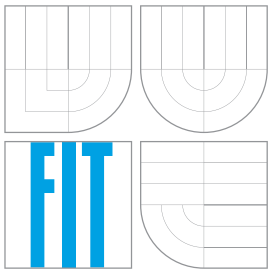
VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INTELLIGENT SYSTEMS

PŘIHLAŠOVÁNÍ DO OS UNIX POMOCÍ ČIPOVÝCH KARET MIFARE

LOGGING ONTO OS UNIX USING MIFARE SMART CARDS

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE — PETER KREMPA
AUTHOR

VEDOUCÍ PRÁCE — Doc. Dr. Ing. PETR HANÁČEK
SUPERVISOR

BRNO 2010

Abstrakt

Tato práce se zabývá využitím technologie bezkontaktních čipových karet při autentizaci v prostředí pracovní stanice s OS Linux. Shrnuje výhody a nedokonalosti této technologie a popisuje její další možné využití. V práci jsou diskutovány používané způsoby autentizace a navržený software pro přihlašování a uzamykání sezení využívající čipové karty Mifare.

Abstract

This thessis discusses the use of contactless smart card technology for authentication in an Linux desktop environment. It summarizes the advantages and imperfections of the technology and suggests its other possible uses. The paper describes the methods of authentication, and software designed for logging in and session locking, using Mifare smart cards.

Klíčová slova

bezkontaktní karta, RFID, autentizace, PAM, Mifare, Unix, Linux, ACR128, ISO 14443

Keywords

contactless smartcard, RFID, authentization, PAM, Mifare, Unix, Linux, ACR128, ISO 14443

Citace

Peter Krempa: Přihlašování do OS UNIX pomocí čipových karet Mifare, bakalářská práce, Brno, FIT VUT v Brně, 2010

Přihlašování do OS UNIX pomocí čipových karet Mifare

Prehlásenie

Prehlasujem, že som túto bakalársku prácu vypracoval samostatne pod vedením pána Doc. Dr. Ing. Petra Hanáčka. Uviedol som všetky literárne pramene a publikácie, z ktorých som čerpal.

.....
Peter Krempa
13.05.2010

Poďakovanie

Týmto by som chcel poďakovať pánovi Doc. Dr. Ing. Petrovi Hanáčkovi za užitočné rady a pripomienky k tejto práci. Ďalej by som chcel poďakovať mojím rodičom za gramatickú a štylistickú kontrolu.

Obsah

Kapitola 1

Úvod

Už od dávna sa využívajú rôzne spôsoby kontroly a zamedzenia prístupu. Používali sa najmä kľúče. Vývoj v oblasti elektroniky a techniky však časom umožnil vytvoriť zabezpečovacie systémy na elektronickom základe. Tie oproti klasickým systémom umožňujú okrem bezpečnostných funckií aj jednoduchú a vzdialenú administráciu a registráciu povolených a nepovolených prístupov. Automatická evidencia poskytuje možnosť okamžitej kontroly prístupov a minimalizuje množstvo chýb vznikajúcich pri manuálnom zadávaní údajov. Preto začali byť elektronické prístupové systémy postupne nasadzované ako evidencia dochádzky a prístupu. Taktiež sa zvyšovali požiadavky na rýchlosť vybavenia v silno vyťažených systémoch, ako MHD alebo dochádzkových systémoch. Súčastne bolo potrebné zamedziť falšovaniu a oklamaniu týchto systémov. Podobné incidenty spôsobovali a spôsobujú prevádzkovateľom takýchto systémov vysoké straty.

Dnes existujú zariadenia schopné sa unikátne identifikovať a prípadne aj uložiť nejaké informácie a to bez potreby kontaktu s technickým vybavením na kontrolnom bode. Tieto zariadenia pracujú na princípe rádiovýc vĺn a súhrnne sa tieto technológie označujú RFID – *Radio Frequency Identification*. Využívajú hlavne sa v odvetviach, kde je rozhodujúcim faktorom rýchlosť vybavenia a bezpečnosť riešenia nie je najkritickejším prvkom. V porovnaní s biometrickými systémami majú systémy bezkontaktnej identifikácie výhodu v rýchlosti spracovania transakcie a možnosti uložiť na transpondér dáta. Toto umožňuje vytvárať rozsiahe off-line systémy, ktoré sú založené na práci s konkrétnou hodnotou, ako napríklad elektronická peňaženka. Elektronická peňaženka zbavuje jej majiteľa nutnosti mať pri sebe hotovosť, čo je obmedzujúce hlavne v prípade častých platieb veľmi malých súm.

Cieľom tejto práce bude vytvoriť autentizačný systém pre OS unixového typu, ktorý bude využívať bezkontaktné transpondéry Mifare vo forme kariet od spoločnosti NXP Semiconductors. Pre účely tejto práce som zvolil operačný systém postavený na jadre Linux, keďže sa jedná o najpoužívanejší OS z rodiny UNIX.

Práca čitateľa postupne prevedie základmi požadovanými pre porozumenie technológii bezkontaktných kariet, systémom, ktoré ich využívaju a bezpečnostným rizikám spojeným s touto technológiou:

Druhá kapitola tejto práce rozoberá históriu bezkontaktných čipových kariet, ich výhody, nevýhody, použitie v praxi, ako aj konštrukciu kariet a čítacích zariadení pre prácu s týmito kartami. V tejto kapitole sa taktiež venujem technológii Mifare Classic ako predchodcovi štandardu ISO/IEC 14443 a jej nedokonalostiam.

Tretia kapitola analyzuje systémy autentizácie v OS Linux, ktoré sú veľmi podobné aj v ostatných OS založených na UNIX-e a uzamykanie aktuálnej relácie používateľa.

Štvrtá a piata kapitola popisuje analýzu, návrh a implementáciu systému autentizácie

užívateľa pomocou bezkontakntých čipových kariet Mifare v prostredí pracovnej stanice. Rozoberám tu problémy, ktoré sa objavili pri riešení tohoto projektu.

V šiestej kapitole stručne popíšem testovanie implementovaných aplikácii na pracovnej stanici a krátko zhrniem zistenia z hľadiska použitelnosti aplikácii. Popísaná je taktiež aplikácia pre meranie času "bežcov" využívajúca bezkontaktné čipové karty.

V závere sa budem venovať vyhodnoteniu splnenia cieľov práce, analýze bezpečnostných rizík aplikácii vytvorených v rámci tejto práce a ich možných riešení a možnostiam rozšírenia z úrovne pracovnej stanice na úroveň zabezpečenia celej siete.

Kapitola 2

Technológia bezkontaktných čipových kariet

Rýchly rozvoj techniky a miniaturizácie v minulom storočí dosiahol úrovne, kedy bolo možné umiestniť jednoduchý pamäťový obvod — čip — na malú plastovú kartu. Jedným z prvých využití boli telefónne karty vo Francúzsku v r. 1983. Dnes sú čipové karty každodennou súčasťou nášho života. Bezkontaktné čipové karty sú ďalším evolučným stupňom kariet kontaktných. Pre nasadenie bezkontaktnej technológie bolo potrebné vyvinúť čipy s nízkym príkonom, aby karty bolo možné napájať pomocou rádiových vĺn. Dnes sú dostupné karty obsahujúce plnohodnotný mikroprocesor a ktorých správanie je možné naprogramovať ľubovolne.

2.1 RFID

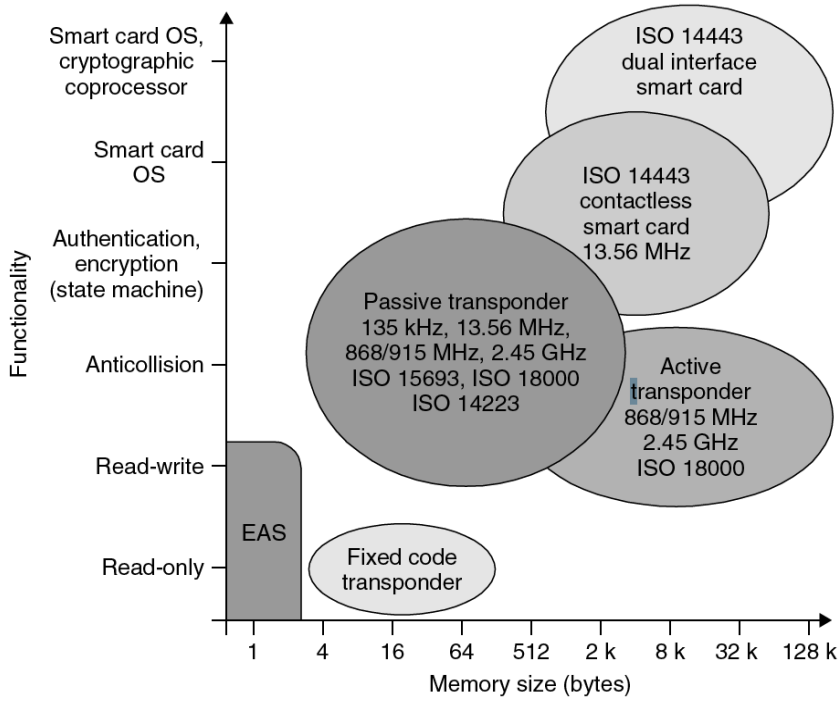
RFID[1] — identifikácia pomocou rádiových vĺn — je pojem, ktorý zastrešuje technológie schopné komunikovať s transpondérom[2] pomocou elektromagnetického vlnenia. Zahrňuje najjednoduchšie zariadenia, pre zabránenie krádežiam v obchode až po najzložitejšie bezkontaktné čipové karty, schopné vykonávať program a zložité kryptografické operácie. Podľa vyspelosti použitej technológie môžeme RFID systémy rozdeliť nasledovne:

Low End — systémy nízkej úrovne, veľmi jednoduché systémy schopné zistiť prítomnosť transpondéru v poli detektoru, preniesť jednoduchú preddefinovanú hodnotu, alebo výrobné číslo. Tieto systémy sa používajú ako náhrada čiarových kódov, prípadne pre bezpečnostné účely. Používa sa viac štandardov a komunikačných frekvencií.[4]

Mid Range — systémy strednej triedy, umožnujú uloženie dátových hodnôt a využívajú sofistikovanejšie princípy kontroly prístupu a šifrovania prenášaných dát. Čipy bývajú tvorené jednoduchým stavovým automatom a vybavené antikolíznym algoritmom, aby bolo možné v dosahu čítačky použiť viacej transpondérov. Používa sa viac štandardov a komunikačných frekvencií.[4]

[1] Angl. *Radio Frequency Identification.*

[2] Transpondér je zariadenie, ktoré vyšle predom definovanú alebo vygenerovanú správu na základe požiadavky[21].



Obr. 2.1: Rozdelenie RFID systémov. [4]

High End — systémy najvyššej triedy, obsahujú plne funkčný mikroprocesor schopný behu operačného systému pre karty alebo programu. Transpondéry sú schopné pokročilých kryptografických a autentizačných operácii ako aj uloženia značného množstva dát. Tieto systémy pracujú takmer výlučne na základe štandardu ISO/IEC 14443 na frekvencii 13.56 MHz.[4]

Transpondéry sa vyrábajú v rôznych tvaroch. Okrem kariet sa vyrábajú aj tzv. *tokeny*. Jedná sa o transpondéry, s iným tvarom obalu. Najčastejšie v tvare kľúčenky, samolepiacich etikiet, náramku, poprípade sú tieto transpondéry zabudované do iných zariadení. Existujú aj transpondéry v sklenom miniatúrnom púzdre, ktoré je možné chirurgicky umiestniť pod kožu človeka alebo zvieraťa.[4]

Obr. 2.2: Bezkontaktné karty a tokeny. Autor: Abel Technology Limited.

Táto práca sa ďalej bude zaoberať hlavne transpondérmi strednej a vyššej triedy vo forme plastových kariet formátu ID-1 podľa normy ISO/IEC 7810[12] s rozmermi 85.60 x 53.98 mm, ktoré komunikujú podľa normy ISO/IEC 14443 a pomocou proprietárnej technológie Mifare.

2.2 Nasadenie technológie bezkontakntých kariet

Karty sú svojími rozmermi predurčené pre nosenie v peňaženke, preto hlavné odvetvia, kde sa využívajú sú spojené s identifikáciou ľudí. Bezkontaktné čipové karty sú využívané najmä v prístupových systémoch v podnikoch, školách, knižniciach a pod. kde slúžia k identifikácii a autentizácii osôb. Veľmi často dochádza k zjednocovaniu systémov a preto sa rovnaké karty často používajú v stravovacích zariadeniach podnikov a menzách.

Ďalším odvetvím použitia týchto kariet je hromadná doprava ľudí (MHD, VHD)[3]. Výhodné sú nielen pre cestujúcich, ktorým odpadá potreba mať pri sebe vhodnú hotovosť na zakúpenie lístka, ale aj pre dopravcov, ktorým umožnuje sledovať presnejšie vyťaženie vozidiel a optimalizovať linky. Tak isto sa skracuje doba potrebná na vybavenie cestujúceho a oproti platbe v hotovosti je možné inštalovať do vozidla viac terminálov. [2]

Obr. 2.3: Bezkontaktný odbavovací systém pre MHD EMTEST. [2]

Medzi novinky patrí možnosť uskutočniť drobné platby pomocou platobnej karty vybavenej bezkontaktným čipom. Transakcie do hodnoty €20 sú autorizované iba priblížením karty do čítacej zóny čítačky. Z dôvodu zvýšenia bezpečnosti sa je v náhodných intervaloch vyžadované zadanie PIN-kódu. Táto technológia však ešte nie je v Českej Republike ani na Slovensku veľmi rozšírená, môžeme sa s ňou stretnúť v krajinách západnej európy, kde sa hlavne využíva v kaviarňach, trafikách a podobne, kde je veľkou výhodou rýchlosť spracovania transakcie. [16]

2.3 Biometrické pasy

Z dôvodu množiacich sa teroristických útokov a iných bezpečnostných hrozieb sa rozhodlo o uložení určitých biometrických údajov do cestovných pasov, aby bolo možné lepšie overiť

[3]Mestská Hromadná Doprava, Verejná Hromadná Doprava

totožnosť a zabránilo sa falšovaniu cestovných pasov. Na tento účel sú tzv. biometrické pasy vybavené čipom bezkontaktnej technológie podľa normy ISO/IEC 14443 s minimálnou kapacitou 32 KiB pamäte, na ktorom je uložená fotografia držiteľa, jeho osobné údaje a poprípade ďalšie biometrické vlastnosti[7].

Z dôvodu obáv o bezpečnosť sú v týchto čipoch použité pokročilé bezpečnostné princípy. Obal pasu je zvyčajne pokrytý materiálom, ktorý neumožňuje komunikáciu s čipom ak je pas zatvorený. Aby bolo možné prečítať údaje z pasu je potrebné poznať šifrovací kľúč, ktorý je odvodený z osobných údajov držiteľa pasu a je možné ho vygenerovať na základe optického rozpoznania strojovo čitateľnej zóny. Toto spolu s šifrovaním spojenia medzi čítačkou a pasom znemožnuje neautorizované čítanie údajov na čipe, a zároveň nevyžaduje špeciálnu databázu prístupových kľúčov ku čipu. Údaje na čipe sú chránené proti zmene použitím digitálneho podpisu. Aby nebolo možné sledovať konkrétny čip v pase, odpovedá tento čip na každú požiadavku o identifikáciu iným náhodne vygenerovaným "unikátnym" identifikátorom. Ukázalo sa však, že niektoré tieto postupy neboli implementované dostatočne kvalitne a preto je napríklad možné v niektorých prípadoch zistiť národnosť držiteľa pasu podľa odpovede na identifikáciu pasu[23]. [7]

2.4 Výhody použitia bezkontaktných kariet

Medzi hlavné výhody bezkontaktných kariet patrí možnosť pracovať s kartou aj vnútri peňaženky alebo iného obalu, čo prináša komfort užívateľom. Taktiež nedochádza k oxidovaniu kontaktov karty a jej opotrebeniu z dôvodu stáleho vyťahovania a zasúvania do čítačky. Predchádza sa aj poškodeniu kontaktov čítačky. Výhodu bezkontaktnej činnosti je možné využiť aj pri identifikácii tovaru v automatických skladiskách. Konštrukcia bezkontaktnej čítačky dovoľuje vyrobiť vodotesné vyhotovenia, ktorých prínos je najmä v *čistých priestoroch*[4] a pri vonkajšom použití. Rýchle vybavenie transakcie umožnuje kratšie čakacie doby na kontrolných bodoch a tým pádom menší počet odbavovacích miest.

2.5 Nevýhody použitia bezkontaktných kariet

Ako hlavnú nevýhodu použitia bezkontaktných kariet možno vnímať bezpečnosť. S kartou je možné komunikovať na diaľku[5] a bez vedomia majiteľa, čo poskytuje širšie možnosti zneužitia ako v prípade krádeže karty. V prípade krádeže je vačšia pravdepodobnosť, že bude skoro zistená a karta bude zablokovaná. Keďže sa karta identifikuje unikátnym výrobným číslom, je možné toto zneužiť na sledovanie osoby vlastniacej kartu a tým pádom narušenie jej súkromia.

Z hľadiska konštrukcie karty možno ako nevýhodu považovať nižšiu mechanickú odolnosť na ohýbanie, kedy môže dôjsť ku poškodeniu antény. To má väčšinou za následok zníženie komunikačnej vzdialenosti, alebo úplnú nefunkčnosť karty.

2.6 Štandard ISO/IEC 14443

Karty komunikujúce s čítačkou na frekvencii 13.56 MHz definuje štandard ISO/IEC 14443. Skladá sa zo štyroch častí, ktoré popisujú mechanické a elektrické vlastnosti čipu a antény a

[4] Priestory s kontrolovanou hygienou. Napr.: zdravotnícke zariadenia, farmaceutická výroba.

[5] Štandardná vzdialenoť pre komunikáciu s kartou je asi 10–15cm, existujú však čítačky schopné prečítať kartu na vzdialenosť niekoľkých metrov.

protokoly a mechanizmy využívané pri komunikácii s čítačkou. Transpondéry tento štandard označuje ako **PICC** – *Proximity Integrated Circuit Card* a čítačky ako **PCD** – *Proximity Coupling Devices*. Technológia je stavaná na bízke vzdialenosti komunikácie[6] do 10 cm. Východzia komunikačná rýchlosť medzi kartou a čítačkou je 106 Kbps, ktorá je povinná pri antikolíznom mechanizme, vyššie rýchlosti sú voliteľné.[19]

Časť 1 definuje rozmery kariet (odvolávajúc sa na štandard ISO 7810[12]), odolnosť voči mechanickému namáhaniu, magnetickým poliam, UV a röntgenovému žiareniu a rozsah teplôt prostredia (0 – +50°C), pri ktorom musí byť karta schopná pracovať.[19][8]

Časť 2 popisuje charakteristiku indukčného spoja pre prenos napájania a komunikáciu medzi PICC a PCD pomocou frekvenčne modulovaného poľa na frekvencii 13.56 MHz +/- 7 kHz. Ďalej definuje 2 transportné protokoly (Type A a Type B), moduláciu, časovanie a kódovanie dátového spoja. Protokoly A a B sa líšia v kódovaní bajtov, antikolíznych mechanizmoch, ale aj v modulácii napájacieho poľa.[19][9]

Časť 3 pojednáva o mechanizmoch inicializácie karty a spôsobe identifikácie a výberu aktívnej karty, ak sa v dosahu čítačky vyskytuje kariet viac (antikolízny mechanizmus). Karta pri vstupe do napájacieho poľa očakáva výzvu od čítačky, na základe ktorej sa identifikuje svojím unikátnym číslom, ktoré sa následne využije pri antikolíznom protokole.[19][10]

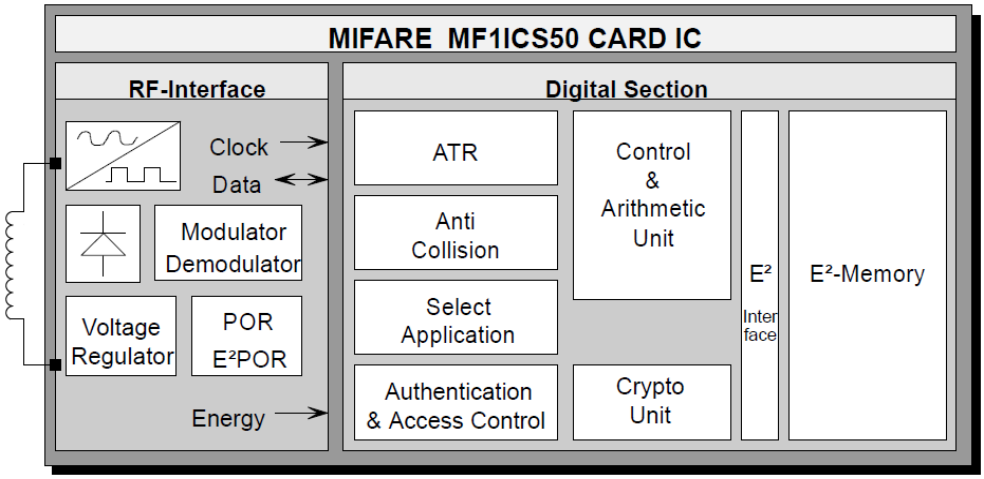
Časť 4 špecifikuje *half-duplex*[7] blokový protokol (T=CL) pre vysokoúrovňovú komunikáciu, nezávislú na nižších vrstvách. Protokol rieši hlavne opravu chybových stavov, rozdelovanie veľkých blokov dát na menšie fragmenty a vyjednávanie vyššej rýchlosti komunikácie. Časť 4 štandardu ISO/IEC 14443 nie je povinná, a preto existujú karty nepodporujúce protokol T=CL.[19][11]

2.7 Napájanie a konštrukcia karty

Na rozdiel od aktívnych transpondérov, ktoré majú vlastný zdroj napájania a signál indukovaný v ich anténe slúži len na ich aktiváciu a komunikáciu, musia pasívne transpondéry, akými sú aj bezkontaktné čipové karty, získavať energiu pre svoju prevádzku čisto z energie indukovanej v anténe. Indukčná väzba, fungujúca na princípe transformátoru, je tvorená dvojicou antén vo vzájomnej rezonancii, čo zvyšuje indukované napätie na anténe transpondéru. Rezonančná frekvencia obvodu transpondéru musí byť veľmi precízne vyladená, aby bol energetický zisk čo najväčší. Keďže napätie indukované na anténe sa mení vo veľkom rozsahu podľa vzdialenosti k čítačke, polohy oboch antén a daľších faktorov, je potrebné získané napätie usmernením a stabilizovaním pripraviť pre obvod karty. Používané regulátory musia mať nízke straty pri usmerňovaní a udržovaní napätia, keďže sa jedná o vzácny zdroj pre transpondér.[4]

Časť energie vyžiarenej čítačkou sa odráža od karty (a ostatných predmetov, to však nie je podstatné) a je vyžiarená späť k čítačke[8]. Súčasťou transpondéru je obvod, ktorý umožnuje meniť impedanciu rezonančného obvodu antény. Ovládaním vstupu tohoto obvodu je odrazené elektromagnetické pole modulované a pomocou neho sú prenášané dáta z karty

[6] Z angl. pojmu *proximity*.
[7] Umožnujúci komunikáciu oboma smermi, nie však súčastne.
[8] Princíp rádiolokátoru.



Obr. 2.4: Schéma bezkontaktej karty Mifare. Autor: NXP Semiconductors

späť do čítačky. Tento signál je veľmi slabý, avšak v čítačke je možné ho zosiliť a extrahovať z neho dáta. Výhodou tohoto prístupu je jednoduchosť realizácie v karte a nízke energetické nároky na prevádzku.[4]

2.8 Kombinovaná karta

Kombinovaná karta[9] obsahuje čip s rozhraním pre kontaktné aj bezkontaktné rozhranie a kombinuje ich výhody. Jedným z prvých pokusov o vytvorenie takejto karty, keď ešte nebolo možné vyrobiť procesor, so spotrebou dostatočne nízkou pre napájanie bezkontakným rozhraním, bola karta Mifare-plus. Obsahovala bezkontaktné rozhranie a logiku Mifare, kontaktné rozhranie s mikroprocesorom a proprietárnym operačným systémom a zdieľanú pamäť EEPROM, ku ktorej pristupovalo práve aktívne rozhranie. Dnes je však možné oddeleniť rozhrania karty od spoločného procesoru a softwaru a tým pádom je technológia pripojenia karty pre aplikáciu abstraktná. Hlavnou výhodou je možnosť rozšírenia existujúcej infraštruktúry kontaktných čítačiek bez jej kompletnej zmeny za bezkontaktné.[4]

Jediným rozdielom medzi použitými rozhraniami je elektrický výkon, ktorý je schopné dodať pre obvody karty. Bezkontaktné rozhranie ISO/IEC 14443 je schopné procesoru karty dodať 5mW pri maximálnej prípustnej vzdialenosti od čítačky, kým pri použití kontaktného rozhrania je možné využiť až 300mW (ISO 7816-3 Trieda A: 5 V, 60mA). Preto bolo potrebné pri návrhu čipu karty použiť moderné technológie ako nízkopríkonovú logiku alebo jednotku riadenia napájania, ktorá umožnuje deaktivovať nepoužívané časti čipu.[4]

2.9 Personalizácia kariet

Pred vydaním karty jej držiteľovi je potrebné túto kartu zviazať s identitou držiteľa – personalizovať. Na kartu býva zväčša vytlačené meno a fotografia držiteľa karty a do dátovej oblasti čipu nahraté informácie, ktoré vyžaduje daný systém a umožňuje to technológia karty. V prípade ak sa jedná o elektronickú peňaženku, je ešte potrebné nahrať výšku kre-

[9] Angl. *Dual Interface Smartcard.*

ditu. Dôležité je však, aby operácie, ktoré by mohli poškodiť, či už bezpečnostne alebo finančne, majiteľa systému boli ochránené a oddelené od bežných operácii s kartou. Dôležitým krokom personalizácie karty je preto inicializácia bezpečnostných prvkov, ktoré karta podporuje, ako napríklad zmena šifrovacích kľúčov a obmedzenie prístupu k niektorým operáciam.[3]

V mnohých prípadoch nie je využitá celá kapacita karty, a preto je možné karty použiť aj pre viac rôznych aplikácii. Vtedy je zväčša potrebné karty personalizovať u každého poskytovateľa služby separátne, pretože systémy pravdepodobne nebudú kompatibilné. Väčšina kariet poskytuje možnosti pre viacaplikačné využitie, avšak závisí hlavne na poskytovateľoch služieb a ich vzájomnej dohode a konfigurácii ich systémov, či takéto riešenie bude možné. V prípade dohody profituje hlavne používateľ, keďže sa minimalizuje množstvo kariet, ktoré musí mať pri sebe.[3]

Personalizácia karty umožnuje v prípade jej straty alebo krádeže, túto kartu v systéme zablokovať a vydať novú, obsahujúcu rovnaké údaje a poprípade zvyšný kredit a starú označiť ako neplatnú a tým znemožnit jej zneužívanie.[3]

2.10 Čítačka

Čítačka je zariadenie, ktoré slúži pre sprístupnenie rozhrania karty do počítača alebo iného zariadenia, ktoré komunikuje s kartou — hostiteľovi. Čítačky aj napriek ich názvu umožňujú obojsmernú komunikáciu a teda dokážu karty aj zapisovať. Úlohou čítačky je poskytnúť karte zdroj napájania, zdroj synchronizačného signálu, hositeľovi signalizovať prítomnosť alebo neprítomnosť karty a v neposlednom rade tvoriť rozhranie pre ich vzájomnú komunikáciu. Pre komunikáciu s užívateľom systému sa do čítačiek obvykle osadzujú optické a akustické signalizačné prvky, obvykle malý piezoelektrický reproduktor a červená a zelená dióda, ktoré upozorňujú užívateľa o stave transakcie s kartou. Bývajú voľne programovateľné a preto je ich využitie špecifické pre konkrétnu aplikáciu.[4]

Obr. 2.5: Čítačka Advanced Card Solutions ACR128. Autor: ACS Ltd.

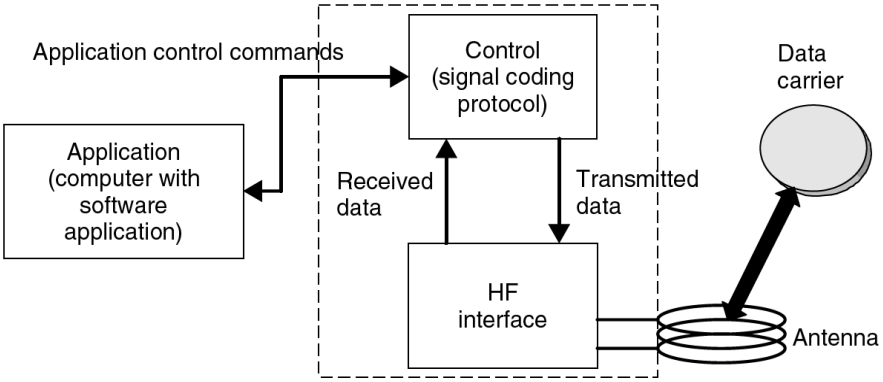
2.10.1 Konštrukcia čítačky

Čítačka sa skladá z bloku komunikujúceho s hostiteľom, ktorý je zväčša tvorený mikroprocesorom. Tento dalej ovláda aj prípadné periférie, ako napríklad klávesnicu a display pre zadávanie PIN kódu, diódy, akustický menič, poprípade digitálne výstupy na pripojenie

externých periférii. Niektoré čítačky vhodné pre vysoko bezpečné aplikácie sú vybavené ďalším konektorom pre kontaktnú kartu, ktorý je neprístupný z vonkajšku a slúži pre overenie autenticity čítačky. Pre komunikáciu s bezkontaktným transpondérom slúži rádiový blok, ktorý je tvorený zväčša špecializovanými obvodmi riadiacimi linkový protokol a zosilňovačmi pre vytovrenie samotného vysielaného a príjmaného signálu. Nemenej dôležitou súčasťou čítačky je anténa, ktorá musí mať optimálne parametre pre čo najlepší prenos signálu medzi transpondérom a čítačkou.[4]

Pre komunikáciu s hostiteľom aplikácie slúži rozhranie čítačky. Prístupové systémy a priemyselné čítačky budú pravdepodobne využívať priemyselné zbernice ako napríklad RS 485, kým čítačky určené pre pripojenie k osobnému počítaču budú skôr vybavené zbernicou RS 232 alebo modernou USB. V prostredí zbernice USB je definovaná trieda zariadení `CCID`, ktoré slúžia ako rozhranie s čipovými kartami a komunikujú s ovládačom čítačky podľa rovnomenného protokolu[13].[4]

Čítačky sa vyrábajú v rôznych vyhotoveniach v závislosti od oblasti ich použitia, čo ovplyvňuje aj množstvo periférii, ktorými sú vybavené. Pre výrobu vstavaných systémov budú najvhodnejšie plošné spoje, ktoré je možno zabudovať do krabice zariadenia, kým pre použitie v prístupových systémoch je často potrebné použiť vodeodolné puzdro pre vonkajšie použitie a možnosť montáže na stenu. Ďalším bežným vyhotovením je obvod umiestnený v plastovej krabičke pre využitie s osobným počítačom.[4]



Obr. 2.6: Bloková schéma bezkontaktnej čítačky.[4]

2.10.2 Rozhranie USB

V osobných počítačoch je rozšírená univerzálna periférna zbernica USB — *Universal Serial Bus*. Výrobcovia čítačiek pre karty ju využívajú pri konštrukcii čítačiek používaných spolu s osobnými počítačmi, pretože poskytuje viaceré výhody. Hlavnou výhodou je, že obsahuje napájacie napätie, ktoré je schopné pokryť spotrebu čítačky a tým pádom nie je potrebný žiadny ďalší napájací zdroj. Toto výrazne zjednodušuje konštrukciu a znižuje výrobné náklady.

USB zariadenia pracujúce s kontaktnými a bezkontaktnými čipovými kartami sa snaží zjednotiť trieda zariadení CCID – *(integrated) Circuit Card Interface Device* – s identifikátorom `0x0B`. Táto trieda definuje rovnomenný protokol pre komunikáciu s čítačkami.

Základom komunikácie po zbernici USB sú tzv. *endpointy*. Sú to ukončenia virtuálnych spojení medzi hostiteľom a zariadením. Trieda CCID definuje, že čítačka kariet musí podpo-

rovať 2 endpointy (typu BULK IN a BULK OUT) pre príkazy. Tretí, slúžiaci pre notifikáciu zmeny stavu (typu INTERRUPT IN) čítačky, je voliteľný.

2.10.3 Protokol CCID

Popis triedy zariadení CCID popisuje aj protokol, ktorým prebieha komunikácia medzi kompatibilnou čítačkou a obslužným softwarom, bežiacim na hostiteľskom počítači. Protokol je založený na jednoduchých transakciách, ktoré z pravidla vyvoláva hostiteľ. Typy správ sú označené predponou, ktorá určuje ktorým smerom bude prebiehať prenos správy. `PC_to_RDR` značí prenos do čítačky a príkaz musí byť zaslaný cez endpoint typu BULK OUT. Odpovede sú označené predponou `RDR_to_PC` a su príjmané pomocou endpointu BULK IN.

Offset:	0	1	2	3	4	5	6	7	8	9	10	...
Význam:	Typ správy	Veľkosť dát				Číslo slotu	Sekv. číslo	Param.			Dáta	

Tabuľka 2.1: Formát rámca protokolu CCID.[13]

Hlavičku protokolu CCID tvorí 10 bajtov, za ktorou voliteľne nasledujú dáta. Položka "Typ správy" definuje operáciu, ktorú požadujeme od čítačky. Veľkosť dát je uvedená ako druhá položka hlavičky ako 32bitové číslo bez znamienka vo formáte big-endian. Tretia položka špecifikuje rozhranie, s ktorým má byť operácia uskutočnená, v prípade čítačiek s viacerými rozhraniami. Sekvenčné číslo slúži na spárovanie požiadavky a odpovede, keďže odpovedné rámce musia mať toto číslo rovnaké. Nasledujúcou položkou sú parametre, ktoré pozmeňujú správanie niektorých príkazov a sú pre každý príkaz špecifické. [13]

Požiadavky	
PC_to_RDR_GetSlotStatus	Zistenie stavu rozhrania a prítomnosti karty
PC_to_RDR_Abort	Zrušenie už prebiehajúcej transakcie
PC_to_RDR_Escape	Dáta určené pre mikroprocesor čítačky
PC_to_RDR_XfrBlock	Dátový blok pre kartu
PC_to_RDR_Mechanical	Ovládanie mechanických zábran vybratia karty

Odpovede	
RDR_to_PC_DataBlock	Dátový blok od karty
RDR_to_PC_SlotStatus	Stav karty a čítačky
RDR_to_PC_Escape	Odpoveď od mikroprocesora čítačky

Tabuľka 2.2: Vybrané typy správ protokolu CCID.[13]

2.11 Mifare

Mifare je ochranná známka spoločnosti NXP Semiconductors, ktorá vyrába obvody pre bezkontaktné čipové karty založené na proprietárej aj štandardnej[10] technológii. Jedná sa o najrozšírenejšiu technológiu používanú v odvetví bezkontaktných kariet. Spoločnosť NXP predala viac ako 1 miliardu obvodov pre karty a viac ako 10 miliónov obvodov pre

[10]Podľa štandardu `ISO 14443-4`

čítačky. Produktovú radu Mifare tvorí viac rád kariet, ktoré sa líšia veľkosťou pamäte a aj zabezpečením prístupu k uloženým informáciam.[18]

2.11.1 Pamäťové karty Mifare

Jednými z prvých bezkontaktných čipových kariet boli karty pamäťové. Obsahujú pamäť pre údaje a jednoduchú logiku overujúcu prístupové práva. Táto konštrukcia už v dávnejšie umožnila vytvoriť čip schopný napájania v obmedzených podmienkach indukčného prenosu energie. Pamäťové karty majú len obmedzené kryptografické schopnosti a neumožnujú vykonávať programy, preto sa hodia len pre jednoduchšie aplikácie. [18]

Mifare Ultralight je najjednoduchšia a najlacnejšia karta. Neobsahuje žiadne bezpečnostné prvky a je vhodná na použitie pre jednorázové alebo krátkodobé vstupenky a menej bezpečné aplikácie. Čip je možné umiestniť aj na papierový podklad a tým ďalej znížiť cenu za kus.[18]

Mifare Classic je najstaršia produktová rada kariet Mifare. Vyrába sa v kapacitách od 320 bajtov do 4KiB, najznámejšia je však 1KiB verzia. Využíva proprietárnu technológiu, ktorá je podobná ISO/IEC 14443, avšak využíva proprietárny šifrovací protokol. Široké nasadenie tejto technológie z nej však spravilo nepísaný štandard.[18]

Mifare Plus vznikla ako náhrada kariet Mifare Classic po prelomení nedostatočného ochranného algoritmu. Podporuje algoritmus AES a teda vyžaduje čítačky schopné s ním pracovať. Zvyšné vlastnosti sú zhodné s Mifare Classic.[18]

2.11.2 Procesorové karty Mifare

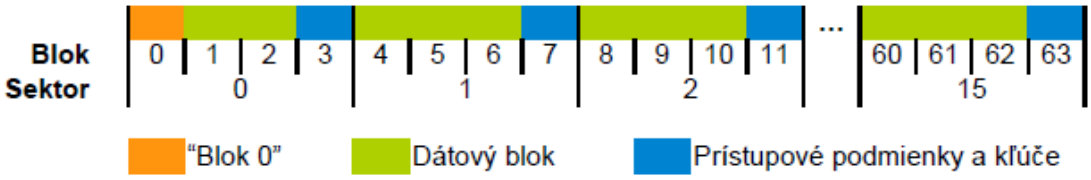
V súčastnosti sa čoraz častejšie siaha po mikroprocesorových kartách, ktoré poskytujú vyšší stupeň zabezpečenia prenosu a uložených informácii. Spoločnosť NXP Semiconductors vyrába mikroprocesorové obvody založené na plnej kompatibilite s štandaradrdom ISO/IEC 14443-4.[18]

Mifare DESFire je karta založená na mikroprocesore 8051 a obsahuje kryptografické akcelerátory pre algoritmy DES a AES a 2, 4, alebo 8 KiB pamäte EEPROM. Karta sa dodáva s predprogramovaným operačným systémom (DESFire OS), ktorý poskytuje jednoduchý súborový systém a zabezpečenie prístupu. Poskytuje vysoký stupeň zabezpečenia oproti Mifare Classic.[18]

Mifare ProX a SmartMX je najvyššia rada kariet. Poskytujú kontaktné aj bezkontaktné rozhranie. Dodávajú sa bez predprogramovaného operačného systému a umožnujú beh komerčných aj otvorených OS ako napríklad Java Card OS. Podpora širokého spektra bezpečnostných funkcií, kryptografických algoritmov a vysoká odolnosť proti úpravám predurčuje kartu na vysoko zabezpečené aplikácie.[18]

2.11.3 Organizácia pamäte karty Mifare Classic 1k

Pamäť kariet Mifare Classic 1k je rozdelená do 16tich sektorov, ktoré obsahujú 4 bloky po 16 bajtov. Využiteľná kapacita pamäte je 752 bajtov, zvyšných 272 bajtov je použitých pre uloženie výrobného čísla a prístupových podmienok pre jednotlivé sektory.[17]



Obr. 2.7: Mapa organizácie pamäte karty Mifare Classic[17]

Bloky je možné využiť buď v dátovom režime, kde je možné využiť operácie čítania a zápisu bloku a je možné využiť celých 16 bajtov. Kontrolu integrity hodnoty v prípade poškodenia si však musí aplikácia zaručiť vo vlastnej réžii. Druhou možnosťou je režim hodnoty, kedy je v bloku uložené 4 bajtové číslo so znamienkom a je možné využiť aj operácie inkrementácie a dekrementácie tejto hodnoty. Hodnota je uložená v jednom bloku dohromady tri razy, a to 2-krát normálne a raz binárne inverzne, z dôvodu kontroly chyby. Vo zvyšných 4 bajtoch je možné uložiť adresu bloku obsahujúceho zálohu aktuálneho bloku. Inicializovať blok do režimu hodnoty je možné len pomocou zápisovej operácie v správnom formáte. [17]

“Blok 0”

Prvý blok pamäte karty (blok 0 sektoru 0), označovaný aj skrátene “*Block 0*”, je vyhradený pre výrobcu čipu. Jeho obsah je neprepisovateľný a prvé 4 bajty obsahujú unikátne výrobné číslo karty, piaty bajt obsahuje kontrolný súčet výrobného čísla a zvyšných 11 bajtov obsahuje dáta určené výrobcom. Výrobné číslo sa využíva pre výber aktívnej karty v energetickom poli čítačky a je možné ho prečítať aj bez znalosti prístupového kľúča pre sektor 0. [17]

Prístupové podmienky pre sektor

Posledný blok sektoru je vyhradený pre uloženie prístupových podmienok a kľúčov pre prístup ku danému sektoru. Blok obsahuje 2x 6bajtový kľúč A a B a bitovú mapu, ktorá nastavuje privilégia pre operácie autentizované jedným z kľúčov. V prípade, poškodenia formátu tohoto sektoru nie je možné vykonať autentizáciu s týmto sektorom a je viac nepoužiteľný, preto treba pri zmene kľúčov a prístupových podmienok dbať na správny formát. Pri čítaní tohoto bloku aj za použitia správneho kľúča nie je možné uložené kľúče spätne prečítať. Pri výrobe sú oba kľúče nastavené na hodnotu `FF:FF:FF:FF:FF:FF`, a je v záujme prevádzkovateľa aplikácie aby tento blok náležite upravil pre ochranu ním uložených údajov.[17]

Pomocou nastavenia prístupových podmienok je možné obmedziť operácie, ktoré je možné vykonať s daným blokom pri použití jedného z kľúčov. Prístupové podmienky sú predpripravené tak, že sa predpokladá použitie kľúča B ako bezpečnejšieho pre závažnejšie operácie, ako napríklad zápis sektora alebo inkrementácia hodnoty, kým kľúč A sa používa na menej kritické operácie čítania a dekrementácie v prípade hodnotových blokov.[17]

2.11.4 Antikolízny mechanizmus a autentizácia

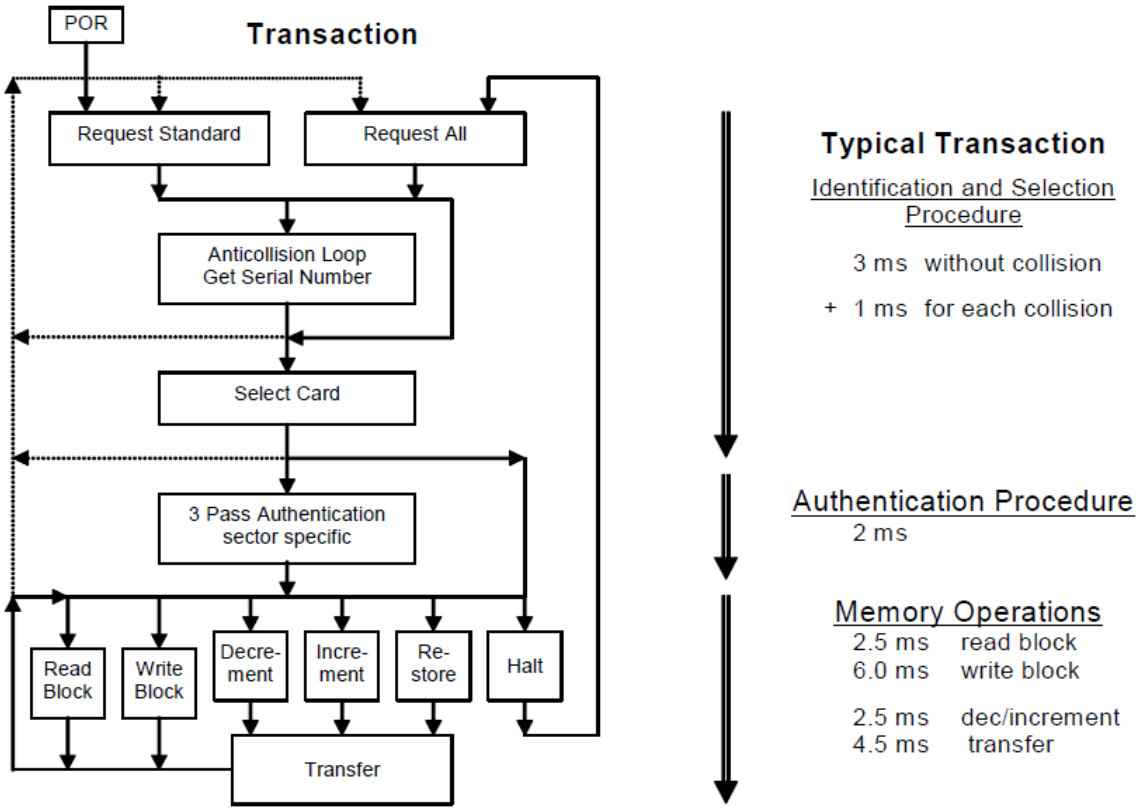
Pred samotnou autentizáciou pre prístup k sektoru prebieha antikolízny mechanizmus, ktorý slúži na výber aktívnej karty, s ktorou bude prebiehať komunikácia, pretože v aktívnej zóne

čítačky sa kariet môže nachádzať viac. Čítačka vyšle požiadavok na zistenie kariet v dosahu. Karty na túto výzvu odpovedajú odoslaním ich výrobného čísla, podľa ktorého sú následne vyzvané k aktivácii. Na výzvu zvolenia karta odpovedá správou, ktorá identifikuje jej typ. Nezvolené karty sa vracajú do stavu čakania na výber aktívnej karty.[17]

Po výbere aktívnej karty je potrebné overiť prístupové práva ku sektoru, s ktorým aplikácia hodlá pracovať. Autentizácia prebieha pomocou 3-cestného algoritmu. [17]

1. Čítačka karte oznámi sektor, s ktorým chce pracovať a ktorý z kľúčov chce použiť pre autentizáciu.
2. Karta vygeneruje náhodné číslo, ktoré zašle ako výzvu čítačke.
3. Čítačka pomocou získanej výzvy a šifrovacieho kľúča vygeneruje odpoveď a zašle ju spolu so svojou výzvou karte.
4. Karta podľa uloženého kľúča overí odpoveď od čítačky a vygeneruje odpoveď pre čítačku, ktorú jej zašle.
5. Čítačka porovná odpoveď a výzvu.

Komunikácia medzi kartou a čítačkou prebieha zašifrovane od kroku 2. Autentifikácia je platná pre všetky bloky sektoru pre každý nasledujúci príkaz. V prípade práce s iným sektorom je potrebné autentizáciu zopakovať, aj ak by sektory používali rovnaké kľúče.[17]



Obr. 2.8: Priebeh transakcie s kartou Mifare Classic a trvanie jednotlivých úsekov.[17]

2.11.5 Bezpečnostné problémy kariet Mifare Classic

Karty Mfare Classic využívajú pre ochránenie komunikácie s čítačkou proprietárny prúdový šifrovací algoritmus `CRYPTO1`[17][14]. Skupina vedcov z Radboudovej Univerzity v holandskom meste Nijmegen však objavila, že daný algoritmus je veľmi slabý a je možné ho prelomiť vo veľmi krátkom čase, pričom postačuje malý úsek zaznamenanej komunikácie medzi kartou a čítačkou, čo znižuje úroveň zabezpečenia týmto algoritmom na takmer nulovú úroveň[14].

Tieto karty sú stále vo veľkej miere používané a teda je potrebné brať do úvahy riziko ich zneužitia. Sú známe útoky na systémy platby cestovného v Londýne kde karty Mifare Classic boli využívané ako elektronická peňaženka. Prelomením zabezpečenia bolo možné na karte modifikovať zostatok[15].

Ako možné riešenia tohoto nedostatku je možné použiť viaceré postupy, avšak tieto vyžadujú daleko väčšiu réžiu na čítacích staniciach a úplne nezabraňujú zneužitiu individuálnych kariet. Jedným z riešení je prepojiť všetky terminálové stanice on-line[11] a kredit užívateľa uschovávať v informačnom systéme, čo zabráni jeho manipulácii, ale nezabráni zneužitiu pomocou emulácie karty. Druhá možnosť počíta s použitím separátneho šifrovacieho kľúča pre každú kartu. Manipulácia s kreditom však nie je úplne vylúčená, avšak obmedzuje rozsah útoku[12].

Jediné úplné riešenie tohoto problému je použitie kartiet s vyšším stupňom zabezpečenia, napríklad Mifare DESFire, ktoré sú však drahšie, avšak umožnujú použiť systém elektronickej peňaženky aj v prostredí off-line terminálov. Toto riešenie zvolila spoločnosť České dráhy a.s pri implementácii elektronickej peňaženky a cestovného dokladu "*In Karta*"[20].

[11] Neustále prepojené pomocou dátového spojenia. Tento spôsob je využívaný v rámci menz VUT.
[12] Využitie zariadenia, ktoré sa správa ako karta a je pomocou neho možné podvrhnúť akékoľvek dáta.

Kapitola 3

Autentizácia v OS Linux

Pod pojmom autentizácia rozumieme proces, pri ktorom sa overuje, či daný užívateľ je skutočne tým, za ktorého sa vydáva. Možností ako overiť identitu[1] existuje viac a ich výber pre konkrétne použitie závisí hlavne od úrovne bezpečnosti, ktorú pre daný systém požadujeme. Môže sa jednať napríklad o niečo, čo človek vie (heslo), niečo, čo človek vlastní (karta, *token*[2]), alebo merateľná vlastnosť daného človeka (biometria).[6]

S autentizáciou súvisí pojem autorizácie. Autorizácia je postup, pri ktorom je užívateľovi umožnené vykonávať činnosti, na ktoré má právo.

OS Linux je viacužívateľský operačný systém, ktorý pochádza z rodiny UNIX a teda je potrebné obmedziť práva jednotlivým užívateľom. Pre základnú autentizáciu — prihlasovanie — v OS Linux sa využíva autentizačný systém PAM (3.1), ktorý vďaka svojej modularite je schopný autentizovať pomocou širokej škály autentizačných mechanizmov. Avšak pri práci na počítači je väčšinou nekomfortné ukončiť celú reláciu pouzívateľa pri krátkych opusteniach pracoviska. Vtedy je vhodnejšie uzamknúť *plochu* a po návrate umožniť pokračovanie v práci.

3.1 Systém PAM

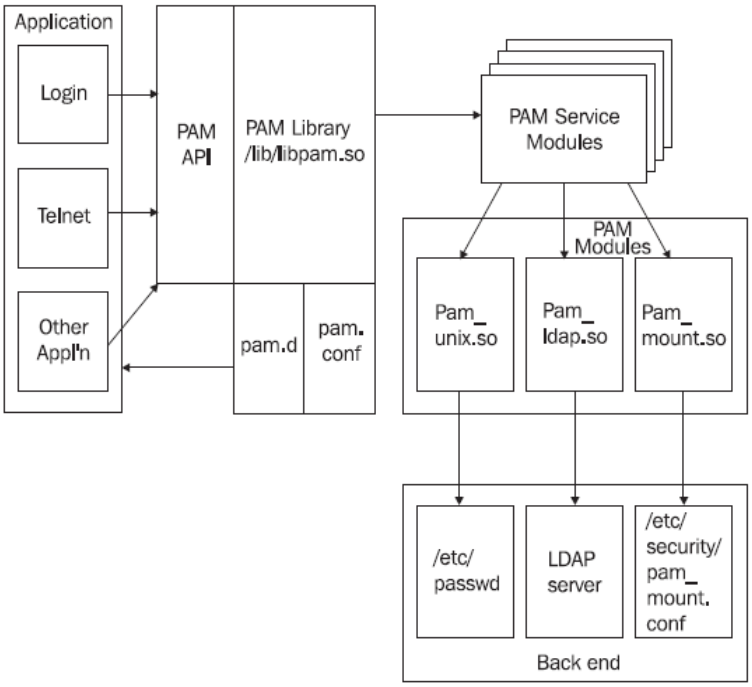
PAM (Pluggable Authentication Modules) je systém, ktorý umožnuje integrovať rôzne autentizačné mechanizmy do zložitejších systémov a tak napríklad umožniť viacfaktorovú autentizáciu, a/alebo použitie biometrických senzorov, čipových kariet a podobne.

Tento systém vznikol ako projekt spoločnosti Sun Microsystems v roku 1995 a časom bol adaptovaný do väčšiny používaných systémov založených na `UNIX`e. Potreba takéhoto systému vznikla z dôvodu, že nebolo možné jednoducho a efektívne centrálne spravovať väčší počet užívateľov pomocou klasických súborov so zoznamom používateľov. Systém PAM umožnuje použiť moduly, ktoré napríklad spolupracuju s databázami hesiel uložených centrálne (napr. modul `pam_ldap.so`). Úlohou systému PAM je vyžiadať od užívateľa potrebné údaje k autentizácii, overiť ich a následne užívateľovi pripraviť pracovné prostredie pre sedenie. [5]

Systém PAM je rozdelený na servisné moduly `auth`, `session`, `account` a `password`. Každý servisný modul sa stará o časť autentizačného procesu. Úlohou modulu `auth` je zistiť a overiť identitu užívateľa. Modul `session` sa stará o vytvorenie sedenia pre užívateľa a prípravy prostredia pre prácu. V moduli `account` sa rozhoduje o autorizácii užívateľa pou-

[1] zhodnosť, totožnosť, rovnakosť

[2] Hardwarové zariadenie, ktorého vlastníctvom preukážeme našu identitu.



Obr. 3.1: Architektúra systému PAM. [5]

žívať danú službu a kontrolujú vlastnosti užívateľského účtu. Modul `password` sa používa pri aktualizácii prihlasovacích údajov.[5]

3.1.1 Konfigurácia systému PAM

Konfigurácia systému PAM je uložená v buď v súbore `/etc/pam.conf` alebo v súboroch, ktoré sú uložené v adresári `/etc/pam.d/`. V druhom prípade je konfiguračný súbor `pam.conf` ignorovaný. V adresári `pam.d` sa nachádzajú súbory s konfiguráciou systému PAM pre každú službu osobitne. Meno služby, pod ktorým sa aplikácia predstaví systému PAM si aplikácia môže zvoliť a špecifikuje ho pri volaní funkcie `pam_start()`. Toto meno býva obyčajne napevno uložené v aplikácii. V niektorých prípadoch by však mohlo byť výhodné mať možnosť využiť rôzne autentizačné princípy pre dve instancie rovnakej aplikácie.[5]

Východzím a rezervovaným názvom konfigurácie pre PAM je `other`. Ten slúži pre aplikácie, ktoré sa predstavia neplatným menom a obvykle býva nakonfigurovaný na znemožnenie prístupu.[5]

Prvý stĺpec konfiguračného súboru (príklad konfigurarčného súboru nájdete v prílohe C) špecifikuje, ktorého servisného modulu sa konfigurácia týka. Druhý stĺpec obsahuje kľúčové slovo, ktoré volí úroveň dôležitosti modulu. K dispozícii sú tieto úrovne:

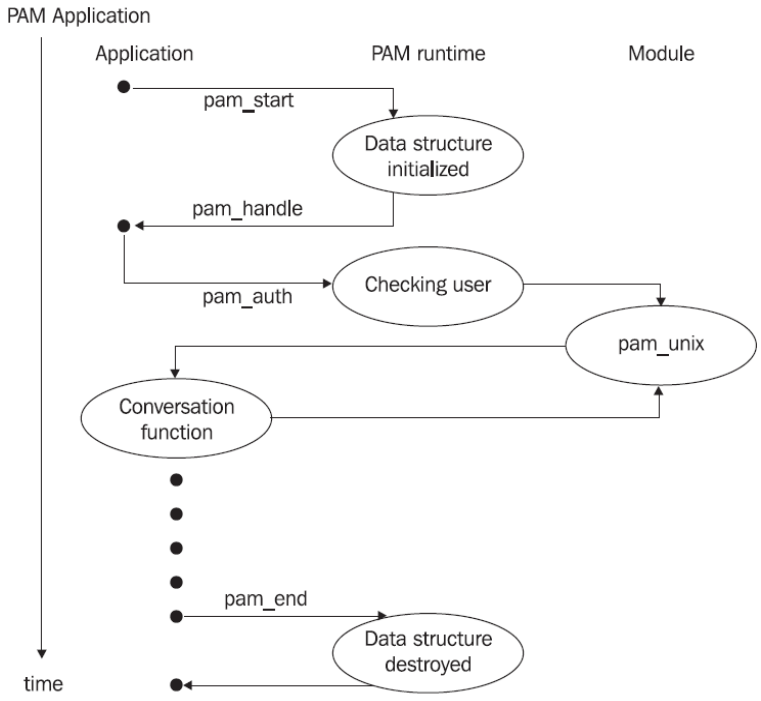
- **requisite** – najvyššia úroveň, v prípade neúspechu je okamžite autentizácia ukončená.
- **required** – v prípade neúspechu autentizácie takto označeným modulom sa pokračuje vo vykonaní ostatných modulov a na koniec je ohlásený neúspech.
- **sufficient** – ak pred úspešným skončením takto označeného modulu nedošlo ku chybe žiadneho modulu označeného `required` je autentizácia ukončená s úspechom. Zvyšné moduly sa nevykonajú.

- **optional** – modul je vykonaný a na jeho výsledok nie je braný ohľad.
- **include** – v mieste tohoto kľúčového slova sú pripojené pravidlá zo súboru nasledovaného ako argument.

V treťom stĺpci je uvedený modul, ktorý sa má použiť a za ním nasledujú argumenty pre modul samotný, ktoré systém PAM odovzdá funkcii modulu. Moduly sú vykonávané v poradí, ktoré je uvedené v konfiguračnom súbore. Ich vhodným usporiadaním je možné vytvoriť pokročilé autentizačné schémy.[5]

3.1.2 Priebeh autentizácie

Hlavnou úlohou systému PAM je autentizovať užívateľov pri prihlasovaní do systému. Programy, ktoré slúžia na prihlásenie (`login` v prípade textového terminálu, alebo napr. `xdm` v prípade terminálu grafického) musia byť skompilované s podporou `libpam.so`. Táto knižnica sa stará o prečítanie konfigurácie a spustenie príslušných servisných modulov. Servisné moduly zavolajú pre svoju činnosť moduly systému PAM, ktoré vykonajú samotnú autentizáciu. [5]



Obr. 3.2: Priebeh autentizácie v rámci aplikácie spolupracujúcej s PAM. [5]

Aby boli moduly PAM schopné komunikovať s užívateľom, ktorý sa práve autentizuje, musí aplikácia využívajúca PAM poskytnúť pri vytvorení PAM sedenia ukazateľ na tzv. *konverzačnú funkciu*. Táto funkcia sprostredkováva hlášky od modulu užívateľovi a späť odovzdáva odpovede užívateľa. Ich zobrazenie a načítanie odpovedí je úlohou aplikácie využívajúcej autentizáciu. Toto riešenie umožnuje vykonať dialóg s užívateľom v rámci diametrálne odlišných aplikácií bez modifikácie modulov PAM. Po úspešnom overení identity užívateľa, by mal mať pripravené kompletné pracovné prostredie.[5]

3.1.3 Moduly PAM

Moduly slúžia pre vykonanie samotnej autentizačnej činnosti. Sú tvorené dynamickými knižnicami, so zhodným API a môžu poskytovať jednu alebo viac služieb servisných modulov. Majú k dispozícii prostredie a oprávnenia aplikácie, v ktorej prebieha autentizácia. Niekedy sa preto musia spoliehať na externé aplikácie, ktoré sú schopné behu s vyššími oprávneniami (`setuid` bit).[5]

3.2 Uzamykanie aktuálnej relácie užívateľa

Uzamykanie relácie slúži pre ochránenie aktuálneho sedenia užívateľa po dobu kratších opustení terminálu. V grafickom rozhraní "*X Window*" pre OS Linux existuje množstvo hotových riešení pre uzamykanie relácie. Hlavné odlišnosti sú v grafickom vyhotovení a vstavaných efektoch pre šetrič obrazovky. Tieto aplikácie bývajú prezývané "šetrič obrazovky", avšak pri použití moderných LCD monitorov ku šetreniu obrazovky nedochádza.[22]

Programy poskytujúce uzamknutie aktuálneho sedenia prekryjú pracovnú plochu užívateľa tak, aby nebolo možné vidieť spustené aplikácie a dáta v nich, a odchytávajú udalosti od klávesnice a myši, aby nebolo možné nič modifikovať. Aby bolo možné počítač znova využívať, je potrebné sa autentizovať. Na overenie autenticity využívajú šetriče obrazovky zväčša systém PAM.[22]

Kapitola 4

Návrh autentizačného systému pre OS Linux

4.1 Personalizácia kariet

Vzhľadom na nedokonalosti kariet Mifare Classic popísané v sekcii 2.11.5 karty nebudem personalizovať zápisom údajov na kartu, ale budem využívať len unikátne výrobné číslo karty, ktoré bude zviazané s užívateľom na úrovni aplikácie. Výrobné číslo je síce možno prečítať aj bez autentizácie s kartou, avšak toto číslo je v karte uložené v neprepisovateľnej pamäti. Pre zneužitie by útočník musel vlastniť emulátor karty, čo je v prípade zamýšlanéj cieľovej skupiny použitia aplikácie sa javí ako veľmi nepravdepodobné. Z bezpečnostného hľadiska to bude mať len minimálny dopad. Zároveň to umožní využívať karty, ktoré neboli primárne určené pre použitie v tejto aplikácii a tým pádom zamedzí možným konfliktom s inými systémami. Zjednoduší sa taktiež návrh modulu PAM, pretože bude možné použiť databázu čitateľnú pre všetkých užívateľov a tým pádom odpadá medzivrstva potrebná na zvýšenie užívateľských práv pre čítanie databázy.

4.2 Prihlasovanie do OS

Pre úvodné prihlásenie k systému bude potrebné vytvoriť modul systému PAM, ktorý poslúži na overenie identity osoby pomocou preukázania sa bezkontaktnou kartou technológie Mifare. Bude potrebné implementovať obslužnú funkciu pre servisný modul `auth` systému PAM, ktorá bude vykonávať túto autentizáciu.

Modul bude používať konfiguráciu a databázu uloženú v súboroch. Pri autentizácii bude modul priamo komunikovať s čítačkou. Keďže sa toto riešenie bude primárne využívať na pracovnej stanici, bude čítačku využívať len jeden užívateľ a to takmer výlučne v jednej aplikácii, preto nieje potrebné sa zaoberať viacnásobným prístupom. Na základe konfigurácie systému PAM bude možné modul využívať ako hlavný aj dodatočný prvok autentizácie.

4.3 Uzamykanie obrazovky

Pre uzamykanie sedenia v prostredí OS Linux bude potrebné vytvoriť aplikáciu, ktorá bude zisťovať prítomnosť karty v poli čítačky. Pre autentizáciu aplikáciou na uzamykanie sedenia by bolo možné využiť moduly systému PAM, keďže väčšina týchto aplikácii autentizuje užívateľov práve týmto spôsobom, avšak opätovné uzamknutie sedenia nie je realizovateľné

systémom PAM a vyžaduje externú službu, ktorá zisťuje prítomnosť karty na čítačke a spustí uzamykaciu aplikáciu v prípade odstránenia karty.

V desktopovom prostredí OS Linux existuje veľké množstvo aplikácii pre ochránenie aktuálneho sedenia. Výber jedného konkrétneho by nemusel vyhovovať všetkým užívateľom, preto by bolo vhodné umožniť zvolenie príkazov na uzamknutie a odomknutie aktuálneho sedenia. Toto riešenie umožnuje využiť aplikáciu aj iným, ako pôvodne zamýšlaným spôsobom[1]. Použitie existujúcich uzamykacích aplikácii umožní použiť pôvodný spôsob odomknutia ako záložné riešenie pre prípad, že užívateľ stratí, alebo si zabudne kartu.

Aby bolo možné využiť túto aplikáciu aj univerzálnejšie, bolo by vhodné umožniť aspoň základné rozdelenie právomocí pre jednotlivé karty. Toto umožní použiť túto aplikáciu samostatne, napríklad ako spoločný terminál v serverovni, kde jednotliví užívatelia budú mať možnosť odomknúť terminál, ale len vybraní správcovia získajú možnosť pridať daľších užívateľov.

[1] Napríklad ovládanie osvetlenia alebo hudobného prehrávača.

Kapitola 5

Implementácia systému

Implementáciu som rozdelil na 3 dielčie úlohy. Ako prvé bolo potrebné vytvoriť ovládač pre čítačku ACR128 aby ju bolo možné použiť v prostredí OS Linux. Pri testovaní ovládača som vytvoril jednoduchú aplikáciu, ktorá je schopná prečítať obsah karty so zadaným kľúčom. Následne som vytvoril aplikáciu pre uzamykanie aktuálneho sedenia užívateľa v grafickom režime pomocou toolkitu Qt. Ako posledné som implementoval modul systému PAM pre prihlasovanie k systému.

5.1 Ovládač `libacr128.so`

Pre účely tejto bakalárskej práce mi bola zapožičaná čítačka kariet Advanced Card Systems ACR128, ktorej výrobca deklaruje kompatibilitu s OS Linux a poskytuje ovládač pre systém PC/SC Lite[1]. Pri úvodnom testovaní čítačky sa však ukázalo, že tento ovládač nepodporuje zasielanie *escape* príkazov a tým pádom znemožnuje konfiguráciu čítačky. Preto som najskôr vytvoril ovládač pre túto čítačku, ktorý implementuje príkazy potrebné pre jej nastavenie a prácu s pamäťovými kartami Mifare.

Pre implementáciu ovládača som zvolil programovací jazyk C++. Čítačka je k počítaču pripojená pomocou rozhrania USB a pre komunikáciu s počítačom využíva protokol CCID. Pre komunikáciu s USB som využil *userspace* knižnicu `libusb-1.0` a protokol CCID som následne implementoval podľa špecifikácie. Pre prácu s kartami Mifare nebolo potrebné implementovať všetky podporované príkazy čítačky ani protokol CCID v plnom rozsahu, postačla podmnožina, ktorá ale umožnla všetky potrebné operácie.

Základ tvorí bázová trieda `reader`, ktorá implementuje základné operácie, ktoré by mala podporovať čítačka kariet. Pri návrhu som vychádzal z hlavne z použitia čítačky ACR128, takže je možné, že pri implementácii ovládača pre inú čítačku ju bude potrebné rozšíriť o funkcie, ktoré čítačka podporuje.

5.1.1 Trieda `ccid_usb`

Táto trieda slúži pre abstrakciu réžie pri komunikácii po zbernici USB. Pri inicializácii vytvorí kontext komunikácie s knižnicou `libusb`. Poskytuje metódy pre komunikáciu cez endpointy `bulk_in` a `bulk_out`. Trieda interne detekuje zariadenie a pripraví zbernicu na komunikáciu.

[1] Personal Computer/Smart Card - Rozhranie pre prácu s čipovými kartami vytvorené poprednými výrobcami kariet a softwaru

Pred začatím komunikácie je potrebné si uzamknúť čítačku pre výlučný prístup pomocou medódy `claim()` a po skončení ju opäť uvoľniť metódou `release()`. Pri pokuse o uzamknutie môže dôjsť ku kolízii s iným pristupujúcim procesom. Trieda to rieši pomocou opakovaného pokusu o uzamknutie. V prípade neúspechu vyvolá výnimku `driver_exception` informujúcu o obsadenosti zariadenia.

5.1.2 Trieda `ccid`

Komunikáciu protokolom CCID nad zbernicou USB obstaráva trieda `ccid`. Obaluje príkazy a dáta do správ a správne nastavuje hlavičky protokolu. V odpovedi na požiadavku kontroluje či patrí k akutálnej transakcii. Abstrahuje teda vybrané časti protokolu CCID, ktoré sú potrebné pre správnu činnosť ovládača a aplikácí na ňom založených. Pre svoju činnosť využíva rozhranie poskytované triedou `ccid_usb`

5.1.3 Trieda `reader`

Abstraktná bázová trieda, ktorá zjednocuje rozhranie pre komunikáciu s čítačkami pomocou mnou navrhnutého ovládača. Poskytuje možnosť rozšírenia o ovládače k iným čítačkám a ich využitie spoločne s aplikáciami podporujúcimi tento ovládač.

5.1.4 Trieda `acr128`

Implementácia konkrétnych príkazov špecifických pre čítačku ACR128 sa nachádza v tejto triede. Rozhranie poskytuje metódy pre komunikáciu s kartou Mifare a pre konfiguráciu čítačky a je implementované podľa manuálu [1]. V priebehu implementácie som zistil, že konfiguračné príkazy obsahujú "magický" prefix, ktorý nie je popísaný v dokumentácii. Pre jeho zistenie som využil postupy reverzného inžinierstva a softwarový analyzátor zbernice USB `USBTrace` od spoločnosti Sysnucleus.

Pre komunikáciu s kartami Mifare využíva čítačka ACR128 emuláciu protokolu T=CL, preto metódy tejto triedy čiastočne využívajú tento protokol. S čítačkou trieda komunikuje pomocou rozhrania, ktoré poskytuje trieda `ccid`.

5.1.5 Výnimka `driver_exception`

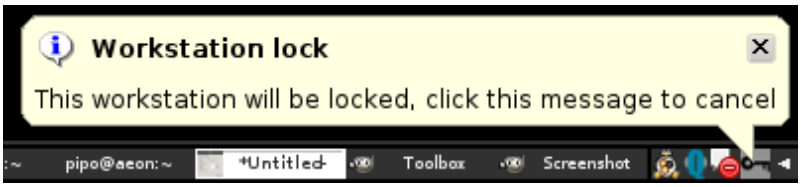
Pre oznamovanie chybových stavov, ktoré môžu vzniknúť pri komunikácii s čítačkou poprípade zbernicou USB slúži trieda výnimky `driver_exception`. Trieda poskytuje rozhranie na odovzdanie informácie o chybovom stave aplikácii a obsahuje položky identifikujúce typ a popis a závažnosť chyby.

5.2 Aplikácia `mifare1kdump`

Táto jednoduchá aplikácia vznikla ako testovací program pri implementácii ovládača. Je možné s ňou prečítať obsah pamäťových kariet Mifare Classic za po užitia východzieho, alebo zadaného šifrovacieho kľúča. Aplikácia sa postupne autentizuje s každým sektorom a prečíta jeho obsah. Pre implementáciu som zvolil programovací jazyk C++, ale nevyužil som princípy OOP. Výsup aplikácie ukázaný v prílohe D.

5.3 Aplikácia `qMifareScreenLocker`

Aplikáciu pre uzamykanie obrazovky som implementoval v programovacom jazyku C++ s pomocou grafického toolkitu `Qt 4.6`. Aplikácia využíva mnou implementovaný ovládač čítačky. Program aktívne dotazuje čítačku v nastavenom intervale (*polling*)[2] na prítomnosť karty a prípadne jej výrobné číslo. Na základe zmeny týchto skutočností a aktuálneho stavu aplikácie rozhoduje, či má dôjsť k uzamknutiu alebo odomknutiu sedenia.

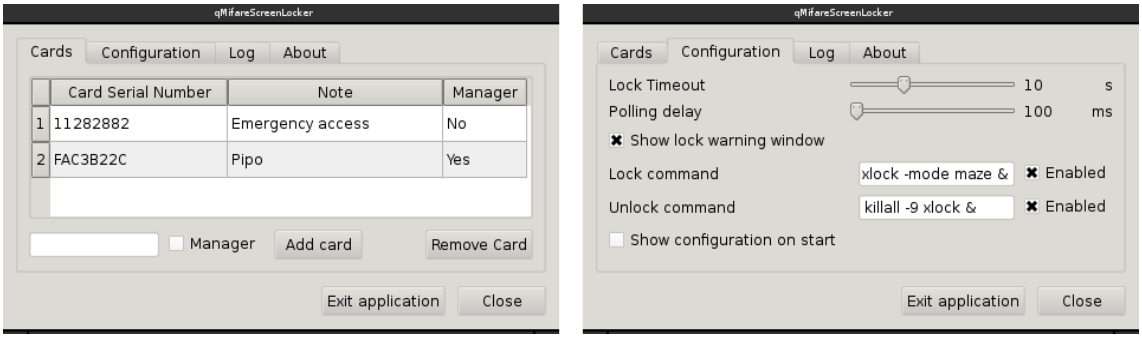


Obr. 5.1: Ikona aplikácie v oznamovacej oblasti a "balón" notifikácie.

Konfigurácia aplikácie prebieha v GUI, ktoré sa zobrazí pri prvom spustení aplikácie a vždy po kliknutí na ikonu aplikácie v systémovej oznamovacej oblasti. Po kliknutí na ikonu v systémovej lište aplikácia vyžaduje priloženie karty, ktorá je v konfigurácii označená ako "Manager". Iba vlastníci tejto karty môžu vstúpiť do konfigurácie. Pre uzamknutie a odomknutie obrazovky je potrebné definovať užívateľské príkazy, ktoré táto aplikácia vykoná na základe zmeny stavu karty. Tieto príkazy možno nastaviť v záložke "Configuration" v GUI.

5.3.1 Trieda `mainWindow`

Základná logika aplikácie a ovládacie rozhranie je implementované v triede `mainWindow`. Pre riadenie udalostí sa využíva systém signálov a slotov, ktorý poskytuje toolkit Qt. Trieda vytvára hlavné konfiguračné okno aplikácie, kde je možné spravovať autorizované karty, modifikovať príkazy vykonané v prípade zmeny stavu karty, poprípade prehliadať záznam prihlásení. V prípade normálnej činnosti je hlavné okno skryté a program sa prejavuje iba ikonou v systémovej oznamovacej oblasti.



Obr. 5.2: Správa autorizovaných kariet a konfigurácia aplikácie.

Keď je karta odstránená z čítačky, zobrazí sa v oznamovacej oblasti "balónik" s notifikáciou, že dôjde ku zamknutiu obrazovky. Užívateľ môže uzamknutie zrušiť ak klikne na

[2] Vzhľadom na spôsob implementácie zbernice USB ani iné riešenie nie je možné.

túto notifikáciu v priebehu časového intervalu, po ktorom nastane zamknutie. Tento časový interval je možné nastaviť v konfigurácii, poprípade vypnúť zobrazovanie notifikácie.

Pre pridanie karty do zoznamu slúži tlačidlo *Add Card*. Pred jeho stlačením je potrebné vyplniť poznámku a zvoliť oprávnenie karty. Pre odstránenie karty postačí v zozname zvoliť kartu a kliknúť *Remove Card*.

5.3.2 Trieda `cardMgrThread`

Riadenie udalostí a stavu čítačky implementuje trieda `cardMgrThread`. Využíva separátne vlákno, v ktorom beži časovač, ktorý pravidelne spúšťa komunikáciu s čítačkou. Pri zmene stavu čítačky bežiace vlákno notifikuje hlavné vlákno, pomocou Qt signálu. Signály sú vyvolané pri priblížení a oddialení karty od čítačky, poprípade pri zmene aktívnej karty. Parametrom signálu je výrobné číslo karty

Táto trieda poskytuje aj metódy pre ovládanie periférii čítačky. Trieda poskytuje aj možnosť jednorázového načítania sériového čísla, ktoré vyvolá iný signál a je možné ho použiť napríklad pre pridávanie kariet do databázy.

5.3.3 Ukladanie nastavení aplikácie

Pre ukladanie aktuálneho stavu konfigurácie aplikácia využíva univerzálne úložisko nastavení aplikácii, ktoré poskytuje toolkit Qt – `QSettings`. Pri ukončení apikácie je aktuálny stav uložený do úložiska poskytovaného triedou `QSettings` a pri spúšťaní aplikácie je obnovený komplentý predošlý stav.

Keďže uložené konfiguračné nastavenia sú prístupné užívateľovi, ktorý spustil aplikáciu, je potrebné pre ich ochranu využiť prostriedkov operačného systému a túto aplikáciu ideálne spusiť s právami iného užívateľa napríklad pomocou nástroja `sudo`.

5.4 Modul `pam_mifare.so`

Autentizáciu pomocou kariet Mifare pri prihlasovaní obstaráva PAM modul `pam_mifare.so`. Vytvoril som ho v jazyku C++, aby bol schopný využívať ovládač `libacr128.so`. Pri autentizácii priamo komunikuje s čítačkou.

Modul implementuje funkcie potrebné pre správnu činnosť servisného modulu `auth`. Konfigurácia modulu sa nachádza v konfiguračnom súbore systému PAM, ktorý sa postará o predanie parametrov autentizačnej funkcii. Povinným parametrom tohoto modulu je cesta k súboru, obsahujúcemu databázu autorizovaných kariet. Databáza je tvorená textovým súborom vo formáte "`login;výrobné číslo;poznámka`". Súbor musí byť čitateľný pre všetkých užívateľov, pod ktorými bežia programy vyžadujúce autentizáciu. Každá položka sa v databázovom súbore môže vyskytovať viackrát, aby bolo možné mať viac kariet pre jeden účet ako aj pristupovať jednou kartou k viacerým účtom. Pri autentizácii je kontrolovaný každý riadok na zhodu užívateľských mien a čísel karty. V prípade zhody je modul ukončený s úspechom. Parametrom konfigurácie je možné špecifikovať, čo modul spraví v prípade neúspechu komunikácie s čítačkou.

Pre pridanie autorizovanej karty do databázy slúži jednoduchý nástroj `mifareaddcard`, ktorého povinný parameter je užívateľské meno, pre koré sa má pridať oprávnenie. Program si vyžiada priloženie karty k čítačke aby mohol načítať jej výrobné číslo a následne ho zapíše do databázového súboru.

Kapitola 6

Testovanie a zhodnotenie výsledkov

6.1 Testovacie podmienky

Implementované riešenie som v priebehu vývoja využíval na svojej pracovnej stanici a to najmä na uzamykanie obrazovky v mojej neprítomnosti. Na pracovnej stanici používam distribúciu Gentoo Linux s jadrom verzie 2.6.33, GUI toolkitom Qt verzie 4.6.2 a knižnicou `libusb-1.0.1`. Najnovšia[1] verzia knižnice `libusb-1.0.6` sa v priebehu implementácie ukázala ako nefunkčná a musel som nainštalovať staršiu verziu.

Ako sekundárny testovací počítač som využil netbook Asus EEE 901, na ktorom je nainštalovaná distribúcia Ubuntu 9.10 s jadrom verzie 2.6.31, Qt toolkitom vo verzii 4.5.2 a knižnicou `libusb-1.0.0`. Oba testovacie počítače používali knižnicu `libpam-1.1.0`.

6.2 Zistenia z hľadiska komfortu

Službu uzamknutia obrazovky som aktívne využíval po celý čas, ktorý som mal zapožičanú čítačku. Pri použití na pracovnej stanici zvyšuje komfort používania tým, že automatizuje uzamknutie pracovnej stanice a s tým spojené činnosti. Výhodou je aj rýchle odomknutie aktuálneho sedenia s minimom námahy po návrate. Vďaka možnosti zadať vlastné príkazy, ktoré sa majú vykonať v prípade odňatia karty z dosahu čítačky som mohol `qMifareScreenLocker` nakonfigurovať tak, že po mojom odchode sa okrem uzamknutia obrazovky pozastavilo prehrávanie hudby z hudobného prehrávača `mpd` a nastavil skrátený interval pre prechod monitoru do úsporného režimu, čo znižovalo spotrebu a opotrebenie monitora.

Na rozdiel od pracovnej stanice, kde sú výhody z hľadiska komfortu nesporné, je riešenie nevhodné pre použitie s prenosnými počítačmi. Hlavným problémom je potreba externej čítačky, ktorú je potrebné nosiť so sebou. Použitie v teréne je takmer nemožné, keďže je potrebné mať kartu umiestnenú na čítačke. Ďalším nezanedbateľným faktorom je energetická spotreba čítačky, ktorá znižuje výdrž na baterky. Podobné riešenie by bolo v prípade prenosných počítačov použiteľné len v prípade, že bezkontaktná čítačka kariet bude integrovaná do počítača.

[1] V dobe písania tejto práce.

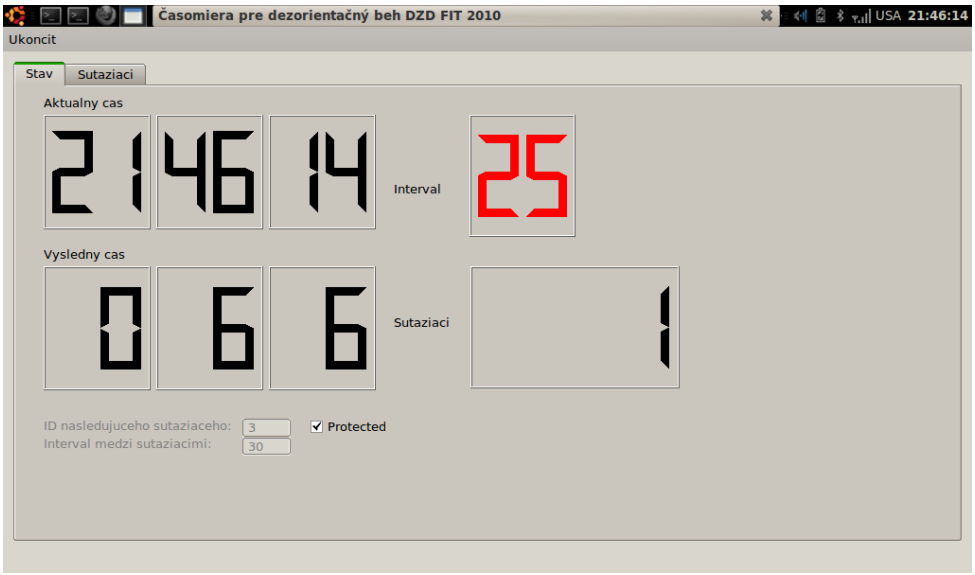
6.3 Zistenia z hľadiska bezpečnosti

Vo veľkej väčšine prípadov je úroveň bezpečnosti systému nepriamo úmerná komfortu používania. Tak tomu je aj v tomto prípade. Odomknutie relácie pomocou karty bez žiadnych ďalších autentizačných mechanizmov zvyšuje riziko kompromitácie relácie užívateľa v prípade, že sa niekto nepovolaný zmocní karty. V období testovania sa mi viackrát stalo, že som si kartu zabudol na prístupnom mieste, a tým pádom to niekto mohol zneužiť. Moja pracovná stanica však neobsahuje žiadne tajné informácie, takže riziko bolo minimálne.

V prípade potreby vyššej úrovne zabezpečenia by bolo potrebné systém vybaviť napríklad o možnosť zadania kódu, alebo iného spôsobu dodatočnej autorizácie, ktorý by nezávisel výlučne na odcudziteľnom zariadení. Pri použití ako dodatočný prvok autorizácie pri prihlasovaní do OS je však toto riešenie bezpečnejšie ako použitie hesla alebo karty samotnej a nemá takmer žiadny negatívny vplyv na komfort využívania, hlavne z dôvodu veľmi rýchlej transakcie s kartou.

6.4 Časomiera pre DZD FIT

Ovládač `libacr128.so` a triedu pre prácu s čítačkou v prostredí toolkitu Qt `cardMgrThread` som využil aj pri implementácii časomiery pre "dezorientačný beh" usporiadaný v rámci akcie "Deň zatvorených dverí" na Fakulte Informačných Technológii. Časomiera umožnila evidovať a merať časy pre väčší počet bežcov, ktorý mohli trasu absolvovať paralelne. Použitie bezkontaktných kariet skrátilo dobu potrebnú pre evidenciu súťažiacich, zaznamenanie a vyhodnotenie ich výsledkov.



Obr. 6.1: Hlavná obrazovka časomiery pre dezorientačný beh.

Časomiera pracovala s presnosťou na 1s, čo bolo obmedzené softwarovo, čítačka zisťovala prítomnoť karty každých 100ms, avšak väčšia presnosť nebola potrebná. Organizátori akcie boli s implemenáciou časomiery spokojní a pochvaľovali si jednoduchosť merania ako aj stabilitu aplikácie a zvažujú využitie tejto aplikácie aj v budúcnosti.

Kapitola 7

Záver

7.1 Zhodnotenie práce

V rámci tejto práce sa podarilo vytvoriť systém pre autentizáciu pomocou bezkontaktných čipových kariet Mifare Classic v prostredí operačného systému Linux. Implementované sú 2 riešenia slúžiace pre systémovú autentizáciu – prihlásenie – a pre uzamknutie aktuálnej rozpracovanej relácie. Práca zhrňuje taktiež najkritickejšie bezpečnostné problémy spojené s využitím bezkontaktných čipových kariet a ich možné riešenie jednoduchými prostriedkami. Použitá technológia kariet je však "preslávená" v negatívnom zmysle slova pre nedokonalosti v zabezpečení. Navrhované riešenia len zdvíhajú úroveň znalostí a vybavenia potrebného na jej prekonanie. Zvyšovanie zabezpečenia v takýchto podmienkach sa nazýva aj *security by obscurity* — bezpečnosť pomocou zahmlievania a komplikácie riešenia.

Pri testovaní ovládača pre čítačku ACR128 som zistil, že študentské karty VUT na čipe neobsahujú žiadne personalizačné informácie a všetky sektory sú zapisovateľné. Je ich teda možné využiť v iných aplikáciach, ktoré taktiež používajú karty Mifare Classic (v prílohe D je výpis pamäte študentského preukazu).

V práci je ukázané aj využitie tejto technológie a to ako časomiera, kde sa bežci identifikovali bezkontaktnými kartami. To dokazuje, že možnosti možnosti využitia tejto modernej a stále sa rozvíjajúcej technológie sú roziahle a nemusia vždy byť úplne tradičné.

7.2 Navrhované rozšírenia systému

Systém bol navrhnutý pre použitie v prostredí pracovnej stanice a preto nepodporuje centralizovanú správu autentizačných údajov. Ako vhodné rozšírenie sa teda javí možnosť vytvorenia sieťového autentizačného serveru, čo by za použitia vhodného autentizačného protokolu umožnilo využívať bezpečnostné výhody a komfort používania napríklad v prostredí školských laboratórii. Odpadli by tak napríklad starosti administrátorov so zmenou hesiel pre žiakov na základných školách, ktorí heslá často zabudnú. Karta by bola následne použitelná aj pre vydávanie obedov v jedálni poprípade ako dopravná karta.

Systém by mohol byť využitý aj na zabezpečovanie miestností po doprogramovaní vhodných fukncii, avšak bežný počítač ma príliš veľkú spotrebu energie, preto by bolo skôr vhodnejšie využiť vstavaný systém s nízkym príkonom.

Z bezpečnostného hľadiska by bolo vhodné využiť dokonalejší typ kariet, napríklad Mifare DESFire, čím by sa značne zvýšila bezpečnosť riešenia. Zaujímavým riešením by bolo aj využitie programovateľných procesorových kariet a využitie asymetrickej kryptografie

nielen pre lokálne ale aj pre vzdialené prihlasovanie, napríklad pomocou protokolu SSH, poprípade využitie pre bezpečné skladovanie šifrovacích kľúčov pre prístup k diskom, alebo podpisovanie e-mailov.

PAM modul `pam_mifare.so` by bolo vhodné rozšíriť o aplikáciu, ktorá by sa starala o overenie karty a bolo by ju možné oddeliť od samotného modulu. Potom by jej bolo možné nastaviť `setuid` bit a vykonávať ju s právami užívateľa `root`, čo by umožnilo ochrániť súbor s povolenými kartami proti čítaniu a zamedziť tak jeho prípadnému zneužitiu. Rovnako by obyčajný užívateľ nemusel mať prístup k zbernici USB z dôvodu komunikácii s čítačkou.

Aby moholi byť pomocné triedy ovládača pre čítačku použité aj s inými čítačkami triedy CCID by bolo potrebné upraviť detekciu endpointov čítačky a zaistiť automatickú detekciu USB zariadení triedy CCID. Aktuálna implementácia počíta len s využitím čítačky ACR128.

7.3 Riešenie niektorých bezpečnostných problémov

Hlavným bezpečnostným rizikom systému je použitie nedostatočne zabezpečenej technológie Mifare Classic. Pre zvýšenie úrovne zabezpečenia by bolo vhodné uložiť na kartu kontrolnú hodnotu, ktorá by bola chránená individuálnym šifrovacím kľúčom pre každú kartu. Nevýhodou tohoto prístupu je potreba mať k dispozícii šifrovacie kľúče pre každú kartu, z čoho by vyplývala potreba mať rozsiahlu databázu na každom termináli poprípade mať terminály pripojené on-line. Toto prináša so sebou ďalšie bezpečnostné riziká, ako napríklad sieťový útok alebo odcudzenie databáze, na druhú stranu silne obmedzuje rozsah útoku na technológiu Mifare Classic, keďže by bolo potrebné odchytiť komunikáciu s každou kartou separátne.

Ďalším bezpečnostným rizikom je odcudzenie karty. Rozsah tohoto druhu útoku je nižší, zväčša skôr odhalený ako klonovanie karty, avšak stále hrozí bezpečnostné riziko. Ako možné riešenie je použiť napríklad PIN[1] kód spolu s použitím karty. Na zadávanie tohoto kódu by bolo možné využiť napríklad výukovú platformu FITKit. Šifrovací kľúč by mohol napríklad byť odvodený od zadávaného PIN-kódu a tým by aj odpadla potreba mať lokálnu databázu kľúčov. Každé zvýšenie zabezpečenia je však na úkor komfortu používania systému a preto som ho neimplementoval do `qMifareScreenLocker`-u. Pri autentizácii pomocou PAM modulu je možné využiť dodatočnú autentizáciu pomocou ďalších modulov a tak zvýšiť úroveň zabezpečenia a kartu používať napríklad ako ďalší faktor autentizácie s heslom alebo napríklad biometrickým snímačom.

[1] Personal Identification Number

Literatúra

[1] Advanced Card Systems Ltd.: ACR 128U Dual-Interface Smart Card Reader: Application Programming Interface [online]. http://www.acs.com.hk/drivers/eng/API_ACR128_v1.9.pdf, 2010 [cit. 2010-05-10].

[2] EMTEST: Vybavenie cestujúcich [online]. http://www.emlines.com/sk/index.php?option=com_content&task=blogcategory&id=93&Itemid=109, 2009 [cit. 2010-04-16].

[3] EMTEST: Personalizácia [online]. http://www.emlines.com/sk/index.php?option=com_content&task=view&id=278&Itemid=110, 2009 [cit. 2010-05-04].

[4] Finkenzeller, K.: *RFID Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification*. John Wiley and Sons, Ltd, 2003, iSBN 0-470-84402-7.

[5] Geisshirt, K.: *Pluggable authentication modules*. Packt Publishing, 2006, iSBN 978-1-904811-32-9.

[6] Huntington, G.: The Business of Authentication [online]. http://www.authenticationworld.com/, 2009-06-27 [cit. 2010-04-16].

[7] ICAO: *Machine Readable Travel Documents. Part 3, Machine Readable Official Travel Documents*. International Civil Aviation Organization, 2008, iSBN 978-92-9231-140-7.

[8] ISO 14443-1:2008: *Identification cards – Contactless integrated circuit(s) cards – Proximity cards – Part 1: Physical characteristics*. Geneva, Switzerland: ISO, 2008. 4 s.

[9] ISO 14443-2:2001: *Identification cards – Contactless integrated circuit(s) cards – Proximity cards – Part 2: Radio frequency power and signal interface*. Geneva, Switzerland: ISO, 2001. 11 s.

[10] ISO 14443-3:2001: *Identification cards – Contactless integrated circuit(s) cards – Proximity cards – Part 3: Initialization and anticollision*. Geneva, Switzerland: ISO, 2001. 48 s.

[11] ISO 14443-4:2008: *Identification cards – Contactless integrated circuit cards – Proximity cards – Part 4: Transmission protocol*. Geneva, Switzerland: ISO, 2008. 37 s.

[12] ISO 7810:2003: *Identification cards – Physical characteristics*. Geneva, Switzerland: ISO, 2003. 11 s.

[13] Kolektív autorov: Universal Serial Bus - Device Class: Smart Card – CCID, rev:1.1 [online]. http://www.usb.org/developers/devclass_docs/DWG_Smart-Card_CCID_Rev110.pdf, 22-04-2005 [cit. 2010-05-06].

[14] de Koning Gans, G.; Hoepman, J.-H.; Garcia, F. D.: A Practical Attack on the MIFARE Classic [online]. http://www.cs.ru.nl/~flaviog/publications/Attack.MIFARE.pdf, 2008-03-15 [cit. 2010-04-20].

[15] Lew, A.: Hackers Crack London Tube's Ticketing System [online]. http://www.wired.com/autopia/2008/06/hackers-crack-l/, 2008-06-24 [cit. 2010-04-20].

[16] MasterCard: Co je to MasterCard PayPass™[online]. http://www.mastercard.com/cz/personal/cz/sluzby/paypass.html, 2010 [cit. 2010-05-01].

[17] NXP Semiconductors GmbH: MF1ICS50 - Functional specification [online]. http://www.nxp.com/acrobat/other/identification/001055.pdf, 2009-12-14 [cit. 2010-04-20].

[18] NXP Semiconductors GmbH: MIFARE Smartcard IC's [online]. http://www.mifare.net/products/smartcardics/index.asp, 2010 [cit. 2010-05-05].

[19] On Track Inovation Ltd.: ISO 14443 - An introduction to the contactless standard for smart cards and its relevance to customers [online]. http://www.otiglobal.com/objects/ISO%2014443%20WP%204.11.pdf, 08-10-2003 [cit. 2010-04-14].

[20] Palán, M.: Bezkontaktní čipové karty Českých drah [online]. http://www.cdrail.cz/VTS/CLANKY/vts21/2108.pdf, 2006 [cit. 2010-04-20].

[21] Wikipedia: Transponder — Wikipedia, The Free Encyclopedia [online]. http://en.wikipedia.org/w/index.php?title=Transponder&oldid=345540742, 2010 [cit. 2010-05-01].

[22] Wikipedia: Screensaver — Wikipedia, The Free Encyclopedia [online]. http://en.wikipedia.org/w/index.php?title=Screensaver&oldid=360497713, 2010 [cit. 2010-05-09].

[23] Zetter, K.: Hackers Clone E-Passports [online]. http://www.wired.com/science/discoveries/news/2006/08/71521, 2006-03-08 [cit. 2010-04-16].

Dodatok A

Obsah CD

- `/`
  - `bin` — Preložené binárne súbory aplikácii a ovládača.
    - `x86` — Pre 32 bitové architektúry.
    - `amd64` — Pre 64 bitové architektúry.
  - `src` — Zdrojový kód všetkých častí.
    - `driver` — Ovládač pre čítačku `ACR 128`.
    - `qMifareScreenLocker` — Aplikácia pre uzamykanie obrazovky.
    - `pam_mifare` — PAM modul.
  - `doc` — Generovaná programová dokumentácia.
  - `text` — Text tejto technickej správy.
  - `latex` — LaTeX-ové zdrojové súbory a obrázky technickej správy.

Dodatok B

Manuál

B.1 Inštalácia

B.1.1 Ovládač `libacr128.so`

Pre inštaláciu ovládača je nutné mať nainštalovanú vývojovú verziu balíka `libusb-1.0`. Inštaláciu spustíme v adresári so zdrojovým kódom príkazom:

```
$ sudo make install
```

Ovládač sa nainštaluje do systému prístupnej cesty spolu s hlavičkovými súbormi. Pre odinštaláciu môžeme využiť príkaz:

```
$ sudo make uninstall
```

B.1.2 Aplikácia `qMifareScreenLocker`

Pre preloženie tejto aplikácie je potrebné mať nainštalovaný ovládač `libacr128.so` a vývojovú verziu knižnice `Qt` aspoň verzie 4.5. Aplikáciu preložíme nasledovne:

```
$ qmake
$ make
```

Aplikáciu potom spustíme pomocou príkazu:

```
$ ./qMifareScreenLocker
```

B.1.3 Modul `pam_mifare.so`

Pre inštaláciu tohoto modulu je potrebné mať nainštalovaný ovládač `libacr128.so` a vývojovú verziu knižnice `libpam`. Modul preložíme a nainštalujeme nasledovne:

```
$ make
$ sudo make install
```

Modul je potom použiteľný ako ostatné PAM moduly. Pre odinštaláciu slúži príkaz:

```
$ sudo make uninstall
```

B.2 Používanie

B.2.1 `qMifareScreenLocker`

Pri spustení sa táto aplikácia minimalizuje do systémovej oznamovacej oblasti. Pre vyvolanie nastavovacieho okna je potrebné kliknúť na ikonku kľuča (Obr. 5.1). V prípade, že sú v internej databázi nahraté oprávnené karty, je pred otvorením okna konfigurácie vyžadované

priloženie karty s oprávnením "Manager". Keď je zobrazené konfiguračné okno, aplikácia nevykonáva uzamknutie a odomknutie pracovnej plochy. V spodnej časti okna sa nachádzajú tlačidlá pre ukončenie aplikácie (Exit application) a minimalizáciu okna (Close). Konfiguračné okno (Obr. 5.2) sa delí na 4 záložky. Záložka "About" obsahuje krátke informácie o programe. V záložke "Log" je vypísaný záznam transakcii pre prípadné overenie.

Záložka "Cards"

Na tejto záložke je možné prehliadať autorizované karty a pomocou tlačitiel "Add Card" a "Remove card" tento zoznam modifikovať. Pre pridanie karty je najprv potrebné vyplniť položku poznámky a voľbu oprávnenia "Manager" a potom kliknúť na tlačidlo "Add Card". Objaví sa okno vyzívajúce užívateľa pre priloženie karty. Karta je hneď pridaná do databázy.

Záložka "Configuration"

Na tejto záložke je možné modifikovať nastavenia aplikácie a príkazy pre uzamknutie a odomknutie pracovnej plochy. Položka "Lock timeout" slúži na nastavenie spozdenia pred zamknutím plochy. "Polling delay" nastavuje dobu, po ktorej je kontrolovaná prítomnosť karty. Daľej je možné zvoliť zobrazenie výstrahy pred uzamknutím, zobrazenie okna konfigurácie po štarte, nastavenie a aktiváciu príkazov uzamknutia a odomknutia.

B.2.2 `pam_mifare.so`

Pre využitie modulu pam_mifare.so v systéme PAM je potrebné pridať riadok do konfigurácie pre aplikáciu, ktorá ho má využiť. Pridanie nasledovného riadku do súboru `/etc/pam.d/gdm` spôsobí prihlásenie užívateľa do systému X window po preukázaní platnou kartou bez nutnosti zadať heslo:

```
auth    sufficient    pam_mifare.so    /etc/mifaredb fail
```

Aby bolo možné využiť kartu Mifare ako dodatočný prvok autentizácie, je potrebné do konfigurácie uviesť nasledujúci riadok.

```
auth    required    pam_mifare.so    /etc/mifaredb fail
```

Parameter `fail` značí, že v prípade neúspechu komunikácie s čítačkou dôjde k chybe autentizácie. Daľšie možnosti nastavenia a vysvetlenie konfiguračných nastavení sú popísané v súbore `README` v adresári s zdrojovým kódom modulu.

Databázový súbor

Databázový súbor je v jednoduchom textovom formáte:

```
login;výrobné číslo;poznámka
```

Pre pridanie autorizovanej karty slúži nástroj `mifareaddcard`. Pre pridanie karty užívateľovi "pipo" s poznámkou "isic" do súboru "/etc/mifaredb" slúži nasledovný príkaz:

```
# mifareaddcard pipo /etc/mifaredb isic
```

Pre odstránenie oprávnení je možné využiť textový editor a oprávnenie zmazať.

Prihlásenie pomocou modulu pam_mifare.so

Prihlásenie v programe `su` vyzerá pri použití modulu pam_mifare.so nasledovne:

```
pipo@arctica:~$ su -
Place card on the reader
root@arctica:#
```

Dodatok C

Konfiguračný súbor systému PAM

```
 #%PAM-1.0

auth       required pam_securetty.so
auth       required pam_tally.so file=/var/log/faillog onerr=succeed
auth       required pam_shells.so
auth       required pam_nologin.so
auth       include system-auth

account    required pam_access.so
account    include system-auth
account    required pam_tally.so file=/var/log/faillog onerr=succeed

password   include system-auth

session    required pam_env.so
session    optional pam_lastlog.so
session    optional pam_motd.so motd=/etc/motd
session    optional pam_mail.so

session    include system-auth
```

Dodatok D

Dáta uložené na študentskej karte

```
pipo@aeon ~/bp/bin $ ./mifare1kdump
Card serial number: FA C3 B2 2C
Reading card data with key FF:FF:FF:FF:FF:FF
Block 00: FA C3 B2 2C A7 88 04 00 46 8E 66 54 5D 70 34 06
Block 01: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 02: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 03: 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF
Block 04: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 05: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 06: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 07: 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF
Block 08: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 09: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 10: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 11: 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF
Block 12: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 13: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 14: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 15: 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF
Block 16: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 17: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 18: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 19: 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF
Block 20: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 21: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 22: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 23: 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF
Block 24: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 25: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 26: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 27: 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF
Block 28: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 29: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 30: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 31: 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF
Block 32: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 33: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 34: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 35: 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF
Block 36: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 37: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 38: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 39: 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF
Block 40: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 41: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 42: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 43: 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF
Block 44: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 45: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 46: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 47: 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF
Block 48: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 49: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 50: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 51: 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF
Block 52: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 53: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 54: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 55: 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF
Block 56: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 57: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 58: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 59: 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF
Block 60: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 61: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 62: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 63: 00 00 00 00 00 00 FF 07 80 BC FF FF FF FF FF FF
```