zařízení. Útočník tak může bez vědomí vlastníka navázat spojení s kartou pomocí vlastního čtecího zařízení a přeposílat komunikaci druhému útočníkovi u vzdáleného čtecího zařízení. Přijatou komunikaci čtecímu zařízení druhý útočník předá, to usoudí, že karta, a tedy i její legitimní uživatel, jsou v blízkosti a vpustí útočníka. Ve skutečnosti může být karta vzdálená několik kilometrů. Data mezi kartou a čtecím zařízením lze posílat jakýmkoliv dostatečně rychlým, přenosovým médiem (Wi-Fi, 4G mobilní síť apod.). Přenášené zprávy sice nebudou pro útočníka čitelné, dokud je ale dokáže nezměněné přeposílat včas, tak mu to nevadí. Výhodou takového útoku je nezávislost na šifrovacím algoritmu systému.

Pro uskutečnění relay útoku jsou nutná minimálně dvě zařízení, v tomto případě Chameleon Mini, propojená komunikační vrstvou. První zařízení se tváří jako legitimní čtecí zařízení a komunikuje s kartou oběti. Toto zařízení je označováno jako Mole. Druhé zařízení komunikuje s legitimním čtecím zařízením a tváří se jako karta oběti. V literatuře bývá označováno jako Proxy.[9]

Obrázek 7.3: Schéma relay útoku

Program nejprve provede antikolizní fázi mezi Mole a kartou a poté, s daty přijatými od karty, provede antikolizní fázi mezi Proxy a cílovým čtecím zařízením. Z předchozí demonstrace víme, že toto bude stačit k úspěšnému útoku na bezpečnostní systém dveří. Relay útok je ale univerzální a po antikolizní fázi pokračuje v přenášení komunikace.

7.3.1 Implementace

Implementace útoku je rozdělena na dvě hlavní části. První část je zjednodušený celý útok tak, že zařízení Mole i Proxy jsou připojeny k jednomu programu a data se vyměňují takřka okamžitě v jednom procesu. Jedná se o ověření konceptu útoku. Druhá část už zahrnuje komunikační vrstvu, a je tak možné komunikovat s jiným počítačem. Obě části jsou obsaženy v programu *MifareProxy.exe* a je mezi nimi možné přepínat pomocí argumentů. Implementace opět využívá knihovny `MifareModules`. Aby mohly obě dvě zařízení využívat příkaz `Send`, jsou přepnuty do konfigurace čtecího zařízení.

Jednoduchý relay útok

Jednoduchý relay útok je standardní chování programu a je spíše demonstračního charakteru. Nejprve jsou inicializovány pomocné moduly pro komunikaci s oběma zařízeními Chameleon Mini. Program předpokládá, že jsou na portech "COM3" pro Proxy a "COM4" pro Mole. Případně je lze změnit argumenty "–pcom", respektive "–mcom". Následuje relay útok. Nejprve je provedena antikolizní fáze s kartou oběti. S odpověďmi je provedena antikolizní fáze také s cílovým čtecím zařízením. Poslední zpráva obsahující výzvu k autentizaci je odeslána kartě a ve smyčce se pak vyměňují data mezi zařízeními Mole a Proxy.

Relay útok přes síť

Pro relay útok s komunikací přes síť je nutné nastavit program do módu Proxy argumentem "-p" a do módu Mole argumentem "-m". Jméno sériového portu se nastaví stejným způsobem jako u jednoduché verze. Komunikace probíhá přes TCP soket, kdy Proxy je klient a připojuje se k serveru Mole. Adresa IP a port serveru je potřeba nastavit v obou instancích programu argumenty "–mip" a "–mport".

Nejprve se opět inicializují obě zařízení Chameleon Mini. Po inicializaci vytvoří Mole server a počká na připojení Proxy. Připojením je vše připraveno k útoku. Jeho logika je prakticky stejná jako u jednoduchého útoku, s tím rozdílem, že data jsou posílána sítí a ne pouze sdílenou pamětí. Při komunikaci po síti se může vyskytnout problém s dlouhou dobou doručení, neboli latencí. Tento problém je nejmarkantnější u antikolizní fáze, neboť při ní mají čtečky striktní časování. Nejprve se tedy provede antikolizní fáze s kartou, která nemá striktní časování, a přijatá data karty se odešlou Proxy. Proxy tak má k dispozici veškerá data nutná k této fázi a v jejím průběhu na straně Proxy nemusí probíhat časově náročná síťová komunikace s Mole.

7.3.2 Provedení

```
** Chameleon Connected via COM3 **
** Chameleon Connected via COM4 **
>ReqA
Mole - SEND 0007 26
<NO DATA
>ReqA
Mole - SEND 0007 26
<0400
>Select
Mole - SEND 0010 9320
```

```
<9E314C8764
>Select(9E314C8764)
Mole - SEND 0048 93709E314C87644B91
<Ack (08B6DD)
*** anticollision on mole passed ***
Proxy - SEND 0010 0400
Proxy - < Received NO DATA
Proxy - SEND 0028 9E314C8764
Proxy - < Received NO DATA
Proxy - SEND 0018 08B6DD
Proxy - < Received NO DATA
```

Výpis 7.4: Výstup programu MifareProxy.exe při demonstraci

Z výstupu programu při provádění relay útoku vidíme, že první část - antikolizní fáze mezi Mole a kartou, skončila úspěšně. Druhá část už úspěšná nebyla a celý útok tak skončil neúspěchem. Proxy sice odeslala data karty, ale čtecí zařízení neodeslalo žádnou odpověď. Z neodemčených dveří lze usoudit, že čtecím zařízením nebyla přijata data. Chameleon Mini v roli Proxy v útoku předstírá, že je karta. Přepnutím zařízení do konfigurace karty se však deaktivuje příkaz `Send` a nelze tak odesílat žádná data[14].

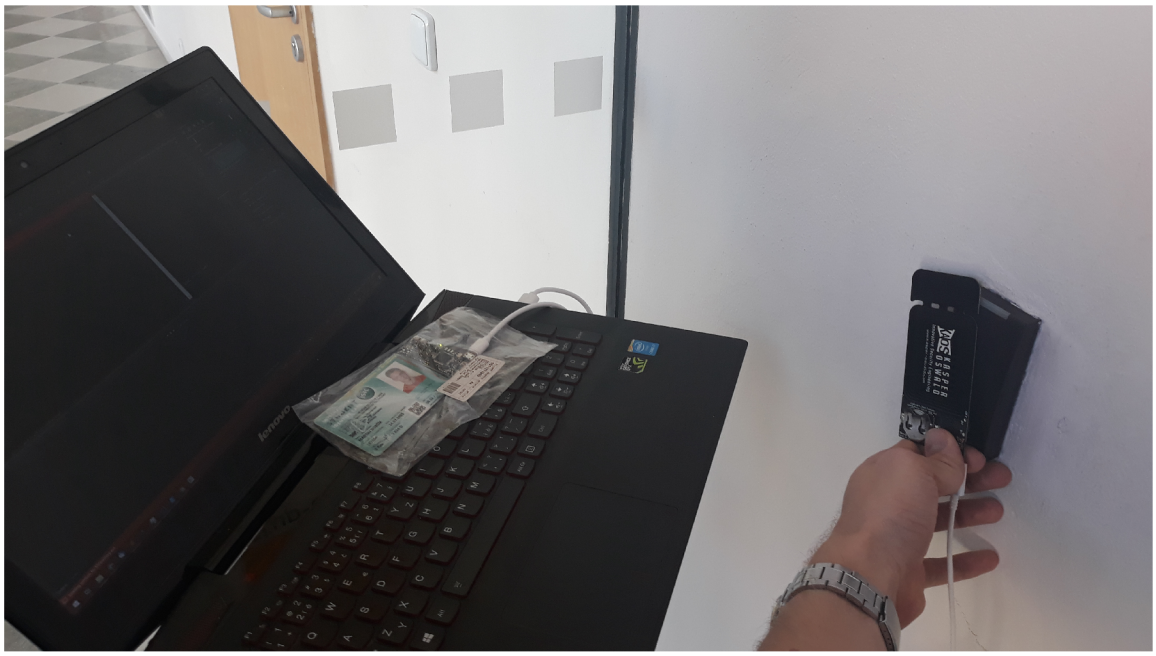
```
Config?
101:OK WITH TEXT
ISO14443A_READER
Send 52
101:OK WITH TEXT
0400
0010
PARITY OK
Config=MF_CLASSIC_1K
100:OK
Send 52
201:INVALID COMMAND USAGE
```

Výpis 7.5: Příkaz Send po přepnutí do konfigurace karty nefunguje

7.3.3 Vyhodnocení

Neúspěšně odeslaná data lze připsat zařízení Chameleon Mini, které v konfiguraci čtecího zařízení vytváří vlastní elektromagnetické pole[14], jímž komunikuje. Cílové čtecí zařízení však očekává modulaci jeho nosné vlny (viz část 2.3.1), což se neděje. Z předchozí demonstrace víme, že Chameleon Mini dokáže se čtecím zařízením komunikovat, avšak pouze v konfiguraci emulace karty. Tato konfigurace pouze pasivně emuluje kartu a nelze se aktivně zapojovat do komunikace pomocí komunikačního portu[14]. Relay útok za pomoci zařízení Chameleon Mini je tak možný pouze částečně, v roli Mole. Pro roli Proxy je nutné využít jiného zařízení, například již zmiňovaného Proxmark3 nebo standardní čtecí zařízení umožňující emulaci karet[2]. Druhou možností je implementace odesílání dat v konfiguraci emulace karet do open source firmware Chameleon Mini. Zde by se ovšem mohl vyskytnout problém z první demonstrace (viz část7.1), kdy USB sběrnice zvyšuje latenci.

Vzhledem k tomu, že relay útok je nezávislý na šifrování karty, je zabezpečení proti němu omezené. Nejzákladnější ochranou je uzavřít kartu při přenosu Faradayovy klece, které zabrání vstupu elektromagnetického pole ke kartě. Při používání je však nutno kartu z klece



Obrázek 7.4: Testování relay útoku

vyjmout, čímž vzniká krátké okno k provedení útoku. Další metodou může být zadáváním PIN kódu při jakékoliv manipulaci s kartou. Dále je možné detekovat vzdálenost karty a relay útok pomocí takzvaných distance-bounding protokolů, které ale zatím nevyužívá žádný standard.

Tento útok je možné v praxi využít. Jsou potřeba dva útočníci. První, v roli Mole, se musí přiblížit ke kartě oběti na dostatečně dlouhou dobu potřebnou k přenesení informací. To je možné například v plné hromadné dopravě nebo při čekání v řadě. Samotná doba přenesení jedné autentizace je krátká, je ovšem nutné vzít v úvahu opakovaní útoku vícekrát z důvodu možného narušení přenosu. Útočník v roli Proxy musí nenápadně přiložit zařízením do blízkosti čtečky. Na takovém místě většinou bývá jiné zabezpečení v podobě kamer či bezpečnostní služby a neznámý předmět by vyvolal rozruch. Je ale možné zařízení zavřít do peněženky či brašny kde bude nenápadné, stejné jako normální karty.[9]

Kapitola 8

Závěr

Cílem této práce bylo seznámit se s kartami Mifare Classic se zaměřením na jejich zranitelnosti a demonstrovat vybrané zneužitelnosti za pomoci zařízení Chameleon Mini.

Studiu technologie RFID a kartám Mifare Classic byly věnovány první dvě kapitoly. Samotné zranitelnosti těchto karet byly rozebrány v kapitole následující. Zadáním bylo určeno, že demonstrace vybraných zranitelností má být provedeno pomocí zařízení Chameleon Mini, bylo tedy nastudováno také jeho využití a možnosti. Na základě informací o komunikačním protokolu a šifrovacím algoritmu těchto karet byly navrženy a provedeny tři demonstrace, z nichž dvě vyžadovaly také implementační část. Ta byla provedena v jazyce C#.Net a byla vytvořena pomocná knihovna pro usnadnění práce se zařízením Chameleon Mini.

Časová kryptoanalýza se zabývala nedostatečnou náhodností generátoru náhodných čísel, který používá šifrovací algoritmus karty. Následoval útok nevyžadující implementaci, při kterém zařízení Chameleon Mini naklonuje a emuluje cizí kartu. Jako poslední byl proveden takzvaný relay útok, který přenáší komunikaci mezi kartou oběti a legitimním čtecím zařízením. Z těchto demonstrací lze považovat pouze útok s emulovanou kartou jako zcela úspěšný. Zbylé dvě demonstrace se nepodařily kvůli nedostatečným schopnostem zařízení Chameleon Mini. Všechny demonstrace byly provedeny v reálném prostředí na dveřích osazených čtecím zařízením HID RP10.

Pokračovat v této práci lze více směry. Jelikož zařízení Chameleon Mini nemělo dostatečné schopnosti pro dva ze tří útoků, nabízí se možnost provést stejné útoky za použití jiných zařízení, případně rozšířit firmware Chameleon Mini o potřebnou funkcionalitu, která prováděné útoky umožní. Dále je možné provést analýzu a útoky na jiné typy karet, nebo rozšířit paletu útoků o další.

Literatura

[1] Blahut, R. E.: Stream ciphers. In *Cryptography and Secure Communication*, Cambridge: Cambridge University Press, 2014, ISBN 9781139013673, s. 181–217.

[2] Činčala, M.: *Provedení relay útoku na karty Mifare*. Bakalářská práce, Vysoké učení technické v Brně, Fakulta informačních technologií, 2012.
URL http://www.fit.vutbr.cz/study/DP/BP.php?id=13915

[3] De Koning Gans, G.; Hoepman, J.-H.; Garcia, F.: A practical attack on the Mifare Classic. 2008, ISBN 354085892X, ISSN 03029743, s. 267–282.
URL http://www.cs.ru.nl/~flaviog/publications/Attack.MIFARE.pdf

[4] Dobkin, D. M.: *The RF in RFID: UHF RFID in Practice*. Elsevier Inc., 2013, ISBN 978-0-12-394830-4.

[5] Garcia, F. D.; De Koning Gans, G.; Muijrers, R.; aj.: Dismantling Mifare classic. 2008, ISBN 3540883126, ISSN 03029743, s. 97–114.

[6] Garcia, F. D.; Rossum, P. v.; Verdult, R.; aj.: Wirelessly Pickpocketing a Mifare Classic Card. IEEE Publishing, 2009, ISBN 978-0-7695-3633-0, ISSN 1081-6011, s. 3–15.

[7] Goldreich, O.: *Foundations of cryptography*. Cambridge, UK New York: Cambridge University Press, 2003, ISBN 9780521830843.

[8] Hanačík, R.: Bezpečnost RFID. 2011.

[9] Hancke, G.: A practical relay attack on ISO 14443 proximity cards. 01 2005.
URL http://www.rfidblog.org.uk/hancke-rfidrelay.pdf

[10] Hind, D. J.: Radio frequency identification and tracking systems in hazardous areas. In *1994 Fifth International Conference on Electrical Safety in Hazardous Environments*, April 1994, ISBN 0-85296-614-8, doi:10.1049/cp:19940408.

[11] ISO Central Secretary: Identification cards – Contactless integrated circuit cards – Proximity cards. Standard, International Organization for Standardization, Geneva, CH, 2001.

[12] Janošík, T.: Emulátor UHF RFID tagu. Vysoké učení technické v Brně. Fakulta elektrotechniky a komunikačních technologií, 2015.

[13] Kasper & Oswald GmbH: Mifare Ultralight Family. 2016.
URL https://shop.kasper.it/chameleonmini/167/chameleonmini-revg-standard-red?c=5

[14] Kasper & Oswald GmbH: Chameleon-Mini Documentation. doxygen, 2018.
URL https://rawgit.com/emsec/ChameleonMini/master/Doc/Doxygen/html/index.html

[15] Kasper & Oswald GmbH: ChameleonMini - A Versatile NFC Card Emulator, and more... Kickstarter, PCB, 2019.
URL https://www.kickstarter.com/projects/1980078555/chameleonmini-a-versatile-nfc-card-emulator-and-mo/description

[16] Khan, M. A.; Sharma, M.; Prabhu, B. R.: A Survey of RFID Tags. *International Journal of Recent Trends in Engineering*, ročník 1, č. 4, 05 2009, copyright - Copyright Academy Publisher May 2009; Poslední aktualizace - 2010-07-15.
URL https://search.proquest.com/docview/594755305?accountid=17115

[17] Mayes, K. E.; Cid, C.: The Mifare Classic story. *Information Security Technical Report*, ročník 15, č. 1, 2010: s. 8–12, ISSN 1363-4127.

[18] Mayes, K. E.; Markantonakis, K.: *Smart Cards, Tokens, Security and Applications*. Boston, MA: Springer US, 2008, ISBN 9780387721972.

[19] Meijer, C.; Verdult, R.: Ciphertext-only Cryptanalysis on Hardened Mifare Classic Cards. 10 2015, s. 18–30, doi:10.1145/2810103.2813641.

[20] NXP Semiconductors: About Mifare. 2002-2019.
URL https://www.mifare.net/en/about-mifare/

[21] NXP Semiconductors: Mifare ICs. 2002-2019.
URL https://www.mifare.net/en/products/chip-card-ics/

[22] NXP Semiconductors: P40C072PU15: Secure smart card controller. 2006-2019.
URL https://www.nxp.com/pages/secure-smart-card-controller:P40C072PU15

[23] Palán, M.: Bezkontaktní čipové karty Českých drah. Vědeckotechnický sborník ČD, 2006.
URL https://vts.cd.cz/documents/168518/195495/2108.pdf/c5cedcdf-1327-4649-bdb3-2082037cda43

[24] Rankl, W.: *Smart card handbook*. Chichester: John Wiley & Sons, Čtvrté vydání, 2010, ISBN 978-0-470-74367-6.

[25] Roel Verdult, Gerhard de Koning Gans: Proxmark. online, 2017.
URL http://www.proxmark.org

[26] Rogaway, P.: Nonce-Based Symmetric Encryption. In *Fast Software Encryption*, editace B. Roy; W. Meier, Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, ISBN 978-3-540-25937-4, s. 348–358.

[27] Teepe, W.: Making the Best of Mifare Classic. 01 2008.
URL http://www.cs.ru.nl/~wouter/papers/2008-thebest.pdf

[28] Wand, R.: *RFID Explained: A Primer on Radio Frequency Identification Technologies*. Morgan & Claypool Publishers, 2006, ISBN 9781598291094.

[29] České dráhy, a.s.: Podmínky pro vydávání a používání In Karty. 2016. URL https://www.cd.cz/info/cim-se-ridime/-30814/