VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

VYUŽITÍ DRUHÉHO SIM SLOTU TELEFONU JAKO BEZPEČNOSTNÍHO MODULU

UTILIZATION OF THE SECOND SIM PHONE SLOT AS A SECURITY MODULE

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE **JAN OŠKERA**
AUTHOR

VEDOUCÍ PRÁCE **Doc. Dr. Ing. Petr Hanáček**
SUPERVISOR

BRNO 2018

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav inteligentních systémů Akademický rok 2017/2018

Zadání bakalářské práce

Řešitel: **Oškera Jan**
Obor: Informační technologie
Téma: **Využití druhého SIM slotu telefonu jako bezpečnostního modulu**
Second SIM Slot as a Secure Element
Kategorie: Bezpečnost

Pokyny:

1. Prostudujte dostupné materiály o problematice SAM (secure access module card) pro GSM telefony.
2. Navrhněte způsob implementace SAM jako modulu ve druhém SIM slotu GSM telefonu.
3. Navrhněte vhodné API pro přístup k SAM z telefonu (Android) a demonstrujte jeho funkčnost.
4. Navržené řešení implementujte a otestujte. Vytvořte podrobnou technickou dokumentaci.

Literatura:

- Podle pokynů vedoucího

Pro udělení zápočtu za první semestr je požadováno:

- Body 1 a 2 zadání.

Podrobné závazné pokyny pro vypracování bakalářské práce naleznete na adrese http://www.fit.vutbr.cz/info/szz/

Technická zpráva bakalářské práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap (20 až 30% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Hanáček Petr, doc. Dr. Ing.**, UITS FIT VUT
Datum zadání: 1. listopadu 2017
Datum odevzdání: 16. května 2018

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav inteligentních systémů
612 66 Brno, Božetěchova 2

L.S.

doc. Dr. Ing. Petr Hanáček
vedoucí ústavu

Abstrakt

Tato bakalářská práce se zabývá návrhem a implementací využití dvojího slotu v GSM telefonu jako bezpečnostního elementu. Pro tento účel je třeba vytvořit SAM podporující GSM funkčnost. Nebo použít již existující SAM podporující tuto funkčnost, takové karty se označují SIM/USIM.

Abstract

This bachelor thesis deals with design and implementation use of dual slot in GSM phone like security element. For this purpose, you need to create a SAM that supports GSM functionality. Or use existing SAM that support this functionality, such cards are called SIM/USIM.

Klíčová slova

GSM (Globální Systém pro Mobilní komunikace), JDK (vývojový nástroj pro Javu), ME (vybavení telefonu), SIM (účastnická identifikační karta), GlobalPlatform (standardizace infrastruktury pro vývoj, nasazení a správu čipových karet), MS (mobilní stanice), ATR (odpověď na restart karty), ISO/IEC 7816 (soubor mezinárodních norem), SW1/SW2 (stavové slovo, odpověď na APDU příkazy), APDU (aplikační protokol datové jednotky), CLA (třída instrukcí), INS (specifikuje instrukci), P1/P2/P3 (parametr APDU příkazu), RFU (rezervováno pro budoucí použití), ADM (bezpečnostní podmínka přístupu), Kc (šifrovací klíč), ID (identita), HPLMN (domovská veřejná pozemní mobilní síť), BCCH (kanál pro řízení vysílání), PLMN (veřejná pozemní mobilní síť), AUTN (autentifikační token), RAND (náhodně vygenerované číslo), AK (klíč anonymity), MODE (specifikuje řízení), MAC (kód ověřování zpráv), RES (generovaná odpověď), CK (šifrovací klíč, někdy označován jako Kc), IK (klíč integrity), OTA (bezdrátové), RLI (knihovna v telefonu, komunikuje se SIM/USIM), GUI (grafické uživatelské rozhraní), PUK (bezpečnostní klíč k odblokování PINU), CHV1/CHV2 (jedná se o PIN), AID (ID aplikace), KI (ověřovací klíč), BTS (Základnové stanice vysílačů sítě GSM), BSC (Systém základnových stanic), TRAU (Transkodér a jednotka pro úpravu rychlosti), MSC (Ústředna veřejné mobilní sítě), PSTN (Veřejné telefonní sítě), HLR (Domovský registr), AUC (Centrum ověřování), EIR (Registr mobilních zařízení), VLR (Návštěvnický registr), SMSC (Středisko krátkých textových zpráv)

Keywords

GSM (Global System for Mobile communications), JDK (Java Development Kit), ME (Mobile equipment), SIM (Subscriber identity module), GlobalPlatform (standardization of infrastructure for the development, deployment and management of smart cards), MS (Mobile Station), ATR (Answer to reset), ISO/IEC 7816 (set of international standards), SW1/SW2 (status word), APDU (application protocol data unit), CLA (instruction class), INS (specifies instruction), P1/P2/P3 (parameter of the APDU command), RFU (reserved for future), ADM (security access condition), Kc (cipher key), ID (Identifier), HPLMN (Home Public Land Mobile Network), BCCH (Broadcast Control CHannel), PLMN (Public land mobile network), AUTN (authentication token), RAND (random number), AK (anonymity key), MODE (specifies management), MAC (Message authentication code), RES (response), CK (encryption key, sometimes referred to as Kc), IK (integrity key), OTA (Over the air), RLI (ibrary in the phone, communicates with SIM/USIM), GUI (Graphical user interface), PUK (PIN Unblocking Key), CHV1/CHV2 (Card Holder Verification information), AID (application ID), KI (authentication key), BTS (Base transceiver station), BSC (Base Station Controller), TRAU (Transcoder and Rate Adaptation Unit), MSC (Mobile Switching Centre), PSTN (Public switched telephone networks), HLR (Home Location Register), AUC (Authentication Center), EIR (Equipment Identity Register), VLR (Visitor Location Register), SMSC (Short Message Service Center)

Citace

Oškera, Jan: Využití druhého SIM slotu v telefonu jako bezpečnostního modulu, bakalářská práce, Brno, FIT VUT v Brně, 2018

VYUŽITÍ DRUHÉHO SIM SLOTU TELEFONU JAKO BEZPEČNOSTNÍHO MODULU

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Doc. Dr. Ing. Petr Hanáčka.
Další informace mi poskytli Zdeněk Skalák a David Přibyl.
Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

……
Jan Oškera
14.05.2018

Poděkování

Převážně bych chtěl poděkovat mému vedoucímu bakalářské práce za vytvoření a vypsání tohoto zadání bakalářské práce. A za cenné rady, jež mi poskytl. Dále bych chtěl poděkovat firmě Monet+ a jejím zaměstnancům za zapůjčení prostředků potřebných k vypracování této práce. A rád bych poděkoval za veškerou důvěru, podporu a pomoc při její tvorbě.

Obsah

1 Úvod ... 3
2 Rozbor tématu a použitých technologií ... 4
2.1 Technologie Java Card ... 4
2.2 Technologie SIM/USIM ... 5
2.3 Technologie GSM ... 6
2.4 Knihovna RIL (Radio Interface Layer) ... 7
2.5 Technologie OTA (Over the air) ... 7
2.6 Sim Application Toolkit ... 8
2.7 Požadavky na aplikaci ... 8
3 Přehled současného stavu ... 9
3.1 Schéma komunikace ... 9
3.1.1 Formát příkazu ... 10
3.1.2 Formát odpovědi ... 10
3.2 Souborový systém ... 11
3.2.1 Přístupové podmínky (Access Conditions) ... 12
3.2.2 Druhy souborů ... 12
3.2.3 Pohyb v souborovém systému ... 13
3.2.4 SIM ... 14
3.2.5 USIM ... 14
3.3 Popis jednotlivých příkazů ... 15
3.3.1 Příkaz umožňující pohyb po souborovém systému ... 16
3.3.2 Žádost o podrobné informace o aktuálním adresáři ... 17
3.3.3 Příkazy pro práci s transparentními soubory ... 17
3.3.4 Správa souborových záznamů ... 18
3.3.5 Správa autorizačních příkazů ... 19
3.3.6 Aktivace a reaktivace souborů ... 20
3.3.7 Příkazy spojené s autentizací SIM/USIM karty v síti ... 21
3.3.8 Zastaralé příkazy ... 22
3.3.9 Příkaz pro získání dodatečných dat ... 22
3.3.10 Příkazy pro obsluhu Sim Application Toolkit ... 22
3.3.11 Správa logických kanálů ... 23
3.4 Popis povinných souborů ... 24
3.4.1 Povinné soubory technologie SIM ... 24
3.4.2 Povinné soubory vyskytující se pouze v technologii USIM ... 27

3.5 Autentizace a její zabezpečení 29
3.5.1 SIM 29
3.5.2 USIM 30
3.6 Rozdíly mezi SIM a USIM 32
4 Návrh řešení 33
4.1 Způsoby řešení 33
4.1.1 Pomocí menu 33
4.1.2 Pomocí technologie OTA 34
4.1.3 Pomocí RIL lib. (Radio Interface Layer) 34
4.1.4 Pomocí originální SIM/USIM a knihovny RIL 35
5 Způsob implementace 36
5.1 Implementace Java karty s funkčností GSM 36
5.1.1 Odposlouchávání komunikace mezi ME a SIM/USIM 36
5.1.2 Debugger 36
5.1.3 Obsahy požadovaných souborů 36
5.1.4 Implementovaná GSM funkčnost 37
5.1.5 Technologie USIM 37
5.1.6 Technologie SIM 38
5.1.7 Pokusy o komunikaci s neautentizovanou SIM/USIM kartou 39
5.2 Využití originální SIM/USIM a knihovny RIL 40
5.2.1 Implementace komunikace 40
5.2.2 Šifrovací algoritmus 42
5.2.3 Interakce s uživatelem 43
5.2.4 Vytvořené funkce 43
5.2.5 Omezení a způsob implementace 44
5.3 Program použitý pro komunikaci s kartou 44
5.4 Použité Java a SIM/USIM karty 44
5.5 Informace o použitých mobilních zařízeních 44
5.6 Informace o použitých IDE 44
6 Závěr 45
Literatura 46
Přílohy 49
A Obsah CD 50
B Manuál k Android aplikaci 51
C Manuál k instalaci a řízení Appletů 52
D Návratové kódy 53

1 Úvod

Tato zpráva vznikla jako dokumentace k bakalářské práci zabývající se využitím druhého SIM slotu v telefonu jako bezpečnostního elementu. Naším úkolem bude navrhnout a implementovat metodu, která bude využívat bezpečnostní přístupový modul v podobě karty. Tuto kartu poté vložíme do SIM slotu mobilního telefonu a s její pomocí vytvoříme bezpečnostní element. Na tento element budeme schopni přes mobilní aplikaci ukládat data. Přístup k takto uloženým datům, musí být samozřejmě vhodně zabezpečen.

Převážná část obyvatelstva již vlastní mobilní telefon nebo jiné zařízení, jehož součástí je i SIM slot. Kdyby bylo možné tuto kartu využít pro ukládání bezpečnostních klíčů, uživatelé by měli bezpečnostní klíče stále u sebe. Tyto klíče by například mohli sloužit pro autentizaci účastníka v libovolné síti. Nebo pro jeho autorizaci a následný přístup ke specifickým službám.

V dnešní době hraje bezpečnost velkou roli. Téměř každý již má k dispozici elektronické a jiné prostředky, které musejí být zabezpečeny. Zneužití nezabezpečených prostředků může přinést uživateli hrozivé důsledky. Z tohoto důvodu existuje celá řada možností, jak tyto nástroje zabezpečit. My se budeme zabývat metodou, která ještě neexistuje. Naše metoda může přinést inovaci do bezpečnostních technologií. Tato technologie by umožňovala uživateli přenášet bezpečnostní klíče mezi jednotlivými zařízeními. Přenášení dat by bylo velmi jednoduché, stačilo by vyndat příslušnou kartu a vložit ji do zvoleného zařízení.

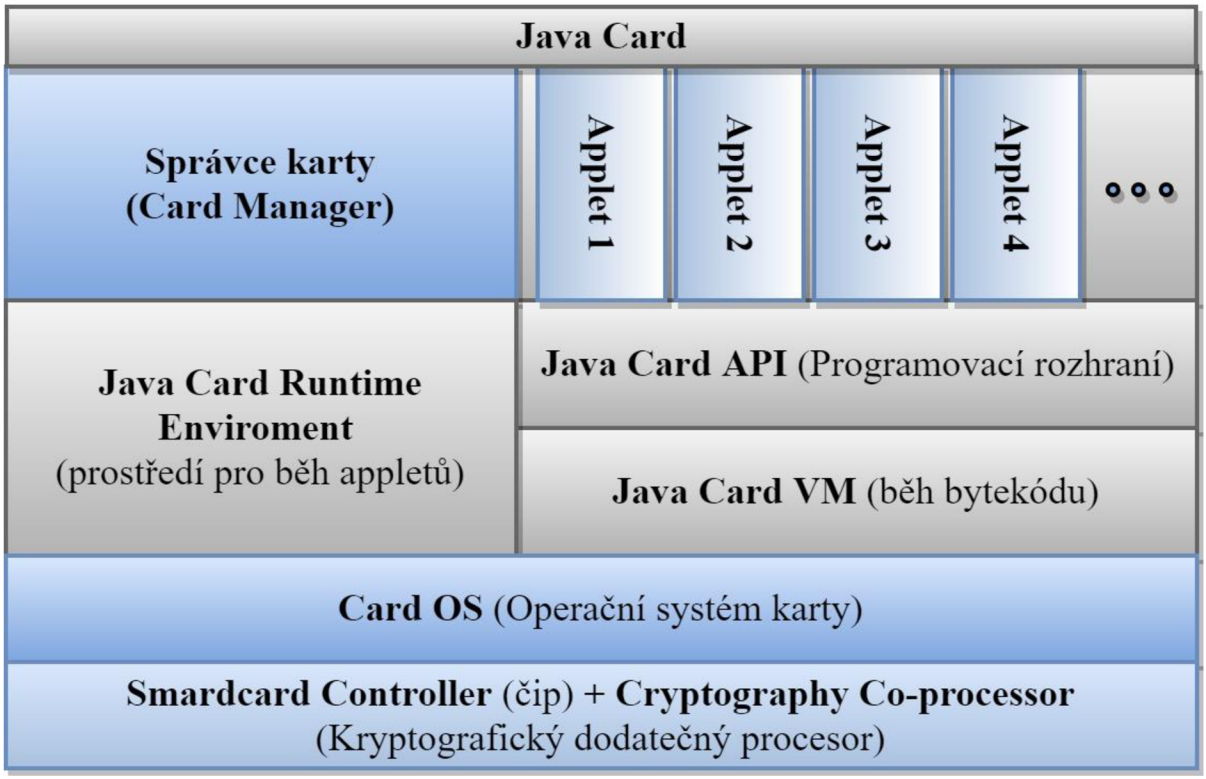
V následujících kapitolách naleznete základní informace o všech použitých technologií. Zaměříme se na jejich současný stav a funkčnost, kterou umožňují. Potom se pokusíme navrhnout možné způsoby řešení. Zvolíme si nejvhodnější řešení a to vhodně implementujeme. V závěrečné kapitole shrneme získané znalosti a obeznámíme vás s výsledkem práce.

2 Rozbor tématu a použitých technologií

V této kapitole jsou popsány jednotlivé technologie, na nichž je tato práce založená. Naleznete v ní i bližší informace o knihovně RIL a celkové požadavky na výslednou aplikaci.

2.1 Technologie Java Card

V této části se dozvíme základní informace o technologii Java Card. Seznámíme se s její strukturou a důležitostí. Náhled na zjednodušenou strukturu je dostupný v následující fotce.



Obrázek 2.1: Struktura technologie Java Card [34]

Tato technologie se široce využívá k bezpečnému běhu aplikací tzv. appletů na čipových kartách, známých také pod názvem smart cards a jiných málo paměťových zařízeních. Je tedy především určena pro jednoúčelová malá vestavěná zařízení.

Jedná se o nejmenší součást jazyka Java, která je kompatibilní se všemi standardy paměťových karet. Pro vývoj těchto aplikací slouží tzv. JDK (Java Development Kit). Pro správné fungování aplikací je třeba, aby karta podporovala danou verzi vývojového nástroje.

Na jednom zařízení se může nacházet i více aplikací. Mezi jednotlivými aplikacemi se přepíná pomocí AID (Application Identifier). Pouze jednu aplikaci lze nastavit jako výchozí, která se po restartu karty automaticky vybere pro komunikaci.

Pro zabezpečení a správu těchto aplikací slouží tzv. GlobalPlatform, který také slouží pro standardizaci infrastruktury vývoje.

Kvůli bezpečnosti lze využít i šifrovanou komunikaci s terminálem. Stahování aplikací je zabezpečeno pomocí digitálních podpisů. Pro nahrávání knihoven a instalaci aplikací je třeba se autorizovat pomocí jednoho z unikátních bezpečnostních klíčů. Tyto klíče zamezují neautorizovanou manipulaci s kartou. Po deseti neúspěšných pokusech o autorizaci se karta nenávratně zablokuje.

Oproti ostatním platformám vycházejících z jazyka Java, podporuje Java Card pouze jednoduché datové typy byte, short, boolean, případně int. Velkým nedostatkem této technologie je i nepodporování více rozměrných polích a vláken. Jednotlivé příkazy musí být vykonávány atomicky.

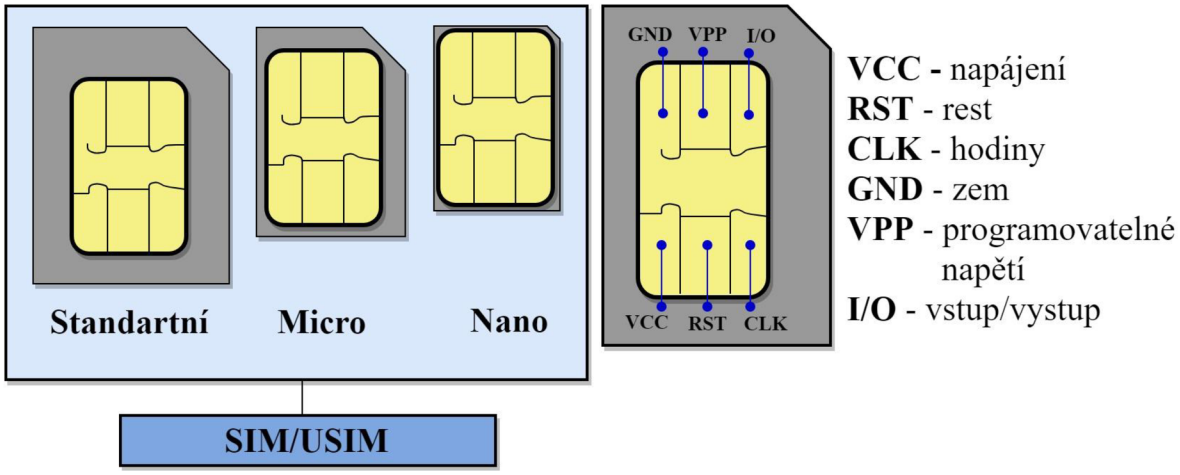
Každá aplikace (applet) vytvořená pro čipové karty, musí obsahovat funkci install pro instalaci aplikace a funkci proces, zajišťující správný běh aplikace. Bytekód poté běží na Java Card VM.

Díky univerzálnosti této technologie vyvinuté společností Sun Microsystems, lze aplikace rychle vyvíjet, testovat i zavádět. Původně byla však vytvořena pouze pro zabezpečení dat uložených na čipové kartě.

2.2 Technologie SIM/USIM

Následující kapitola udává základní informace o technologii SIM/USIM. S tuto technologií se nepřímo setkáváme denně, kdykoliv zapneme mobilní telefon. Je velmi rozšířená po celém světě.

Karty jsou dostupné ve třech velikostech. Náhled je dostupný níže. Snímek obsahuje i jednotlivé kontakty a jejich význam.



Obrázek 2.2: SIM/USIM a jejich hardware [42][38]

Naleznete ji na čipových kartách označovaných jako SIM (Subscriber identity module). Jedná se vlastně o jistý typ SAM (Secure access module card). Tyto karty slouží pro bezpečnou identifikaci účastníka v mobilních sítích. Používají se běžně v mobilních telefonech, GPS navigaci i pro bezdrátový přístup k internetu.

Největší výhodou SIM je, že je snadno přenositelná mezi pestrou škálou různých zařízení. Tento bezpečnostní modul obsahuje údaje potřebné pro autentizaci a přístup ke službám poskytovaných operátorem.

Pro autentizaci v mobilní síti je třeba celá řada informací. Ty jsou společně s uživatelskými daty bezpečně uloženy na kartě. Jedná se o IMSI (International Mobile Subscriber Identity) sloužící k dohledání detailů o uživateli. Unikátní sériové číslo identifikující kartu, označované jako ICCID (Integrated Circuit Card Identifier). Tyto dva údaje dohromady jednoznačně identifikují zákazníka. Pro ještě větší úroveň zabezpečení slouží KI (Authentication key), jehož účelem je ověřit pravost SIM karty. Tento klíč nelze získat z karty přímo. Je uložen nejen na SIM kartě, ale i u poskytovatele služeb. Používá se při běhu autentizačního algoritmu, výsledek je poté porovnán a vyhodnocen. Uložené data mají různou úroveň zabezpečení. Jednotlivé úrovně zabezpečení a podrobnější informace lze nalézt ve třetí kapitole.

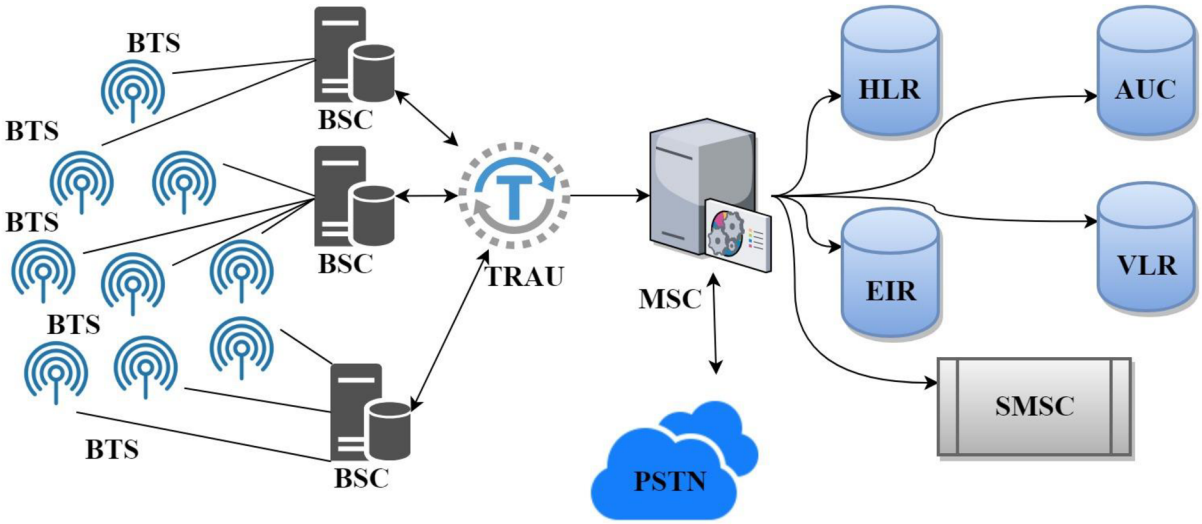
Autorizace uživatele na straně mobilního telefonu probíhá pomocí bezpečnostního čísla PIN (Personal identification number). Po úspěšné autorizaci si vlastník karty může toto bezpečnostní kritérium libovolně zapínat a vypínat. Minimální velikost bezpečnostního kódu je 4 a maximálně může dosahovat velikosti 8. Je-li dosaženo maximálního počtu neúspěšných pokusů o autorizaci, je třeba

zadat nový PIN pomocí PUK (Personal Unlocking Key). Může obsahovat i více PIN a PUK kódů. Po dosažení maximálního počtu neúspěšných pokusů zadání PUK kódu, je karta nenávratně zablokována.

Existují i další technologie, jako jsou CSIM, ISIM atd. Jsou ale méně známé a hůře dostupné. Proto nebudou podrobněji probírány.

2.3 Technologie GSM

V této podkapitole se seznámíme s technologií GSM. Zaměříme se na její význam, strukturu a důležité části systému. V následujícím snímku si můžete prohlédnout zjednodušenou strukturu GSM sítě. Definici jednotlivých prvků sítě naleznete na konci podkapitoly.



Obrázek 2.3: Struktura GSM sítě [4][37]

Jedná se o globální systém pro mobilní komunikaci, který je nejpoužívanějším standardem pro mobilní telefony na světě. Specifikující například možnosti způsobů komunikace mezi mobilním telefonem a SIM kartou, jejímž hlavním účelem je autentizace a komunikace zařízení s GSM sítí.

Zachovává zpětnou kompatibilitu se staršími verzemi. Jedná se o buňkovou síť fungující na několika rádiových frekvencích. Koncová zařízení jsou mobilní telefony, které se připojují k síti pomocí nejbližší buňky. Celá síť se skládá z různých elementů, jako jsou vysílače, přijímače, registry, autentizační středisko a další. A hlavní sítí nazývající se GPRS (General packet radio service), umožňující spojení na bázi šifrované paketové komunikace.

Novější technologie GSM sítě se označuje jako UMTS (Universal Mobile Telecommunication System). Její struktura je velmi podobná struktuře GSM sítě. Pro zjednodušení zde její rozšíření nebudeme uvedádět.

Důležité části systému

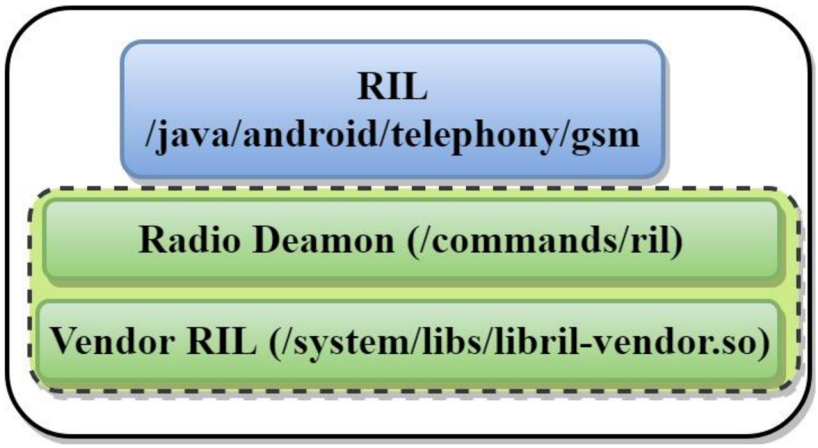
- BTS (Base transceiver station) – Základnové stanice vysílačů sítě GSM
- BSC (Base Station Controller) - Systém základnových stanic
- TRAU (Transcoder and Rate Adaptation Unit) - Transkodér a jednotka pro úpravu rychlosti
- MSC (Mobile Switching Centre) - Ústředna veřejné mobilní sítě
- PSTN (Public switched telephone networks) - Veřejné telefonní sítě
- HLR (Home Location Register) - Domovský registr
- AUC (Authentication Center) - Centrum ověřování
- EIR (Equipment Identity Register) - Registr mobilních zařízení
- VLR (Visitor Location Register) - Návštěvnický registr
- SMSC (Short Message Service Center) - Středisko krátkých textových zpráv
- PLMN (Public land mobile network) - Veřejná pozemní mobilní síť

V této části jsme se seznámili se základní strukturou a funkčností GSM sítě.

2.4 Knihovna RIL (Radio Interface Layer)

Zde se seznámíme s knihovou RIL a jejím účelem. Je nepřímo součástí balíčku android, kterou naleznete v každém telefonu s operačním systémem android. Pomocí této knihovny lze nepřímo komunikovat mezi telefonní aplikací a SIM/USIM kartou.

Komunikace probíhá přes prostředníka, který se nazývá Baseband procesor. Jedná se o zařízení, které nám řídí veškerou funkčnost rádia. Tento prostředník dále komunikuje přímo se SIM/USIM kartou.



Obrázek 2.4: RIL knihovna [41][25]

V obrázky výše je k dispozici náhled na knihovnu RIL a její komponenty.

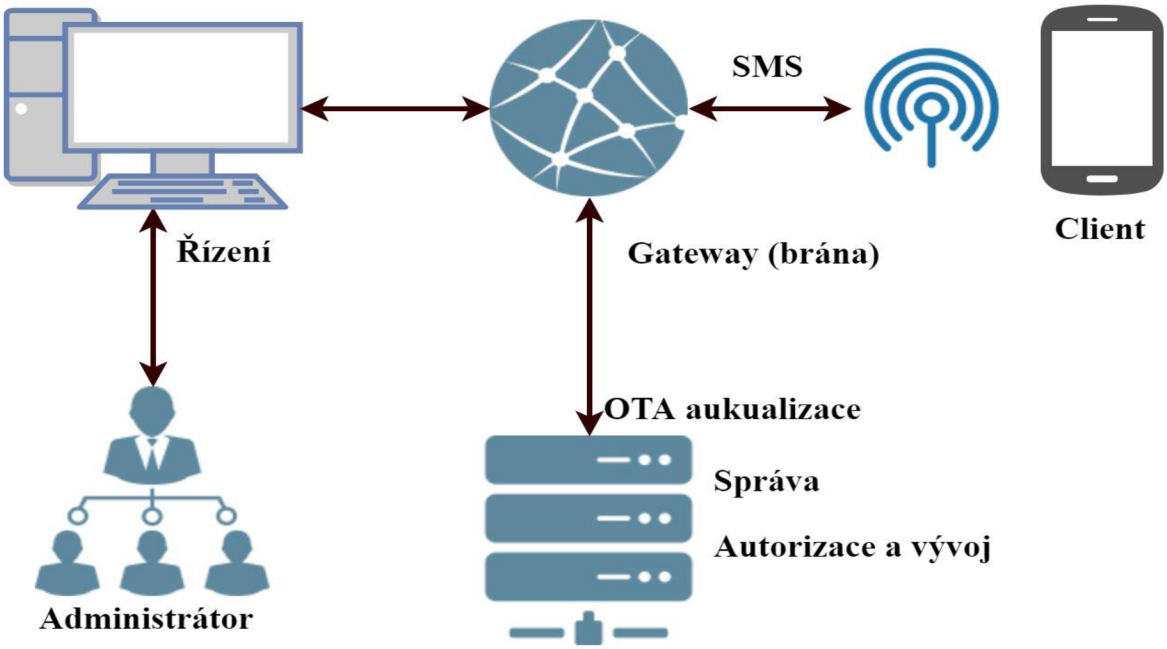
Knihovna RIL se skládá ze dvou komponentů. Mezi ně patří například Vendor RIL, který je specifický pro každou implementaci rádio komunikace. Komponent Radio Daemon je ve spojení s telefonními službami. Účelem RIL (Radio Interface Layer) je poskytnout rozhraní rádiové komunikaci a modemu. Modemem je myšleno rádio, které přijímá a odesílá data.

Tato knihovna vyřizuje dva druhy příkazů. Prvním z nich jsou vyžádané příkazy, které jsou iniciované z horní vrstvy. Do horní vrstvy patří například obsluha vytáčení, zasílání SMS atd. Tyto požadavky jsou přijaty pomocí Radio Daemon, který je poté přepošle do Vendor RIL. Druhým typem jsou nevyžádané příkazy, ty jsou inicializovány modemem. Jsou zpracovávány komponentou Vendor RIL. Nevyžádané příkazy jsou obsluhovány až po vyžádaných příkazech. Požadavky od služeb nemůžeme odkládat, proto jsou vyřízeny prioritně.

2.5 Technologie OTA (Over the air)

Tato technologie umožňuje vzdálené nahrávání, aktualizaci, instalaci appletů a služeb. Poskytovatel služeb, tímto způsobem může modifikovat aplikace a data uložená na SIM/USIM kartě. Veškerým modifikacím, měnícím uložená data, budeme dále říkat aktualizace.

Aktualizace jsou odeslány do OTA brány (OTA Gateway). Brána transformuje žádost do krátkých zpráv. Ty jsou poté doručeny do SIM/USIM karty pomocí služby SMS. Průběh komunikace si můžete prohléhnout v obrázku níže.



Obrázek 2.5: Specifikace technologie OTA [43]

Řízení OTA brány (Gateway) je spravováno administrátory. S touto bránou komunikuje orgán spravující a vyvíjející nové aktualizace. Tyto aktualizace jsou poté pomocí brány a SMS služby odeslány do příslušných klientských zařízení. Zařízením je myšleno vybavení obsahující SIM/USIM karty.

2.6 Sim Application Toolkit

Zde se seznámíme se základními informacemi spojenými s touto technologií. Jedná se o speciální applet, který umožňuje spravovat aplikace uložené na SIM/USIM kartě. Při příchozím příkazu vygeneruje vhodnou událost a informuje vybranou aplikaci. Vytváří sdílené prostředí pro všechny aplikace. Umožňuje speciální funkčnost, jakou je například vytváření výběrového menu atd. Nenachází se na starých kartách typu SIM. Bližší informace naleznete ve specifikaci GSM [11.14].

2.7 Požadavky na aplikaci

Následující podkapitola definuje požadavky na naši výslednou aplikaci. I když neobsahuje příliš požadavků, uvedené nezbytnosti jsou nepostradatelné pro správné fungování naší aplikace.

Naším úkolem je využít SAM (secure access module) podporující GSM funkčnost pro bezpečné ukládání dat. Z tohoto důvodu budeme potřebovat, aby výsledná aplikace podporoval následující požadavky.

- Přístup k datům a následná manipulace s daty musí být zabezpečená alespoň pomocí PINU (CHV 1)
- Použitý zabezpečený přístupový modul (SAM) musí podporovat GSM funkčnost (vytvořit vlastní nebo použít již existující)
- Správa data musí být řízená Android aplikací
- Manipulace s daty musí být jednoduchá
- Grafické rozhraní by mělo být snadno ovladatelné a příjemné na pohled

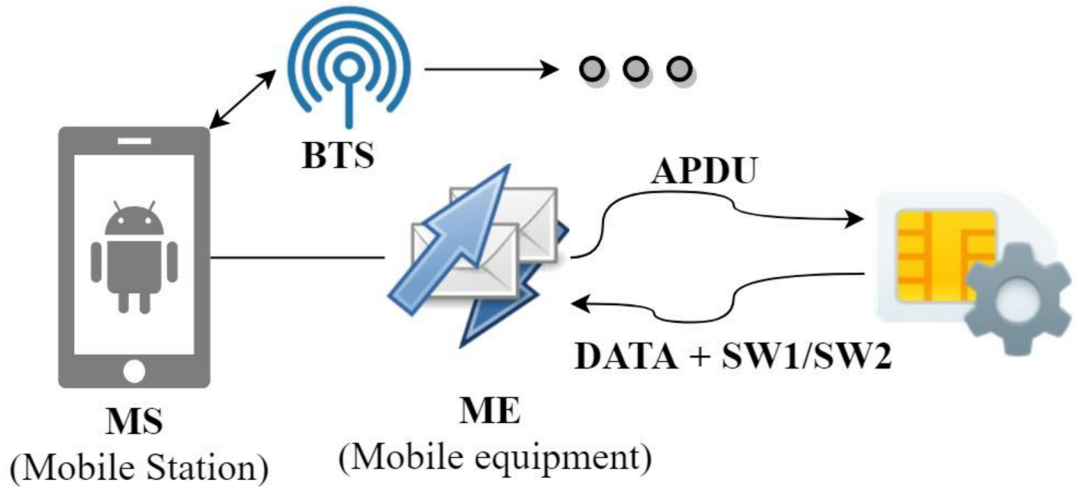
V závěrečné implementaci se budeme snažit splnit všechna zadaná kritéria.

3 Přehled současného stavu

Tato kapitola obsahuje informace týkající se současného stavu dostupných prostředků, které se využívají v SIM/USIM technologii. V částech této kapitoly se seznámíme s komunikací, souborovým systémem a podporovanými příkazy. V posledních částech nalezneme i informace o průběhu autentizace karty v GSM síti a celkové srovnání jednotlivých technologiích.

3.1 Schéma komunikace

Následující podkapitola nám umožňuje nahlédnout do komunikace mezi mobilním telefonem, označovaným jak MS (Mobilní stanice) a SIM/USIM kartou. Veškerá komunikace probíhá pomocí tzv. APDU příkazů. Informace týkající se formátu příkazu a odpovědi jsou uvedeny v následujících částech této podkapitoly.



Obrázek 3.1: Schéma komunikace MS a SIM/USIM [5]

Pro následující část práce budeme používat zkratku ME (Mobile Equipment) pro zařízení nacházející se v mobilním telefonu, jež slouží ke komunikaci se SIM kartou.

Jakmile se SIM karta ocitne pod proudem, ME si vyžádá ATR (Answer to reset). Jedná se o odpověď vycházející z normy ISO/IEC 7816, kterou obdrží terminál po elektronickém restartu karty. Obsahuje komunikační parametry navržené kartou a povahu a stav karty. Tyto parametry nelze měnit, kromě tzv. Historical bytes o maximální délce 15 bytů. Ty jsou součástí ATR a specifikující bližší údaje o způsobu použití dané karty.

Interakce mezi ME a SIM je založená na modelu Master/Slave. Kde mistrem je ME řídící komunikaci s otrokem. Otroka zde reprezentuje SIM karta. Ale i ona může zasílat příkazy do ME pomocí tzv. proaktivních příkazů. Pro správné fungování musí být tento druh příkazu podporován telefonem i SIM kartou. Příkazy lze odesílat ze SIM karty kvůli tomu, že se ME každých 30s ptá, je-li něco nového. Další podrobnější informace jsou uvedeny dále.

3.1.1 Formát příkazu

Formát komunikace je založen na protokolu APDU (Application Protocol Data Unit) vycházející ze standardu ISO 7816, využívající protokol T = 1. Prvotním příkazem ME zjistí, zdali má používat třídu sady instrukcí pro USIM nebo SIM technologii. Je-li SIM karta schopna vykonat sadu instrukcí pro USIM, je dále užívána tato sada instrukcí. Jinak předpokládá, že se jedná o SIM.

Jednotlivé příkazy musejí dodržovat následující formát.

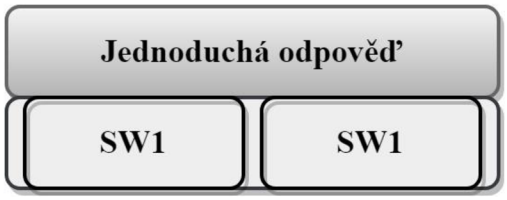
Obrázek 3.2: Formát APDU příkazu [1]

CLA – určuje třídu příkazů (SIM 0xA0, USIM 0x00 a 0x80)
INS – slouží pro rozlišení jednotlivých příkazů dané třídy
P1, P2 – parametry příkazu
P3 – třetí parametr, může reprezentovat např. délku následujících dat, počet požadovaných bytů a jiné
DATA – jedná se o volitelnou část příkazu obsahující data, jejichž délka by se měla nacházet v P3
V této části jsme se dozvěděli základní informace o formátu APDU příkazu.

3.1.2 Formát odpovědi

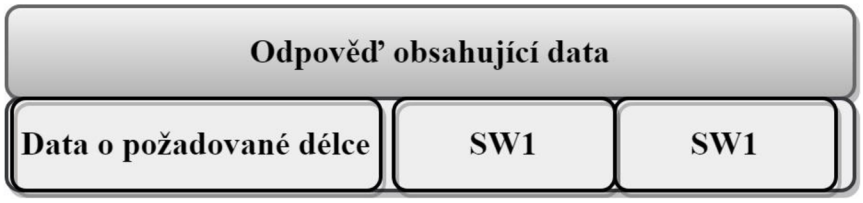
O výsledku provedení jednotlivých příkazů musí SIM karta informovat ME pomocí příslušných návratových kódů. Odeslaná odpověď musí dodržovat jeden z níže uvedených formátů.

Formát jednoduché odpovědi se skládá pouze z dvoubytové návratové hodnoty. Hodnota prvního bytu je označována jako SW1. Specifikuje kategorii, do které daná odpověď spadá. Již podle této hodnoty zjistíme, zdali byl příkaz vykonán úspěšně. Případně jaký druh chyby se vyskytl, v průběhu vykonání příslušného příkazu. Druhý byte nám slouží k poskytnutí bližších informací o jednotlivých chybách. Při úspěšném vykonání příkazu, nás může obeznámit s velikostí dostupných dat. Návratové kódy a jejich význam je dostupný v příloze D.



Obrázek 3.3: Formát jednoduché odpovědi [1]

Dalším typem je rozšířený formát jednoduché odpovědi. Odpověď nás nejen informuje o skončení vykonávání příkazu, ale obsahuje i přiložená data o maximální velikosti 256 bytů. Maximální velikost celkové odpovědi tedy může dosahovat až 258 bytů.

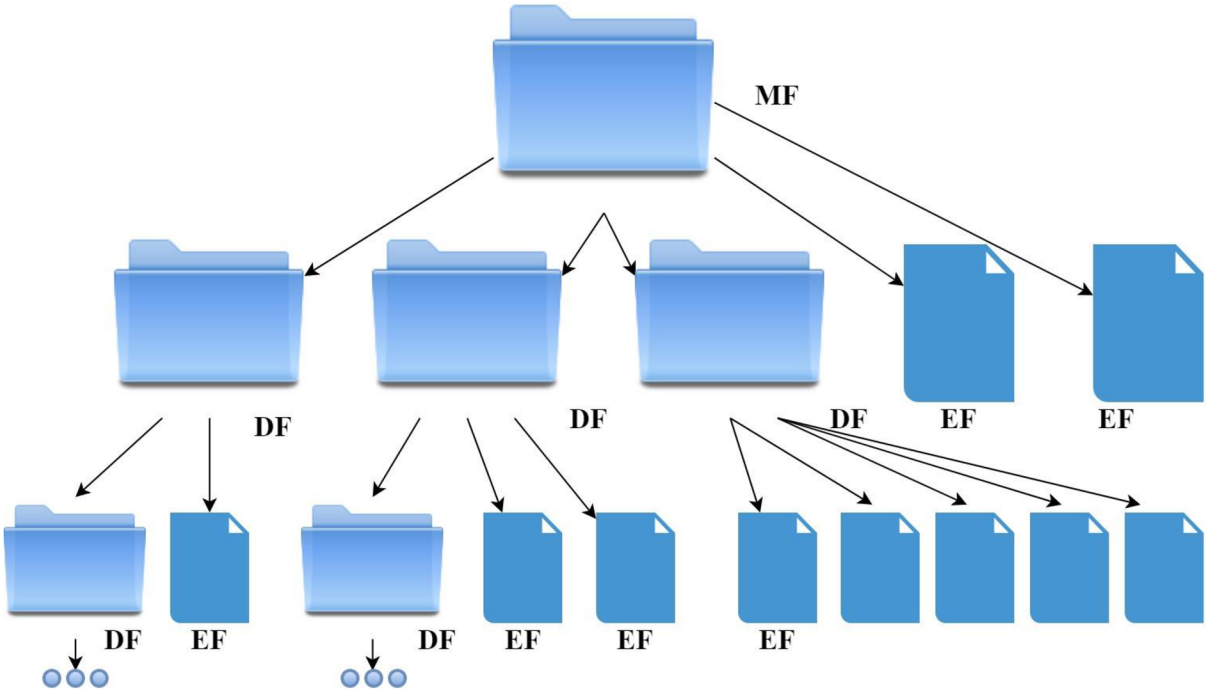


Obrázek 3.4: Formát odpovědi obsahující data [1]

SW1 – (Status Word 1) stavové slovo 1 o velikosti 1 byte
SW2 – (Status Word 2) stavové slovo 2 o velikosti 1 byte
SW1 a SW2 jsou povinnou součástí každé odpovědi

3.2 Souborový systém

Tato podkapitola nás seznámí s tím, jak vlastně souborový systém vypadá, co všechno obsahuje a jak ho správně ovládat. V přiloženém obrázku si můžete prohlédnout příklad souborové struktury.



Obrázek 3.5: Příklad souborového systém technologie SIM/USIM [1]

Na všech kartách typu SIM/USIM se nachází strukturovaný souborový systém. Pohyb po tomto systému nám umožňuje příkaz z bodu 3.3.1. Tento kapacitně limitovaný prostor nám slouží pro ukládání a čtení dat, uložených výrobcem, operátorem nebo mobilním zařízením.

Jednotlivé soubory se skládají z hlavičky a těla. Hlavička slouží pouze k vnitřním účelům a zůstává skryta mobilnímu zařízení. Obsahuje podrobné informace o daném souboru.

3.2.1 Přístupové podmínky (Access Conditions)

V tomto bodě se seznámíme s pojmem přístupové podmínky. Jedná se o podmínky, které musí uživatel splnit, aby mohl přistoupit k obsahu daného souboru. Každý soubor může mít definovanou jinou přístupovou podmínku. Jednotlivé přístupové podmínky mají různou úroveň zabezpečení, která vyjadřuje jejich obtížnost. S rostoucí úrovní se obtížnost zvyšuje. Podobná metoda se využívá i pro příkazy, aby bylo možné zabezpečit správu souborů. Tabulka obsahující tyto podmínky je k dispozici následovně.

Úroveň	Přístupová podmínka
0	Vždy
1	CHV1
2	CHV2
3	RFU
4 až 14	ADM
15	Nikdy

Tabulka 3.1: Přístupové podmínky souborů [1]

RFU - Rezervováno pro budoucí použití
ADM - Jiné bezpečnostní podmínky specifikované tvůrcem souboru
Soubory s bezpečnostní úrovní 0 jsou nezabezpečené. Naopak mají-li úroveň 15, nikdo k nim již nemá přístup.

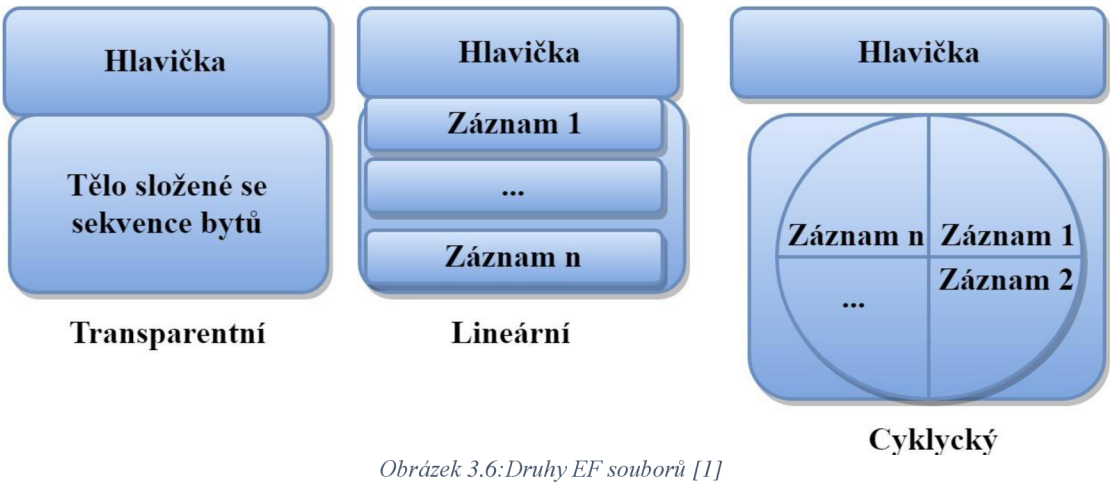
3.2.2 Druhy souborů

Následující úsek obsahuje informace o základních typech souborů, případně o jeho následném dělení na podtypy.

Existují tři základní typy souborů.

- **Hlavní soubor** (Master file) - zkráceně MF
  Jedná se o povinný adresář nadřazený všem ostatním podadresářům. Slouží jako kořen příslušného souborového systému.
- **Vyhrazený soubor** (Dedicated file) - zkráceně DF
  Podadresář MF nebo jiného DF.
- **Elementární soubor** (Elementary file) - zkráceně EF
  Běžný soubor nesoucí data. Dělí se na další kategorie.

Elementární soubory dělíme na následující podtypy.



Obrázek 3.6:Druhy EF souborů [1]

- **Transparentní EF**
  Nejběžnější typ souboru, jehož struktura těla se skládá z posloupnosti bytů.
- **Lineárně pevný EF**
  Struktura těla souboru se skládá ze sekvence záznamů s fixní délkou.
- **Cyklický EF**
  Jednotlivé záznamy mají kruhovou charakteristiku. Nelze-li vytvořit nový záznam z důvodu dosažení maximálního počtu záznamů, dojde k přepsání nejstaršího záznamu. Jednotlivé záznamy mají stejnou velikost. Záznamy obsahují ukazatel na záznam následující.

Velikost a počet záznamů je uložen v hlavičce daného souboru. Každý EF může mít až 255 záznamů o maximální velikosti 255 bytů.

3.2.3 Pohyb v souborovém systému

Zde naleznete pravidla, která je nutné dodržovat při tvorbě souborů a pohybu v souborovém systému. Pro specifikování jednotlivých souborů se využívá ID souboru. Tento identifikátor má velikost 2 byty a má hexadecimální notaci.

První byte nám specifikuje základní typ souboru.

MF	0x3F
DF pod MF	0x7F
DF pod DF	0x5F
EF pod MF	0x2F
EF pod DF	0x6F
EF pod DF pod DF	0x4F

Tabulka 3.2: Základní typy souborů [1]

ID souboru musí splňovat následující podmínky.

- ID musí být přiděleno v době vytvoření souboru
- Ve stejném adresáři nemohou existovat 2 soubory se stejným ID
- Soubor nemůže mít stejné ID s adresářem, ve kterém se nachází. (kromě kořenového adresáře)

Při pohybu v adresářích se musejí dodržovat následující podmínky.

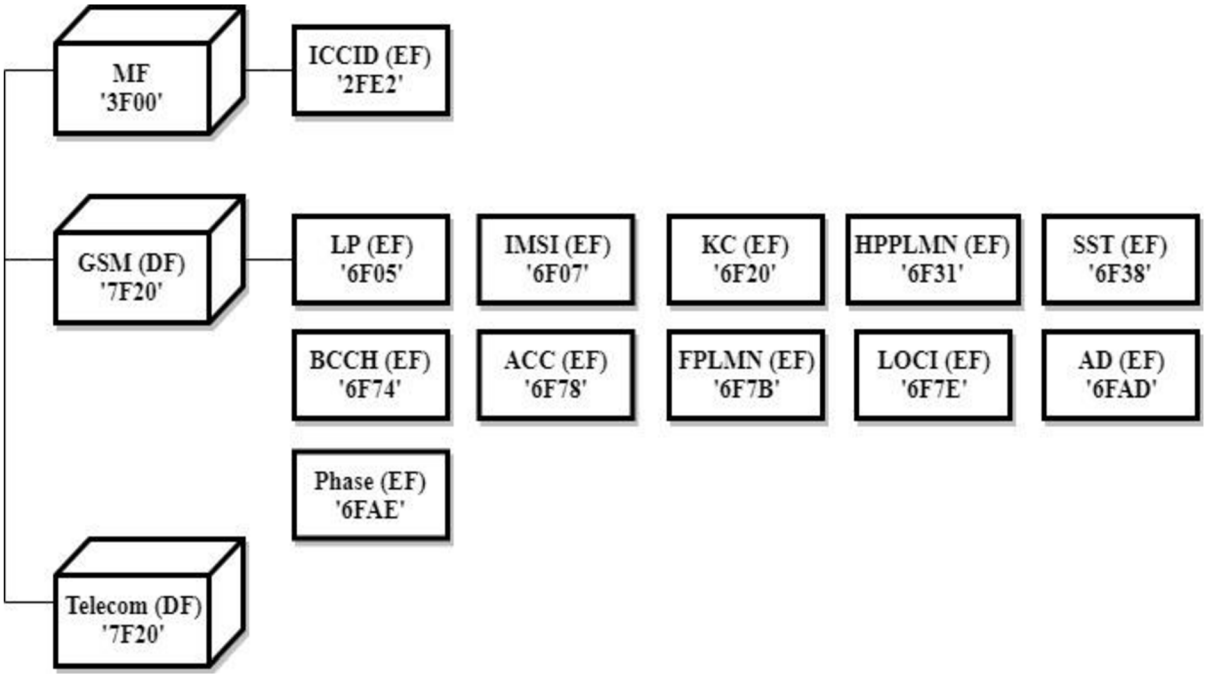
- Lze vybrat každý soubor a adresář nacházející se v aktuálním adresáři
- Lze přejít do nadřazeného adresáře aktuálního DF
- Kdykoliv můžete vybrat hlavní soubor/adresář MF

- Je možné znovu vybrat aktuální adresář

Pro pohyb uvnitř jednotlivých souborů slouží parametry příkazů.

3.2.4 SIM

V tomto bodě jsou k dispozici informace týkající se souborového systému technologie SIM. Je zde k dispozici i náhled na povinnou strukturu a její soubory. Kořenovým adresářem v této technologii je soubor s ID 3F00. V obrázku níže naleznete povinné adresáře a jejich soubory.

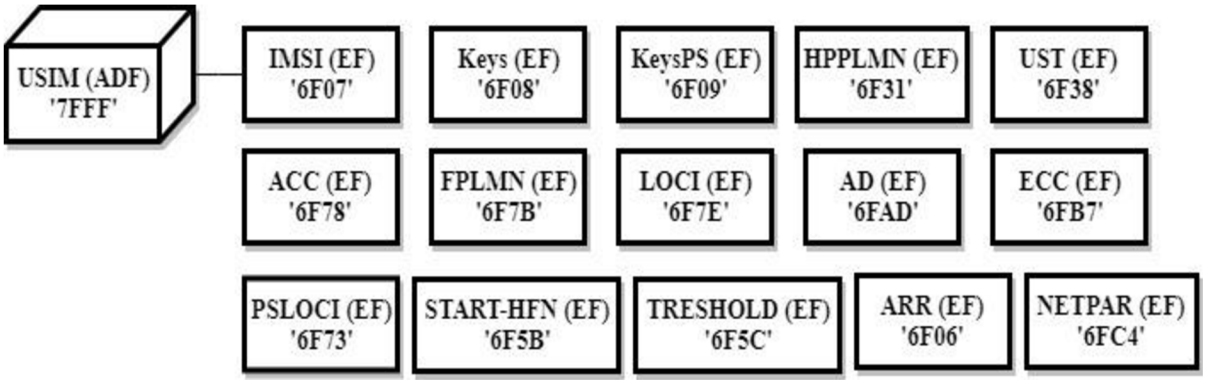


Obrázek 3.7: Povinná souborová struktura SIM [1]

Můžete si povšimnout, že povinných souborů není příliš. Některé adresáře jsou i bez souborů.

3.2.5 USIM

V této pasáži se budeme zabývat povinnou souborovou strukturou technologie USIM. Za kořenový adresář USIM se považuje soubor s ID 7FFF. Pro přístup k souborům uložených v tomto adresáři je třeba vybrat aplikaci obsluhující USIM souborový systém. Zde můžete vidět náhled na povinné soubory nacházející se pod tímto adresářem. Ke správnému běhu jsou třeba i povinné soubory z předchozího bodu.



Obrázek 3.8: Povinná souborová struktura USIM [17]

ADF – jedná se o DF vybrané aplikace (Application dedicated file)
Bližší informace o významu jednotlivých souborů naleznete níže v podkapitole 3.4.

3.3 Popis jednotlivých příkazů

Tato podkapitola se zabývá příkazy, které jsou podporovány v technologii SIM a USIM. Příkazy jsou dále vhodně rozděleny do skupin. Jednotlivé skupiny jsou k vidění v částech 3.3.1 až 3.3.11. Tam naleznete i popis a účel jednotlivých příkazů. Pro přehled byla vytvořena následující tabulka, která obsahuje všechny podporované příkazy i jejich formát.

Příkazy	CLA	INS	P1	P2	P3
SELECT	'A0'/'00'	'A4'	'00'	'00'	'02'
STATUS	'A0'/'00'	'F2'	'00'	'00'	délka
READ BINARY	'A0'/'00'	'B0'	od (offset high)	(offset low)	délka
UPDATE BINARY	'A0'/'00'	'D6'	od (offset high)	(offset low)	délka
READ RECORD	'A0'/'00'	'B2'	č. záznamu	mód	délka
UPDATE RECORD	'A0'/'00'	'DC'	č. záznamu	mód	délka
SEEK / SEARCH RECOR	'A0'/'00'	'A2'	'00'	typ/mód	délka
INCREASE	'A0'/'80'	'32'	'00'	'00'	'03'
VERIFY CHV	'A0'/'00'	'20'	'00'	č. CHV	'08'
CHANGE CHV	'A0'/'00'	'24'	'00'	č. CHV	'10'
DISABLE CHV	'A0'/'00'	'26'	'00'	'01'	'08'
ENABLE CHV	'A0'/'00'	'28'	'00'	'01'	'08'
UNBLOCK CHV	'A0'/'00'	'2C'	'00'	č. CHV	'10'
INVALIDATE / DEACTIVAT	'A0'/'00'	'04'	'00'	'00'	'00'
REHABILITATE / ACTIVATE	'A0'/'00'	'44'	'00'	'00'	'00'
RUN GSM ALGORITHM	'A0'	'88'	'00'	'00'	'10'
SLEEP	'A0'	'FA'	'00'	'00'	'00'
GET RESPONSE	'A0'/'00'	'C0'	'00'	'00'	délka
TERMINAL PROFILE	'A0'/'80'	'10'	'00'	'00'	délka
ENVELOPE	'A0'/'80'	'C2'	'00'	'00'	délka
FETCH	'A0'/'80'	'12'	'00'	'00'	délka
TERMINAL RESPONSE	'A0'/'80'	'14'	'00'	'00'	délka

Tabulka 3.3: Popis jednotlivých příkazů SIM/USIM [1][18]

Technologie USIM je rozšířená o následující příkazy.

AUTHENTICATE	'00'	'88'	'01'	mód	délka
MANAGE CHANNEL	'00'	'70'	'00'/'80'	'00'/kanál	'00'

Tabulka 3.4: Rozšířené příkazy technologie USIM [18]

Hodnota CLA:

- SIM 'A0'
- USIM '00', '80'

Technologie USIM je zpětně kompatibilní s technologií SIM.
Nebude-li blíže specifikováno fungování příkazu pro obě technologie SIM/USIM, předpokládá se stejná funkčnost u obou technologií. Nebude-li explicitně specifikováno jinak, veškeré hodnoty jsou v hexadecimální soustavě. Pro účel tohoto zadání, není třeba implementovat veškerou funkčnost v plném rozsahu. Ale pro přehled a možný budoucí vývoj zde mohou být uvedeny.

3.3.1 Příkaz umožňující pohyb po souborovém systému

Tato pasáž slouží ke specifikaci příkazu, který nám umožňuje pohybovat se po souborovém systému podle kapitoly 3.2.3. Jedná se o nejčastěji užívaný příkaz. Je známý pod jménem SELECT. Dovoluje procházení souborového systému například podle ID souboru. Po úspěšném provedení tohoto příkazu, je možné získat důležité informace týkající se stavu SIM/USIM karty, souboru a adresáře. Následně jsou uvedeny informace o formátu příkazu a struktury výstupu.

Formát příkazu

Příkaz	CLA	INS	P1	P2	P3
SELECT	'A0'/'00'	'A4'	'00'	'00'	'02'

Tabulka 3.5: Formát APDU příkazu SELECT [1][18]

Podrobné informace dostupné přes GET RESPONSE (pouze SIM):

- Pro MF, DF

Byte	Popis	Délka
1 - 2	RFU	2
3 - 4	dostupné místo v adresáři	2
5 - 6	ID	2
7	typ souboru	1
8 - 12	RFU	5
13	velikost od 14 bytu po konec odpovědi	1
14	hardwarové požadavky souboru	1
15	počet přímých podadresářů	1
16	počet přímých (vlastních) EF	1
17	počet CHV, PUK a administrativních kódů	1
18	RFU	1
19	stav CHV1	1
20	stav PUK 1	1
21	stav CHV 2	1
22	stav PUK 2	1
23	RFU	1
24 - 34	optimální, podle operátora	0-11

Tabulka 3.6: Struktura odpovědi na příkaz SELECT (MF,DF) [1]

- Pro EF

Byte	Popis	Délka
1 - 2	RFU	2
3 - 4	velikost souboru	2
5 - 6	ID	2
7	typ souboru	1
8	podmínka pro INCREACE, jinak RFU	1

9 - 11	bezpečnostní podmínky přístupu	3
12	stav souboru	1
13	velikost od 14 bytu po konec odpovědi	1
14	typ EF	1
15	délka záznamu	1

Tabulka 3.7: Struktura odpovědi na příkaz SELECT (EF) [1]

Pro bližší informaci navštivte specifikaci GSM [11.11] str. 38 – 40.

Specifikace pro technologii USIM:
Všechny podrobné informace týkající se daného souboru jsou odeslány ihned po úspěšném vykonání příkazu SELECT. Tyto informace jsou podrobnější než u SIM. Bohužel není význam jednotlivých bytů uveden v žádné zveřejněné specifikaci.

Umožňuje výběr souboru i podle cesty, potom P3 odpovídá velikosti cesty. Cesta se skládá z posloupnosti ID procházených souborů.

Parametr P2 musí mít hodnotu '04' a P1 označuje použitý mód.
Módy P1:

- '00' výběr souboru MF,DF,EF podle ID
- '01' výběr DF podle ID
- '04' výběr pomocí cesty od kořenového adresáře (bez MF ID)
- '05' výběr pomocí cesty od aktuálního DF (bez aktuálního DF ID)

3.3.2 Žádost o podrobné informace o aktuálním adresáři

Daný oddíl definuje příkaz, kterým je možné získat podrobné informace o aktuálním adresáři. Stejné informace lze získat i znovu vybráním adresáře pomocí výše uvedeného příkazu SELECT. Výhodou tohoto pokynu je, že nezmění hodnotu aktuálně vybraného EF souboru. Dále je uveden název a formát příkazu.

Formát příkazu

Příkaz	CLA	INS	P1	P2	P3
STATUS	'A0'/'00'	'F2'	'00'	'00'	délka

Tabulka 3.8: Formát APDU příkazu STATUS [1][18]

Parametr P3 specifikuje počet bytů, jenž ME vyžaduje. Musí být větší než 0 a menší rovno maximální délce odpovědi.

3.3.3 Příkazy pro práci s transparentními soubory

Uvedený úsek obsahuje příkazy, které nám umožňují pracovat s daty transparentních EF souborů. Jedná se o požadavek pro získání dat a modifikaci aktuálně vybraného souboru. K získávání dat nám slouží uvedený příkaz READ BINARY. Pod ním naleznete požadavek UPDATE BINARY, který se používá k modifikaci definovaného EF souboru.

Formát příkazu

Příkaz	CLA	INS	P1	P2	P3
READ BINARY	'A0'/'00'	'B0'	od (offset high)	(offset low)	délka

Tabulka 3.9: Formát APDU příkazu READ BINARY [1][18]

Jsou-li splněny bezpečnostní podmínky pro čtení dat vybraného souboru, vrací požadovaná data aktuálního EF o délce P3. Lze požadovat informace od 0xP1P2 bytu.

Formát příkazu

Příkaz	CLA	INS	P1	P2	P3
UPDATE BINARY	'A0'/'00'	'D6'	od (offset high)	(offset low)	délka

Tabulka 3.10: Formát APDU příkazu UPDATE BINARY [1][18]

Jsou-li splněny bezpečnostní podmínky pro úpravu dat vybraného souboru, nahradí/rozšíří obsah aktuálního EF. Data jsou nahrazeny/rozšířeny od 0xP1P2 bytu. Velikost připojeného datového řetězce je známá pomocí parametru P3.

3.3.4 Správa souborových záznamů

Všechny soubory neobsahují záznamy. Můžeme je nalézt pouze v cyklických a lineárních EF. Pro řízení a správu individuálních záznamů, nám slouží níže uvedené 4 příkazy. Ve formátu příkazu můžete nalézt jeho název a parametry. Pod ním se nachází jeho základní definice a ovládání.

Formát příkazu

Příkaz	CLA	INS	P1	P2	P3
READ RECORD	'A0'/'00'	'B2'	č. záznamu	mód	délka

Tabulka 3.11: Formát APDU příkazu READ RECORD [1][18]

Tato funkce je obdobou READ BINARY, slouží pro lineární a cyklické soubory ke čtení záznamů. Jsou-li splněny bezpečnostní podmínky pro čtení dat vybraného EF, jsou v odpovědi poslána data jednoho požadovaného záznamu o velikosti P3.

Po vybrání lineárního EF je hodnota ukazatele neznámá. Ukazatel cyklického souboru je nastaven na naposledy modifikovaný záznam.

Možnosti módů P2:

- '00' přečte aktuálně vybraný záznam
- '02' vrátí následující záznam
- '03' přečte předchozí záznam
- '04' vrátí záznam číslo P1 (nezmění hodnotu ukazatele)

Formát příkazu

Příkaz	CLA	INS	P1	P2	P3
UPDATE RECORD	'A0'/'00'	'DC'	č. záznamu	mód	délka

Tabulka 3.12: Formát APDU příkazu UPDATE RECORD [1][18]

Lze ho užít pro modifikaci zvoleného záznamu, jsou-li splněny přístupové podmínky nutné pro aktualizaci dat. Velikost připojeného datového řetězce naleznete v P3. Není-li vykonán úspěšně, hodnota ukazatele se nezmění.

Po vybrání lineárního EF je hodnota ukazatele neznámá. Ukazatel cyklického souboru je nastaven na naposledy modifikovaný záznam.

Módů P2:

- '00' přečte aktuálně vybraný záznam
- '02' vrátí následující záznam
- '03' přečte předchozí záznam
- '04' vrátí záznam číslo P1 (nezmění hodnotu ukazatele)

Formát příkazu:

Příkaz	CLA	INS	P1	P2	P3
SEEK	'A0'/'00'	'A2'	'00'	typ/mód	délka

Tabulka 3.13: Formát APDU příkazu SEEK [1][18]

Podle nastaveného způsobu hledání a požadovaného formátu odpovědi, je vyhledán záznam podle připojeného vzoru. Délka vzoru musí odpovídat parametru P3, nesmí přesáhnout velikost záznamu. Používá se pouze pro lineární EF.

Módy P2:

- 'X0' hledání od začátku do konce
- 'X1' zpětné hledání od konce
- 'X2' od následujícího záznamu vpřed (při nedefinovaném ukazateli od prvního z.)
- 'X3' od předchozího záznamu pozpátku (při nedefinovaném ukazateli od posledního z.)

Je-li X=='0' ukazatel je nastaven na nalezený záznam. Při X=='1' je aktualizován ukazatel i vrácen celý záznam.

Formát příkazu

Příkaz	CLA	INS	P1	P2	P3
INCREASE	'A0'/'80'	'32'	'00'	'00'	'03'

Tabulka 3.14: Formát APDU příkazu INCREASE [1][18]

Tento příkaz rozšíří poslední modifikovaný záznam cyklického EF o připojenou hodnotu. Rozšířený záznam je nastaven jako první a je na něj aktualizován ukazatel. Aby byl příkaz vykonán úspěšně, musí byt splněna odpovídající podmínka pro INCREASE a délka záznamu nesmí přesáhnout maximální povolenou hodnotu.

3.3.5 Správa autorizačních příkazů

Tato pasáž nám definuje 5 příkazů, kterými je možné řídit proces autorizace uživatele v mobilním telefonu. Jedná se o verifikaci, změnu bezpečnostního klíče, deaktivaci a reaktivaci potřeby verifikace. Poslední požadavek lze použít pro odblokování zablokovaného bezpečnostního klíče (PINU). Ve formátu příkazu naleznete jeho název a parametry. Pod ním se nachází jeho základní definice a ovládání.

Formát příkazu

Příkaz	CLA	INS	P1	P2	P3
VERIFY CHV	'A0'/'00'	'20'	'00'	č. CHV	'08'

Tabulka 3.15: Formát APDU příkazu VERIFY CHV [1][18]

Tato funkce slouží pro ověření CHV (PINU) s odpovídající bezpečnostní hodnotou uloženou na SIM. Po úspěšné autorizaci dojde k resetování počtu pokusů o ověření daného CHV.

Jestliže příkaz nedopadne úspěšně, počet zbývajících pokusů bude snížen o 1. Počet pokusů přetrvává i reset karty. Nezbývají-li již žádné pokusy, CHV je zablokován. Odblokování lze provést pomocí UNBLOCK CHV odpovídajícím PUK. Jednotlivé čísla CHV jsou v ascii hodnotě. Je-li jeho celková délka menší než 8, jsou ostatní číslice nahrazeny hodnotou 0xFF. Pro úspěšný průběh nesmí být požadovaný CHV již deaktivován či zablokován.

Formát příkazu

Příkaz	CLA	INS	P1	P2	P3
CHANGE CHV	'A0'/'00'	'24'	'00'	č. CHV	'10'

Tabulka 3.16: Formát APDU příkazu CHANGE CHV [1][18]

Následující příkaz slouží ke změně CHV uloženého na SIM/USIM kartě. Jednotlivé hodnoty CHV jsou v ascii hodnotě. Je-li délka CHV menší než 8, jsou ostatní číslice nahrazeny hodnotou 0xFF. V připojených datech na bytu 1-8 naleznete aktuální hodnotu CHV uloženou na SIM kartě. Ta slouží k autorizaci vlastníka. Aby bezpečnostní klíč nemohl být změněn neautorizovaným uživatelem.

Jestliže autorizace nedopadne úspěšně, počet zbývajících pokusů bude snížen o 1. Nezbývají-li již žádné pokusy, CHV je zablokován. Odblokování lze provést pomocí UNBLOCK CHV odpovídajícím

PUK. Na bytu 9-16 se nachází nová hodnota CHV ve stejném formátu. Pro úspěšný průběh nesmí být požadovaný CHV již deaktivován či zablokován.

Formát příkazu

Příkaz	CLA	INS	P1	P2	P3
DISABLE CHV	'A0'/'00'	'26'	'00'	'01'	'08'

Tabulka 3.17: Formát APDU příkazu DISABLE CHV [1][18]

Tato funkce může být použita pouze na CHV 1, který telefon obvykle vyžaduje při jeho zapnutí. Deaktivuje nutnost autorizace prováděnou daným bezpečnostním kódem. Po úspěšné autorizaci připojeným CHV dojde k resetování počtu pokusů o ověření daného CHV.

Jestliže příkaz nedopadne úspěšně, počet zbývajících pokusů bude snížen o 1. Počet pokusů přetrvává i restart karty. Nezbývají-li již žádné pokusy, CHV je zablokován. Odblokování lze provést pomocí UNBLOCK CHV odpovídajícím PUK. Jednotlivé čísla CHV jsou v ascii hodnotě. Je-li jeho celková délka menší než 8, jsou ostatní číslice nahrazeny hodnotou 0xFF. Pro úspěšný průběh nesmí být požadovaný CHV již deaktivován či zablokován.

Formát příkazu

Příkaz	CLA	INS	P1	P2	P3
ENABLE CHV	'A0'/'00'	'28'	'00'	'01'	'08'

Tabulka 3.18: Formát APDU příkazu ENABLE CHV [1][18]

Tento příkaz reaktivuje deaktivovaný CHV 1. Po úspěšné autorizaci připojeným aktuálním CHV dojde k resetování počtu pokusů o ověření daného CHV.

Jestliže příkaz nedopadne úspěšně, počet zbývajících pokusů bude snížen o 1. Počet pokusů přetrvává i restart karty. Nezbývají-li již žádné pokusy, CHV je zablokován. Odblokování lze provést pomocí UNBLOCK CHV odpovídajícím PUK. Jednotlivé čísla CHV jsou v ascii hodnotě. Je-li jeho celková délka menší než 8, jsou ostatní číslice nahrazeny hodnotou 0xFF. Pro úspěšný průběh nesmí být požadovaný CHV již aktivován či zablokován.

Formát příkazu

Příkaz	CLA	INS	P1	P2	P3
UNBLOCK CHV	'A0'/'00'	'2C'	'00'	č. CHV	'10'

Tabulka 3.19: Formát APDU příkazu UNBLOCK CHV [1][18]

Pomocí odpovídajícího PUK se pokusí odblokovat zablokovaný CHV. Při úspěchu dojde ke splnění odpovídající bezpečnostní podmínky a k restartu počtu pokusů o autorizaci daného CHV i PUK.

Jestliže příkaz nedopadne úspěšně, počet zbývajících pokusů o ověření tohoto PUK bude snížen o 1. Maximální počet pokusů je obvykle 10. Nezbývají-li již žádné pokusy, PUK je navždy zablokován. Jednotlivé čísla PUK jsou v ascii hodnotě. Lze použít i při nezablokovaném CHV.

3.3.6 Aktivace a reaktivace souborů

Daný oddíl definuje příkazy, kterými je možné spravovat soubory. Souborový systém umožňuje deaktivaci a následnou reaktivaci souborů. Níže je k dispozici název a formát těchto příkazů. Pod formátem naleznete bližší informace týkající se daného příkazu.

Formát příkazu

Příkaz	CLA	INS	P1	P2	P3
INVALIDATE	'A0'/'00'	'04'	'00'	'00'	'00'

Tabulka 3.20: Formát APDU příkazu INVALIDATE [1][18]

Funkce zruší platnost vybraného EF. Lze ji použít při splnění odpovídající bezpečnostní podmínky. Danému souboru je nastaven odpovídající příznak. Na soubor lze poté aplikovat pouze výše uvedený požadavek SELECT a nadcházející příkaz REHABILITATE. A ty příkazy, které EF explicitně umožňuje i v tomto stavu.

Formát příkazu

Příkaz	CLA	INS	P1	P2	P3
REHABILITATE	'A0'/'00'	'44'	'00'	'00'	'00'

Tabulka 3.21: Formát APDU příkazu REHABILITATE [1][18]

Tento příkaz lze použít jen při splnění odpovídající bezpečnostní podmínky. Dojde k reaktivaci vybraného EF. Souboru je odebrán příznak značící deaktivovaný stav.

3.3.7 Příkazy spojené s autentizací SIM/USIM karty v síti

V této sekci naleznete příkazy, které jsou nedílnou součástí autentizace SIM/USIM karty v GSM/UTMS síti. Bližší informace o autentizaci naleznete v podkapitole 3.5. Následně uvedené příkazy slouží pro vypočítání bezpečnostního klíče. Tento klíč poté mobilní telefon odešle do sítě, kde je srovnán se správnou hodnotou. První uvedený příkaz je využíván v SIM technologii. Druhý užívá technologie USIM. Ve formátu příkazu můžete nalézt název a parametry. Pod ním se nachází jeho základní definice a ovládání.

Formát příkazu

Příkaz	CLA	INS	P1	P2	P3
RUN GSM ALG.	'A0'	'88'	'00'	'00'	'10'

Tabulka 3.22: Formát APDU příkazu RUN GSM ALGORITHM [1]

Lze ji vykonat pouze z GSM adresáře (DF '7F20'), při úspěšné autorizaci užitím CHV 1. Pomocí přiloženého náhodného 16 bytového čísla a zabezpečeného autentizačního klíče KI, uloženého na SIM kartě. Vypočítá algoritmem A3 hodnotu SRES (Signed RESponse) o velikosti 4 byty. Algoritmem A8 spočítá i šifrovací klíč Kc o délce 8 bytů. Tyto údaje poté pošle v odpovědi do ME.

ME po obdržení odpovědi odešle spočítaný SRES k autentizaci. K šifrování přenášených dat použije algoritmus A5 a šifrovací klíč Kc.
Nepodporováno v technologii USIM.

Formát příkazu

Příkaz	CLA	INS	P1	P2	P3
AUTHENTICATE	'00'	'88'	'01'	mód	'délka'

Tabulka 3.23: Formát APDU příkazu AUTHENTICATE [18]

Tento příkaz má dvojí funkčnost.

- Jedním z nich je autentizace USIM karty v telefonní síti. Jedná se o bezpečnější způsob autentizace. Lze ji vykonat pouze z USIM aplikace, tu lze vybrat pomocí příkazu SELECT (ADI 'A0000000871002FF47F00189000001FF').

  Na vstupu se očekává RAND a AUTN (Authentication token) v uvedeném pořadí. Aby se jednotlivé hodnoty nesmíchaly, musí být před každou hodnotou 1 byte značící její velikost. AUTN se skládá z více hodnot a to konkrétně SQN | AK (anonymity key), MODE, MAC.

  Výstupem jsou kódy RES (response), CK (cipher key) a IK (integrity key). Před každou hodnotou je 1 byte značící její velikost.
- Druhým je resynchronizace sloužící k synchronizaci globálního času a potvrzovacích kódů. Ty se využívají při odesílání a přijímání dat.

  Při úspěšném vykonání vrátí naposledy přijaty RAND a AUTN (Authentication token) v uvedeném pořadí. Aby se jednotlivé hodnoty nesmíchaly, je před každou hodnotou 1 byte značící její velikost.

Nepodporováno v technologii SIM.

Módy P2:

- '01' autentizace pomocí přepínače obvodů (circuit switching)
- '02' autentizace pomocí přepínače paketů (packet switching)
- '11' resynchronizace pomocí přepínače obvodů (circuit switching)
- '12' resynchronizace pomocí přepínače paketů (packet switching)

Přepínače obvodů a paketů jsou dva způsoby odesílání a přijímání dat. U prvního způsobu musejí všechny data znát úplnou adresu. Druhý způsob využívá směrování paketů (postačí znát cílovou destinaci).

3.3.8 Zastaralé příkazy

V tomto oddíle naleznete zastaralé příkazy technologie SIM, které se využívali pouze v prvních SIM kartách fáze 1. Uvedený příkaz se používal k šetření energie. Uspal kartu až do příští komunikace.

Formát příkazu

Příkaz	CLA	INS	P1	P2	P3
SLEEP	'A0'	'FA'	'00'	'00'	'00'

Tabulka 3.24: Formát APDU příkazu SLEEP [1]

Jedná se o zastaralý příkaz SIM karty fáze 1 k uspání. Novější fáze ji nesmějí používat.
Nepodporováno v technologii USIM.

3.3.9 Příkaz pro získání dodatečných dat

Nadcházející pasáž obsahuje podrobné informace o příkazu, který se využívá k získání dodatečných dat. Lze pomocí něj získat data generované předchozími příkazy.

Formát příkazu

Příkaz	CLA	INS	P1	P2	P3
GET RESPONSE	'A0'/'00'	'C0'	'00'	'00'	délka

Tabulka 3.25: Formát APDU příkazu GET RESPONSE [1][18]

Dodatečné data jsou generovány například při použití příkazů RUN GSM ALGORITHM, SEEK, SELECT, INCREASE. Není-li o data zažádáno ihned a jsou přepsány jinými, může být navrácen příznak značící technické problémy. Lze zažádat i o část přítomných dat o délce P3. Nebude-li o data zažádáno ihned, SIM bude vracet návratový kód č. 2 místo č. 1. Dokud nedojde k přijetí požadavku o data. Návratové kódy jsou dostupné v příloze D.

3.3.10 Příkazy pro obsluhu Sim Application Toolkit

V této sekci naleznete příkazy, které mají něco společného se Sim Application Toolkit. Úvodní příkaz obeznámí mobilní telefon o podporovaných službách. Dále zde máme příkaz pro odeslání dat do Sim Application Toolkit. Následuje požadavek, kterým je možno zasílat příkazy ze SIM/USIM karty do telefonu. Naposledy zde máme uveden požadavek pro získání odpovědi na předcházející zaslaný příkaz. Podrobnější informace jsou uvedeny následovně.

Formát příkazu

Příkaz	CLA	INS	P1	P2	P3
TERMINAL PROFILE	'A0'/'80'	'10'	'00'	'00'	délka

Tabulka 3.26: Formát APDU příkazu TERMINAL PROFILE [1][18]

Tímto příkazem obeznámí ME přítomnou SIM kartu o podporovaných funkcích Sim Application Toolkit. Připojené data mají délku P3.

Po úspěšném vykonání, jsou generována data, která obsahují informace o podporované funkčnosti na straně SIM. Ty jsou dostupné pomocí FETCH. O velikosti dat je ME informováno v jednoduché odpovědi.

Formát příkazu

Příkaz	CLA	INS	P1	P2	P3
ENVELOPE	'A0'/'80'	'C2'	'00'	'00'	délka

Tabulka 3.27: Formát APDU příkazu ENVELOPE [1][18][3]

Touto funkcí ME odešle přiřazená data o velikosti P3 do Sim Application Toolkit.
Podrobnější informace jsou k dispozici v GSM 11.14.

Formát příkazu

Příkaz	CLA	INS	P1	P2	P3
FETCH	'A0'/'80'	'12'	'00'	'00'	délka

Tabulka 3.28: Formát APDU příkazu FETCH [1][18][3]

Slouží pro odeslání proaktivního příkazu Sim Application Toolkit ze SIM do ME. Telefon se v časových intervalech pomocí tohoto příkazu ptá, zdali je něco nového.
Podrobnější informace jsou k dispozici v GSM 11.14.

Formát příkazu

Příkaz	CLA	INS	P1	P2	P3
TERMINAL RESPONSE	'A0'/'80'	'14'	'00'	'00'	délka

Tabulka 3.29: Formát APDU příkazu TERMINAL RESPONSE [1][18]

Tímto příkazem odešle ME dopověď na předchozí příkaz, který obdržel ve FETCH. Délka přiložených dat je uložena v P3.

3.3.11 Správa logických kanálů

Tento bod obsahuje data a příkazy sloužící pro řízení logických kanálů, které nejsou podporovány v technologii SIM. Příslušné informace jsou k vidění pod tímto textem.

Formát příkazu

Příkaz	CLA	INS	P1	P2	P3
MANAGE CHANNEL	'00'	'70'	'00'/'80'	'00'/kanál	'00'

Tabulka 3.30: Formát APDU příkazu MANAGE CHANNEL [18]

Tato funkce slouží k otevírání a uzavírání logických kanálů, které slouží ke komunikaci s USIM kartou. Otevřené kanály mohou komunikovat i s různými aplikacemi, které se nacházejí na USIM kartě. Při žádosti o otevření nového logického kanálu je v odpovědi navráceno číslo reprezentující daný kanál. Jehož velikost je 1 byte.

Mody P1:

- '00' žádost o otevření nového logického kanálu
- '80' žádost o uzavření otevřeného logického kanálu v P2

3.4 Popis povinných souborů

Tato podkapitola obsahuje informace týkající se povinných souborů souborového systému. Podkapitola je složena ze dvou pasáží. V první pasáži naleznete povinné soubory technologie SIM a bližší informace týkající se těchto EF souborům. Ve druhé jsou k dispozici nezbytné soubory, které naleznete pouze v technologii USIM.

Jak jste si již mohli všimnout v bodech 3.2.4 a 3.2.5. Některé soubory se jmenují stejně v technologii SIM i USIM. A mají i podobný nebo stejný význam. Proto budou následně uvedeny pouze jednou.

3.4.1 Povinné soubory technologie SIM

V bodu číslo 3.4.1 naleznete bližší informace o povinných souborech, které naleznete v technologii SIM i USIM. Individuální soubory a informace týkající se jejich obsahu a účelu jsou k dispozici v odrážkách níže. Uvedené odrážky obsahují základní informace o obsahu daného EF souboru. K vidění jsou i příslušné přístupové podmínky. Tyto podmínky naleznete napravo od základní definice specifikovaného EF souboru.

- **EF_{ICCID} (2FE2)**

Obsahuje unikátní identifikační číslo SIM karty o velikosti 10 bytů.

Typ souboru: Transparentní

Přístupové podmínky	
READ	vždy
UPDATE	nikdy
INVALIDATE	ADM
REHABILITATE	ADM

Tabulka 3.31: Přístupové podmínky ICCID [1]

- **EF_{LP} (6F05)**

Obsahuje kódy podporovaných jazyků seřazených podle priority. Každé číslo reprezentující jazyk má velikost 1 byte.

Typ souboru: Transparentní

Přístupové podmínky	
READ	vždy
UPDATE	CHV1
INVALIDATE	ADM
REHABILITATE	ADM

Tabulka 3.32: Přístupové podmínky LP [1]

- **EF_{IMSI} (6F07)**

Daný soubor obsahuje hodnotu IMSI (International Mobile Subscriber Identity) o velikosti 8 bytů, sloužící k dohledání detailů o uživateli na straně operátora. Celková velikost souboru je 9 bytů, protože první byte obsahuje velikost uložené hodnoty.

Typ souboru: Transparentní

Přístupové podmínky	
READ	CHV1
UPDATE	ADM
INVALIDATE	ADM
REHABILITATE	CHV1

Tabulka 3.33: Přístupové podmínky IMSI [1]

- **EF_{KC} (6F20)**

Je v něm uložen šifrovací klíč Kc (Ciphering Key) o velikosti 8 bytů. A sekvenční číslo n s velikostí 1 byte vyjadřující stav klíče. Hodnota n je před autentizací v GSM síti rovna hodnotě 0x07 (klíč je nedostupný). Hodnota Kc se generuje při každé nové autentizaci, bližší informace v kapitole 3.5.

Typ souboru: Transparentní

Přístupové podmínky	
READ	CHV1
UPDATE	CHV1
INVALIDATE	ADM
REHABILITATE	ADM

Tabulka 3.34: Přístupové podmínky KC [1]

- **EF_{HPPLMN} (6F31)**

V daném souboru se nachází hodnota n ovlivňující časový interval pro hledání HPLMN (Home public land mobile network). Jedná se o vyhledávání přístupních bodů sítě. Následující vyhledávání se poté nastaví na n*konstanta. Je-li n == 0 nedochází k vyhledávání.

Typ souboru: Transparentní

Přístupové podmínky	
READ	CHV1
UPDATE	ADM
INVALIDATE	ADM
REHABILITATE	ADM

Tabulka 3.35: Přístupové podmínky HPPLMN [1]

- **EF_{SST} (6F38)**

Tento EF obsahuje informace o službách dostupných a aktivních v dané SIM kartě (SIM service table). Každá služba je reprezentována jedním bytem. Podporuje-li karta danou službu, liché bity nabývají hodnoty 1. Je-li služba aktivní, nabývají hodnotu 1 i sudé bity.

Pro bližší informace o jednotlivých službách a jejich pozici v souboru si vyhledejte následující specifikaci: GSM [11.11] strana 56.

Typ souboru: Transparentní

Přístupové podmínky	
READ	CHV1
UPDATE	ADM
INVALIDATE	ADM
REHABILITATE	ADM

Tabulka 3.36: Přístupové podmínky SST [1]

- **EF_{BCCH} (6F74)**

Poskytuje informace ovlivňující BCCH (Broadcast control channels) o velikosti 16 bytů. Slouží například k redukci rozsahu vyhledávání v mobilním telefonu, pokoušejícího se vyhledat přístupové buňky (body) GSM (mobilní) sítě.

Typ souboru: Transparentní

Přístupové podmínky	
READ	CHV1
UPDATE	CHV1
INVALIDATE	ADM
REHABILITATE	ADM

Tabulka 3.37: Přístupové podmínky BCCH [1]

- **EF_{ACC} (6F78)**

V tomto souboru naleznete 15 přiřazených tříd pro řízení přístupu (Access control class). Spravující a kontrolující jednotlivé pokusy o přístup. Všechny třídy jsou zastoupeny 2 byty, každá využívající 1 bit. Podrobnější informace naleznete ve specifikaci GSM [02.11].

Typ souboru: Transparentní

Přístupové podmínky	
READ	CHV1
UPDATE	ADM
INVALIDATE	ADM
REHABILITATE	ADM

Tabulka 3.38: Přístupové podmínky ACC [1]

- **EF_{FPLMN} (6F7B)**

Tento EF obsahuje kódování pro čtyři tzv. zakázané PLMN (public land mobile network). Tyto informace jsou požadovány ME v inicializační části. Označují ty PLMN, ke kterým se zařízení nemá pokoušet připojovat automaticky. Každý je reprezentován hodnotami MCC (Mobile Country Code) a MNC (Mobile Network Code) uložených na 3 bytech.

Typ souboru: Transparentní

Přístupové podmínky	
READ	CHV1
UPDATE	CHV1
INVALIDATE	ADM
REHABILITATE	ADM

Tabulka 3.39: Přístupové podmínky FPLMN [1]

- **EF_{LOCI} (6F7E)**

Jsou v něm uloženy lokalizační informace a to konkrétně aktualizační stav zastupovaný 1 bytem, LAI (Location Area Information) unikátní identifikátor PLMN o velikosti 5 bytů a TMSI (Temporary Mobile Subscriber Identity) s velikostí 4 byty. TMSI je náhodné číslo, které je přiřazeno mobilnímu telefonu ihned po jeho zapnutí. Pomáhá s lokalizací telefonu. Při vstupu do nové oblasti musí být aktualizováno.

Typ souboru: Transparentní

Přístupové podmínky	
READ	CHV1
UPDATE	CHV1
INVALIDATE	ADM
REHABILITATE	CHV1

Tabulka 3.40: Přístupové podmínky LOCI [1]

- **EF_{AD} (6FAD)**

Obsahuje provozní režim reprezentován 1 bytem podle typu dané SIM karty. Může se jednat o režim normální, schválení (umožňuje specifické použití ME), buněčné testování (otestuje buňku sítě před jejím použitím), specifikovaný výrobcem. Podrobnější informace jsou dostupné ve specifikaci GSM [11.11] str.64.

Typ souboru: Transparentní

Přístupové podmínky	
READ	vždy
UPDATE	ADM
INVALIDATE	ADM
REHABILITATE	ADM

Tabulka 3.41: Přístupové podmínky AD [1]

- **EF_{PHASE} (6FAE)**

Obsahuje fázi dané SIM karty na 1 bytu.

- Fáze 1 (00)
  Umožňuje jen základní funkčnost.
- Fáze 2 (02)
  Umožňuje službu FDN (Fixed Dialing Number). Je-li aktivována, umožňuje volání jen známým předčíslím (např. +420). A AoC (Advice of Charge) poskytující informace o poplatcích za služby.
- Fáze 2+ (03)
  Podporuje SIM Application Toolkit a požaduje vykonání příkazu PROFILE DOWNLOAD.

Typ souboru: Transparentní

Přístupové podmínky	
READ	vždy
UPDATE	ADM
INVALIDATE	ADM
REHABILITATE	ADM

Tabulka 3.42: Přístupové podmínky PHASE [1]

Pro bližší informace o jednotlivých souborech SIM si vyhledejte následující specifikaci: GSM [11.11]. Odkaz na tuto specifikaci naleznete i v literatuře.

3.4.2 Povinné soubory vyskytující se pouze v technologii USIM

V bodu číslo 3.4.2 naleznete bližší informace o povinných souborech, které naleznete pouze v technologii USIM. Jednotlivé soubory a informace týkající se jejich obsahu jsou k dispozici v odrážkách níže. Tyto odrážky obsahují základní informace o obsahu specifikovaného EF souboru. K vidění jsou i příslušné přístupové podmínky povinných souborů. Tyto podmínky naleznete napravo od jejich základní definice.

- **EF_{Keys} (6F08)**

Naleznete v něm šifrovací klíče CK (Ciphering key 16 bytes), IK (Integrity key 16 bytes), KSI (Key set identifier 1 byte). Využívají se k šifrování v USIM technologii.

Typ souboru: Transparentní

Přístupové podmínky	
READ	CHV1
UPDATE	CHV1
INVALIDATE	ADM
REHABILITATE	ADM

Tabulka 3.43: Přístupové podmínky Keys [17]

- **EF_{KeysPS} (6F09)**

Tento EF slouží k uschování šifrovacího klíče CKPS (Ciphering key PS 16 bytes), klíče integrity IKPS (Integrity key 16 bytes), identifikátoru sady klíčů KSIPS (Key set identifier PS 1 byte) pro paketovou komunikaci pro USIM.

Typ souboru: Transparentní

Přístupové podmínky	
READ	CHV1
UPDATE	CHV1
INVALIDATE	ADM
REHABILITATE	ADM

Tabulka 3.44: Přístupové podmínky KeysPS [17]

- **EF_{UST} (6F38)**

Nachází se v něm seznam aktivních služeb pro danou USIM. Každá služba je uložena v 1 bytu.

Typ souboru: Transparentní

Přístupové podmínky	
READ	CHV1
UPDATE	ADM
INVALIDATE	ADM
REHABILITATE	ADM

Tabulka 3.45: Přístupové podmínky UST [17]

- **EF_{ECC} (6FB7)**

Soubor se seznamem tísňových volání pro USIM. Každé číslo je uloženo ve 3 bytech.

Typ souboru: Lineárně pevný

Přístupové podmínky	
READ	vždy
UPDATE	ADM
INVALIDATE	ADM
REHABILITATE	ADM

Tabulka 3.46: Přístupové podmínky ECC [17]

- **EF_{PSLOCI} (6F73)**

Obsahuje informace potřebné pro směrování paketů v síti. Jeho celková velikost je 14 bytů. Nachází se pouze v USIM.

Typ souboru: Transparentní

Přístupové podmínky	
READ	CHV1
UPDATE	CHV1
INVALIDATE	ADM
REHABILITATE	ADM

Tabulka 3.47: Přístupové podmínky PSLOCI [17]

- **$EF_{START\text{-}HFN}$ (6F5B)**

Jeho data se používají ke kontrole životnosti klíčů v USIM.

Typ souboru: Transparentní

Přístupové podmínky	
READ	CHV1
UPDATE	CHV1
INVALIDATE	ADM
REHABILITATE	ADM

Tabulka 3.48: Přístupové podmínky START-HFN [17]

- **$EF_{TRESHOLD}$ (6F5C)**

Obsahuje maximální hodnotu pro $EF_{START\text{-}HFN}$.

Typ souboru: Transparentní

Přístupové podmínky	
READ	CHV1
UPDATE	ADM
INVALIDATE	ADM
REHABILITATE	ADM

Tabulka 3.49: Přístupové podmínky TRESHOLD [17]

- **EF_{ARR} (6F06)**

Tento EF obsahuje pravidla přístupu pro soubory umístěné v USIM ADF. Pravidla jsou uloženy formou záznamů.

Typ souboru: Lineárně pevný

Přístupové podmínky	
READ	vždy
UPDATE	ADM
INVALIDATE	ADM
REHABILITATE	ADM

Tabulka 3.50: Přístupové podmínky ARR [17]

- **EF_{NETPAR} (6FC4)**

Naleznete v něm informace o frekvencích známých buněk sítě.

Typ souboru: Transparentní

Přístupové podmínky	
READ	CHV1
UPDATE	CHV1
INVALIDATE	ADM
REHABILITATE	ADM

Tabulka 3.51: Přístupové podmínky NETPAR [17]

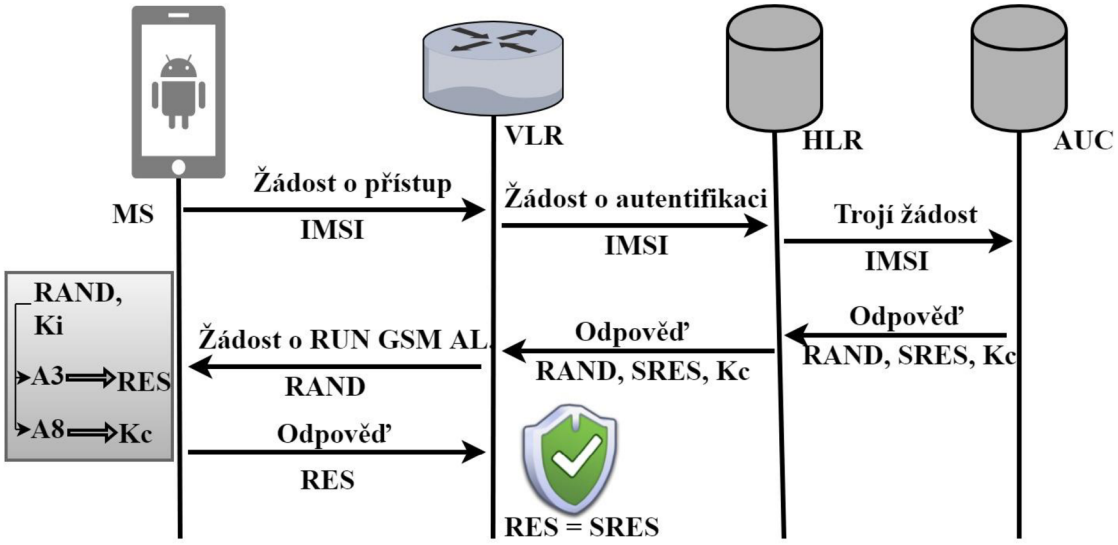
Bližší informace o individuálních EF souborech naleznete ve specifikaci TS [131 102].

3.5 Autentizace a její zabezpečení

Tato část se zabývá průběhem autentizace technologií SIM a USIM. Dozvíte se zde, jak jsou žádosti o autentizaci vyřizovány v GSM/UMTS síti. Můžete si zde prohlédnout, jaké klíče a hodnoty se na této činnosti podílejí. Bližší informace o autentizaci jednotlivých technologiích naleznete v následujících oddílech.

3.5.1 SIM

V tomto oddíle se budeme zabývat zabezpečením technologie SIM. Hlavně se zaměříme na její autentizaci v GSM síti. K dispozici jsou i detaily o užitých bezpečnostních klíčích. Pro pochopení průběhu autentizace poslouží uvedená fotografie.



Obrázek 3.9: Autentizace SIM [1][4]

Z fotografie vyplívá, že žádost o autentizaci je zahájena ze strany mobilního telefonu. Mobilní telefon bude dále označován jako mobilní stanice (MS). Mobilní stanice zažádá o přístup do sítě zahájením autentizace. Žádost je odeslána do VLR (Visitor Location Register), jedná se o návštěvnický registr účastníků sítě. Ten ji přepošle do HLR (Home Location Register). Zastupuje centrální databázi v mobilní síti. Následně je žádost odeslána do autorizačního střediska AUC. Zde jsou spočítány příslušné bezpečnostní údaje, které jsou poté navráceny až do VLR. Registr dále požádá o autentizaci MS. Obdrží výsledný bezpečnostní klíč, který porovná s originálem. Pro bližší informace o individuálních krocích autentizace a užitých bezpečnostních prvcích, nahlédněte níže.

Bezpečnostní prvky:

- UICCID
  Unikátní sériové číslo SIM. Je k dispozici vždy.
- IMSI
  Sloužící k dohledání detailů o uživateli u operátora. Vyžaduje autorizaci pomocí CHV1.
- KI
  Má velikost 16 bytů. Reprezentuje autentizační klíč. Nelze ho nikdy získat přímo.
- Kc
  Užívá se k šifrování komunikace mezi ME a telefonní GSM sítí. Vytvoří se při zavolání RUN GSM ALGORITHM pomocí algoritmu A8 a náhodného čísla.
- SRES

Je vypočítán pomocí funkce RUN GSM ALGORITHM. Má velikost 4 byty. Vypočítá se algoritmem A3. Pro výpočet se používá klíč KI a náhodně vygenerované číslo RAND, které původně pochází z AUC.

- A3 a A8 algoritmus
  Je známý i pod názvem COM128. Tento algoritmus je již známý veřejnosti.

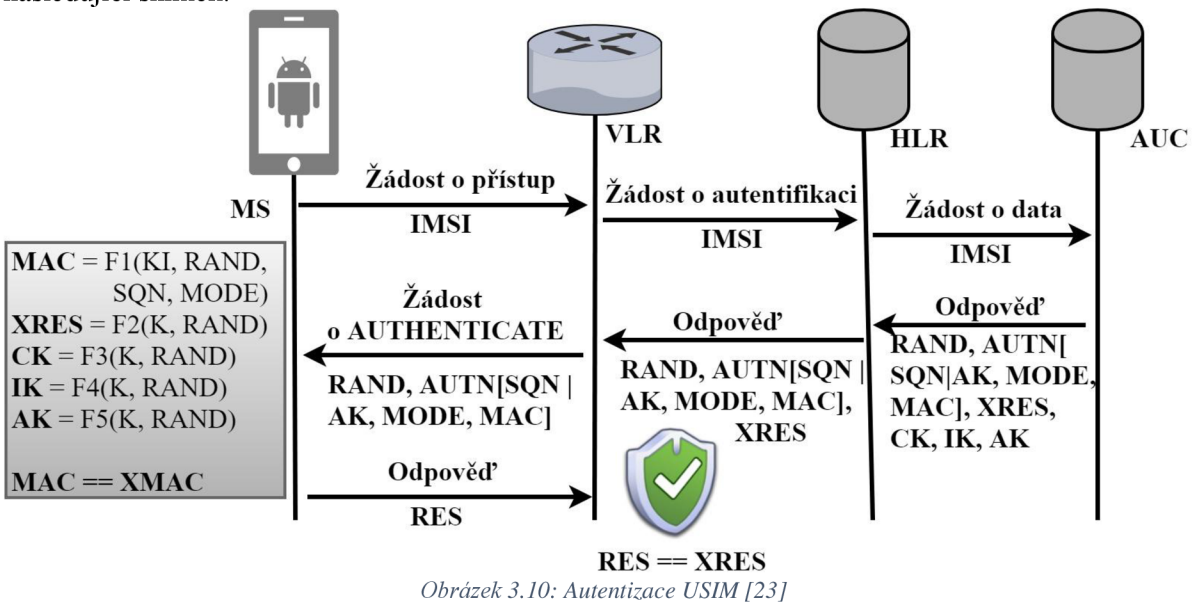
Jednotlivé kroky autentizace:

1. ME přečte UICCID
2. ME získá IMSI
3. MS (Mobile station) požádá o přístup do GSM sítě zasláním IMSI do VLR (visitor location register)
4. Centrum pro autentizaci v GSM síti spočítá KI, SRES a pošle je návštěvnickému registru (VLR), společně s použitým vygenerovaným číslem RAND
5. VLR zašle obdržený RAND do MS
6. MS použije ME k zaslání příkazu RUN GSM ALGORITHM do SIM, na vstupu je obdržený RAND
7. Výsledná hodnota SRES je odeslána do VLR, tam následně dojde k ověření hodnoty s originálem

V dubnu 1998 se skupině kryptologů povedlo objevit metodu pro extrakci KI klíče uloženého na SIM kartě. A to za pomocí opakování příkazu RUN GSM ALGORITHM (přibližně 150 000 krát).

3.5.2 USIM

V tomto oddíle se budeme zabývat zabezpečením technologie USIM. Převážně se zaměříme na její autentizaci v síti a užitých bezpečnostních klíčů. Pro zjednodušený příklad autentizace poslouží následující snímek.



Obrázek 3.10: Autentizace USIM [23]

Na počátku autentizace odešle mobilní stanice, reprezentující mobilní telefon, žádost o autentizaci. Příslušná žádost je odeslána do VLR (Visitor Location Register), jedná se o návštěvnický registr účastníků v mobilní síti. Ten tento požadavek obdrží a přepošle do HLR (Home Location Register), který nám zastupuje centrální databázi v mobilní síti. Obsahuje i detailní informace o každém účastníkovi, který je autorizovaný pro GSM/UMTS síť. Následně zažádá o příslušné informace autentizační středisko AUC (Authentication Center). V tomto středisku se použitím náhodně vygenerovaného 16 bytového čísla spočítají autentizační hodnoty. Tyto hodnoty jsou dále přeposlány až do VLR. Ten dále zašle žádost o autentizaci do mobilní stanice. Obdrží odpověď a tu porovná

s původní hodnotou. Jsou-li si hodnoty rovny, autentizace je dokončena. Jednotlivé kroky autentizace a informace týkající se bezpečnostních prvků, budou uvedeny následovně.

Bezpečnostní prvky:

- UICCID
  Unikátní sériové číslo USIM. Je k dispozici vždy.
- IMSI
  Sloužící k dohledání detailů o uživateli u operátora. Vyžaduje autorizaci pomocí CHV1.
- KI
  Jedná se o autentizační klíč. Má velikost 16 bytů. Nelze ho nikdy získat přímo.
- CK
  Užívá se k šifrování komunikace mezi ME a telefonní GSM sítí. Vytvoří se při zavolání AUTHENTICATE pomocí algoritmu, který budeme označovat F3. Tento algoritmus potřebuje znát i hodnotu Ki a příslušné náhodné číslo.
- RES
  Je vypočítán pomocí funkce AUTHENTICATE. Vypočítá se algoritmem, který budeme nazývat F2. Tento algoritmus použije náhodně vygenerované číslo RAND, které původně pochází ze střediska AOC.
- MAC (Message Authentication Code)
  Kód sloužící k ověření platnosti zprávy. Počítá se z SQN, RAND, MODE. Způsob není znám.
- IK (integrity key)
  Získává se z RAND. Zajišťuje integritu přenášených dat. Použitý algoritmus není znám.
- AK (anonymity key)
  Získává se z RAND. Jeho velikost je 6 bytů. Spočítá se algoritmem označovaným jako F5. Tento algoritmus není zveřejněn.
- AUTN
  Používá se pro zjednodušení. Obsahuje SQN | AK (anonymity key), MODE, MAC.
- F3 a F2 algoritmy
  Tyto algoritmy jsou pro správný průběh autentizace nejdůležitější. Momentálně je jejich implementace nezveřejněna.

Jednotlivé kroky autentizace:

1. ME přečte UICCID
2. ME získá IMSI
3. MS (Mobile station) požádá o přístup do telefonní sítě zasláním IMSI do VLR (visitor location register)
4. Centrum pro autentizaci v telefonní síti vygeneruje RAND a spočítá MAC, RES, CK, IK a AK. Následně vypočítané hodnoty přepošle návštěvnickému registru VLR, společně s použitým náhodně vygenerovaným 16 bytovým číslem (RAND)
5. VLR zašle obdržený RAND a AUTN do MS
6. MS použije ME k zaslání příkazu AUTHENTICATE do SIM, na vstupu je obdržený RAND a AUTN
7. USIM vypočítá MAC a AK a porovná je se vstupem.
8. Jeli hodnota SQN dostatečně čerstvá (nová), vypočítá RES, CK a IK .
9. Získanou hodnotu RES odešle VLR a tam dojde k ověření hodnot

O výsledku autentizace je mobilní stanice vhodně informována.

3.6 Rozdíly mezi SIM a USIM

Pojďme si shrnout základní rozdíly mezi technologiemi SIM a USIM. K tomuto účelu nám poslouží nadcházející obrázek.

Parametry	SIM	USIM
Třída	CLA = 'A0'	CLA = '00/80'
Kořenový adresář	'3F00'	'7FFF'
Autentifikace	RUN GSM ALG.	AUTHENTICATE
Sim Application Toolkit	ano/ne	ano
bezpečnostní algoritmy	známé	neznámé
podpora více logických kanálů	ne	ano
spuštěna	1991	2002-2004
vývoj	zastaven	probíhá

Obrázek 3.11: Základní rozdíly mezi SIM a USIM [38][1][18]

Jak už to tak bývá, čím novější technologie, tím je dán větší důraz na bezpečnost. Novější algoritmy používané pro autentizaci v telefonní síti, jsou dosud nezveřejněny. I délky používaných klíčů expandují.

Z hlediska složitosti je USIM mnohem výše, než SIM. Její struktura je propracovanější a obsahuje mnohem více souborů. Umožňuje ukládat více informací do jednotlivých adresářů. Může obsahovat i informace týkající se bankovnictví, ukládat MMS na kartu atd. Lze ji využít v novějších sítích, jakou je například LTE.

4 Návrh řešení

V této kapitole se budeme zabývat různými způsoby řešení našeho problému. Vybereme si nejvhodnější řešení a to následně implementujeme. V případě, že žádné řešení nebude možné realizovat, budeme dále pokračovat pomocí simulace.

Nevhodnější by bylo, kdyby se nám povedlo, nahrát vlastní applet na skutečnou SIM/USIM kartu. Ale na originální SIM/USIM kartu nelze jednoduše nahrávat neautorizované applety, neznáme potřebné bezpečnostní klíče.

Pro realizaci nápadu, v kterém bude SIM/USIM karta využita jako bezpečnostní element, by bylo proto vhodné vytvořit napodobeninu SIM karty podporující GSM funkčnost.

4.1 Způsoby řešení

Zde naleznete způsoby, které lze teoreticky a snad i prakticky využít k řešení našeho problému.

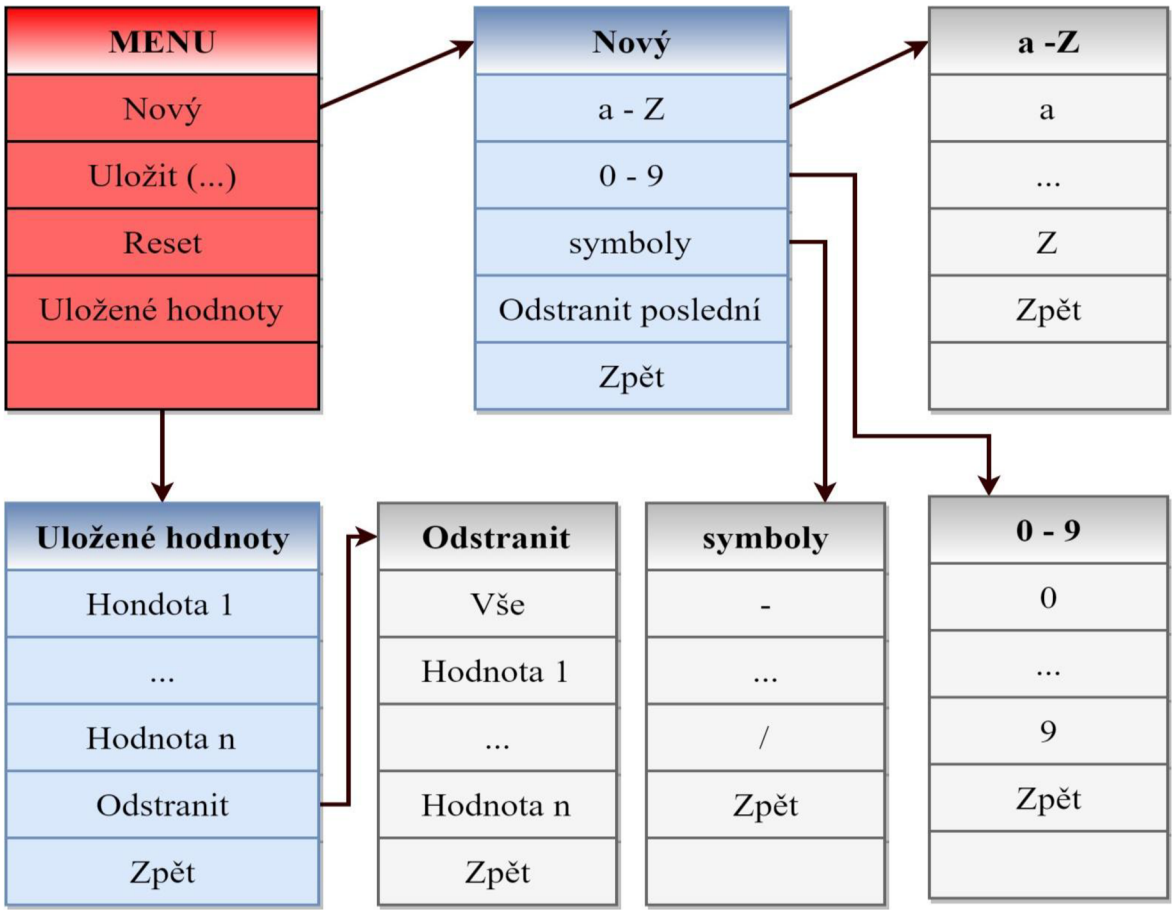
4.1.1 Pomocí menu

Pomocí Sim Application Toolkit, je možné vytvořit výběrové menu na SIM/USIM. Toto menu je odesláno do ME použitím proaktivních příkazů. Proaktivní příkazy jsou realizovány pomocí příkazu FETCH, jedná se o příkazy odeslané ze SIM do ME.

Aby tento způsob byl realizovatelný, musí SIM i MS podporovat proaktivní příkazy. Tato funkčnost, je až na výjimky, podporována všemi zařízeními. Mezi výjimky patří například nejstarší typ karet (fáze 1).

Návrh implementace

Tento návrh se skládá z více stupňového menu. Umožňuje ukládat i získávat uložené hodnoty. Zvolená hodnota menu je odeslána v odpovědi do SIM/USIM. Karta odešle menu, které se má momentálně zobrazovat. Toto menu je k dispozici v nastavení telefonu. Pro zobrazení menu lze využít i externí aplikace. Návrh pokračuje na další straně.



Obrázek 4.1: Návrh řešení pomocí menu

Obrázek výše obsahuje menu, které jsme navrhli jako komunikační prostředek mezi mobilním telefonem a SIM/USIM kartou. Na začátku komunikace je odesláno červené menu. To se dále dělí na další výběrové tabulky. Funkčnost jednotlivých položek všech tabulek je vyjádřena jejich jménem. Mezi jednotlivými tabulkami lze jednoduše přecházet pomocí položky zpět, která se vyskytuje ve všech pomocných tabulkách. Tímto způsobem se můžeme dostat až na počáteční menu.

4.1.2 Pomocí technologie OTA

V této části se pokusíme využít technologii OTA (Over-The-Air). Tento návrh se zabývá zneužitím technologie OTA k nahrání vlastní aplikace (appletu) na skutečnou SIM/USIM kartu.

Myšlenkou je poslat do telefonu SMS zprávu, obsahující aktualizační data, která by následně nahrála náš applet na SIM/USIM kartu. Tento návrh lze ale použít jen teoreticky. Bylo by třeba nejen implementovat všechny používané šifrovací a integrační algoritmy technologie OTA. Ale především přesvědčit SIM/USIM kartu, že se jedná o legitimní žádost. K tomuto účelu bychom potřebovali aktualizační klíč OTA, který se používá k verifikaci žádostí o aktualizaci.

Pro odeslání námi vytvořené SMS bychom se museli nabourat do již existující komunikace nebo si postavit vlastní komunikační síť.

4.1.3 Pomocí RIL lib. (Radio Interface Layer)

SIM/USIM karta je typicky spojena pouze s Baseband procesorem. Jedná se o integrovaný obvod v bezdrátových komunikačních zařízeních, který zpracovává signál a umožňuje rádiovou komunikaci v reálném čase. Veškerá komunikace mezi mobilními aplikacemi a SIM/USIM musí procházet přes RIL knihovnu. Ta je nepřímo součástí balíčku android, kterou naleznete v každém telefonu s operačním systémem android.

Tato knihovna nám umožňuje přistupovat k SIM/USIM z důvodu autentifikace, čtení a zápisu kontaktů. Podpora pro přímou výměnu APDU příkazů, mezi aplikací a kartou, není vždy k dispozici.

Naším návrhem je využít výše zmíněnou knihovnu ke komunikaci mezi mobilní aplikací se SIM/USIM kartou. Zabezpečená data by byla uložena v souborech na SIM/USIM. Mobilní aplikace by k těmto datům přistupovala pomocí knihovny RIL.

4.1.4 Pomocí originální SIM/USIM a knihovny RIL

Pro realizaci bezpečného přístupu a ukládání dat využít originální (skutečnou) SIM/USIM kartu. Pro nepřímou komunikaci se SIM kartou ze strany mobilního telefonu využít knihovnu RIL.

Ukládat a šifrovat data do nově vytvořených kontaktů. Po dešifrování by bylo možné pracovat s daty v původním formátu.

Obrázek 4.2: Návrh výsledné aplikace

Návrh implementace

Pro dokázání našeho návrhu, nám postačí jednoduché GUI, které může vypadat například takto. Celá aplikace se bude skládat pouze s textového vstupu, výstupu a tří obslužných tlačítek.

Tlačítko ULOŽIT nám zašifruje vstupní text a bezpečně ho uloží do kontaktů.

Tlačítko zrušit nám odstraní uložená data. Bude třeba specifikovat, jaká data mají být odstraněna.

Tlačítko načíst nám rozšifruje uložená data a zobrazí je do výstupního textového pole.

Tento návrh v případě potřeby rozšíříme o další funkčnost.

5 Způsob implementace

Následující kapitola se bude zabývat implementací dvou návrhů. Oba dva byly zrealizovány, ale od jednoho se po neúspěšných pokusech upustilo. Pro bližší informace čtěte dále.

5.1 Implementace Java karty s funkčností GSM

V této části se budeme zabývat využitím obyčejné Java karty. Budeme se snažit aplikovat GSM funkčnost na tuto kartu, tak aby mobilní telefon předpokládal, že se jedná o skutečnou SIM/USIM kartu. Tuto kartu dále použijeme k bezpečnému ukládání a přístupu k zabezpečeným datům.

Budeme-li úspěšní, zrealizujeme návrh 4.1.3 nebo 4.1.1. V případě neúspěchu se pokusíme zrealizovat jiný návrh. Zjistíme-li, že žádné zadání není realizovatelné, budeme pokračovat pomocí simulace.

5.1.1 Odposlouchávání komunikace mezi ME a SIM/USIM

K realizaci našeho návrhu, budeme potřebovat odposlouchávat komunikaci mezi mobilním telefonem a SIM/USIM kartou.

K tomuto účelu lze využít specializovaná zařízení, jako jsou například Turbo Lite 2, Turbo Motion 2 nebo SIMtrace. Uvedená zařízení nejsou zrovna levná. Proto jsme použili obyčejný debugger, který se nachází na Java kartě a slouží k ukládávání příchozích požadavků.

5.1.2 Debugger

Debugger se nachází v samostatném souboru. Umožňuje uložit až 7655 bytů. Při jeho vytváření lze nastavit jeho funkčnost, parametr offset nám udává, od kolikátého příkazu máme začít ukládat. Parametr internalStart nám určuje, zda se mají data zachytávat od začátku komunikace, nebo až po úspěšném zadání CHV 1. Lze změnit i velikost poli bytů, které slouží pro ukládání dat. Změnu lze provést parametrem size. Defaultně je velikost nastavena na 7655.

Seznam dostupných funkcí:

- Pomocí funkce SaveCommand, která má na vstupu pole bytů APDU, uloží příchozí příkaz.
- Data příkazu lze uložit pomocí pomocné funkce SaveData, která má na vstupu totožné pole bytů APDU.
- Data lze obdržet pomocí příkazu PrintBuf. Ten na vstupu požaduje APDU, parametr P1 a velikost. Vrátí do ME data o zadané velikosti od (P1*velikost) bytu.
- Pro vrácení informací týkajících se paměti karty, lze využít příkazy PrintPersistentMemmory, PrintDeselectMemmory a PrintResetMemmory. Vyžadují na vstupu APDU a vracejí bližší informace o typu paměti, kterou naleznete v jejich názvu. Byly vytvořeny i další pomocné funkce.

Obsluhu debuggeru nám zajišťuje třída s CLA=60 v USIM a CLA=70 v SIM.

5.1.3 Obsahy požadovaných souborů

Abychom si byli zcela jistí správností obsahů jednotlivých souborů, budeme aplikovat data získaná ze skutečné SIM/USIM karty. Tyto data si případně upravíme k obrazu svému.

Pomocí výše uvedeného debuggeru jsme zjistili, která data souborů ME požaduje, ty jsme poté replikovali. K získání potřebných dat ze SIM/USIM jsme použili GPShell a čtečku karet. Byly vytvořeny i soubory, které jsou optimální, abychom na nic nezapoměli.

5.1.4 Implementovaná GSM funkčnost

Na Java kartě byla vytvořena samostatná aplikace, která obstarává veškerou GSM funkčnost. Vytvořili jsme obsluhu a realizaci většiny požadavků uvedených v kapitole 3.3.

Nebyla aplikována veškerá funkčnost, funkce měnící obsahy souborů nebyly realizovány. Na tyto a další požadavky jsou vráceny vhodné odpovědi, aby si případně ME myslel, že se jedná o skutečnou SIM/USIM.

5.1.5 Technologie USIM

V této části se pokusíme implementovat GSM funkčnost, která bude replikovat chování USIM karty. Byla vytvořena aplikace (applet) s AID A0000000871002FF47F00189000001FF. Tato aplikace umožňuje fingovaný přechod na applet ADFUSIM AID A0000000871002FF47F00189000001FF, ve které se nachází souborový systém USIM na skutečné kartě.

Byla aplikována funkčnost podle výše uvedené části 5.1.4. Navíc byl umožněn mnohonásobný výběr stejné aplikace. Umožňující vytváření více logických kanálů spojených s danou aplikací.

Obsahy souborů a odpovědi na SELECT a STATUS jsou uloženy ve vhodných konstantách na začátku souboru. Byly vytvořeny podpůrné funkce zajišťující správný běh aplikace. Některé z nich jsou uvedeny níže.

Seznam důležitých funkcí:

- Funkce FileInDic slouží k testování existence souboru podle ID, využívá se při SELECT. Jedná-li se o existující soubor, vrací část ID náležící adresáři, ve kterém se daný soubor nachází. Je-li testován sám adresář, je navrácena část jeho ID. Nebyl-li soubor nalezen, je vrácen byte o hodnotě 0xFF.
- Funkce TryAccessActualFile nám zajišťuje správný pohyb po souborovém systému. Říká nám, zdali můžeme přejít na daný soubor pomocí SELECT z aktuální pozice. K této funkčnosti potřebuje pouze ID souboru a hodnotu parametru P1. Tento parametr nám specifikuje, zdali se jedná o výběr z kořenového adresáře.
- Funkce NeedVerification, která má na vstupu ID souboru, nám sděluje, zdali zadaný soubor potřebuje pro přístup splnit bezpečnostní podmínku CHV 1. Je-li to tak vrací true, jinak false.
- Funkce RecordFileData nám poskytuje data uložená v jednotlivých záznamech, záznamy jsou dostupné pouze pro soubory typu cyklický a lineární. Na vstupu vyžaduje ID souboru, parametr P1 a požadovanou délku. Parametr P1 nám reprezentuje číslo záznamu v souboru. Byl-li záznam nalezen, vrací požadovanou délku záznamu v poli bytů. Nebylo-li hledání úspěšné, vrací pole bytů s požadovanou velikostí obsahující hodnoty 0xFF.
- Funkce RecordFileSize nám podle ID souboru vrátí délku záznamů daného souboru. Nebyl-li soubor nalezen, vrací hodnotu -1;
- Pomocná funkce SetFileSize vrací velikost transparentního souboru. Na vstupu požaduje ID souboru a pomocný parametr P1, který zajišťuje správné fungování.
- Funkce RCreateResponse nám vrací požadovanou velikost dat zadaného souboru. Na vstupu přijímá ID souboru, požadovanou velikost, pole bytů a parametr P1. Požadovaná velikost dat je zkopírována do zadaného pole bytů. Parametr P1 nám zajišťuje speciální funkčnost. Při úspěchu vrací true, jinak false.
- Funkce SCreateResponse nám podle zadaného ID souboru vrací odpověď na SELECT. Odpověď je dostupná přes ukazatel SelectResponsePointer o velikosti SelectResponseSize. Jedná se o globální proměnné.
- Funkce SelectByPath má na vstupu pole bytů, ve kterém se nachází cesta k souboru. Zajišťuje nám výběr souboru podle cesty. Cesta je tvořena posloupností procházených souborů, ty jsou reprezentovány pomocí ID. Při úspěchu vrací true, v jiném případě false.

- Byly vytvořeny i pomocné funkce ArrayCompare, ArrayCopy, které pracují s poli bytů. První slouží pro porovnání, druhý pro překopírování dat. Lze jim zadat požadovanou délku a offset.
- Funkce AfterReset nám slouží pro znovu nastavení hodnot po restartu aplikace. Je důležitá pro resetování bezpečnostních podmínek pro přístup.

Hlavní funkcionalitu nám zajišťuje funkce process, která přijímá APDU příkazy a generuje vhodné odpovědi. Pro správnou funkčnost využívá výše uvedené funkce.

V průběhu implementace byla zjištěna skutečnost, že algoritmy spojené s autentizací USIM nejsou doposud známé veřejnosti, proto byla práce na této technologii zastavena a přešlo se na technologii SIM.

5.1.6 Technologie SIM

Následující pasáž obsahuje informace spojené s pokusem implementace GSM funkčnosti technologie SIM. Výsledkem implementace je applet s AID A0000000090001. Byla aplikována GSM funkčnost podle výše uvedené části 5.1.4. Obsahy jednotlivých souborů jsou uloženy ve vhodných konstantách na začátku souboru. Byly vytvořeny podpůrné funkce zajišťující správný běh aplikace. Některé z nich jsou uvedeny níže.

Seznam důležitých funkcí:

- Funkce NeedVerification, která má na vstupu ID souboru, nám sděluje, zdali zadaný soubor potřebuje pro přístup splnit bezpečnostní podmínku CHV 1. Je-li to tak vrací true, jinak false.
- Funkce CardUnlocked nám vrací true, byla-li splněna bezpečností přístupová podmínka CHV1 nebo CHV2. V opačném případě vrací false.
- Pomocné funkce Clean a FillArray, slouží pro práci s poli bytů. První nám umožňuje vynulovat hodnoty v poli o velikosti n. Druhá nám poskytuje možnost naplnit pole zadaným bytem. Lze nastavit i počáteční byte a velikost.
- Funkce FileInDic slouží k testování existence souboru podle ID, využívá se při SELECT. Jedná-li se o existující soubor, vrací část ID náležící adresáři, ve kterém se daný soubor nachází. Je-li testován sám adresář, je navrácena část jeho ID. Nebyl-li soubor nalezen, je vrácen byte o hodnotě 0xFF.
- Funkce TryAccessActualFile nám zajišťuje správný pohyb po souborovém systému. Říká nám, zdali můžeme přejít na daný soubor pomocí SELECT z aktuální pozice. K této funkčnosti potřebuje pouze ID souboru a hodnotu parametru P1. Tento parametr nám specifikuje, zdali se jedná o výběr z kořenového adresáře.
- Pomocná funkce Detail2CHV požaduje na vstupu počet zbývajících pokusů autorizace a stav autorizace. Využívá se pro CHV1,CHV2,PUK1,PUK2. Jejím účelem je pomáhat vytvářet odpověď na příkazy SELECT a STATUS.
- Funkce SetFileSize nastaví velikost souboru zadaného pomocí ID do určeného pole. Třetí parametr n nám udává index v poli, kam se má velikost nastavit.
- Funkce RecordLength nám vrací velikost záznamu specifikovaného souboru podle ID. Je-li soubor nenalezen, vrací hodnotu 0x00;
- Funkce StructOfFile nám vrací typ souboru zadaného pomocí ID. ('00' transparent, '01' linear fixed, '03' cyclic.) Není-li soubor nalezen, vrací 0x00.
- Funkce CommandFileCond nám vrací bezpečnostní podmínky pro soubor specifikovaný pomocí ID. Parametr Command nám specifikuje, funkce kterým mají podmínky náležet. (1- UPDATE, READ, SEEK; 2- INCREASE; 3-INVALIDATE, REHABILITATE)
- Funkce RCreateResponse nám vrací požadovanou velikost dat zadaného transparentního souboru. Na vstupu přijímá ID souboru, požadovanou velikost, pole bytů. Požadovaná velikost dat je zkopírována do zadaného pole bytů.

- Funkce SCreateResponse nám do zadaného pole bytů vygeneruje odpověď na SELECT a STATUS pro specifikovaný soubor podle ID.
- Funkce AfterReset nám slouží pro znovu nastavení hodnot po restartu aplikace. Je důležitá pro resetování bezpečnostních podmínek pro přístup.
- Byly vytvořeny i pomocné funkce ArrayCompare, ArrayCopy, které pracují s poli bytů. První slouží pro porovnání, druhý pro překopírování dat. Lze jim zadat požadovanou délku a offset.

Hlavní funkcionalitu nám zajišťuje funkce process, která přijímá APDU příkazy a generuje vhodné odpovědi. Pro správnou funkčnost využívá výše uvedené funkce.

Bylo zjištěno, že Java karta implementující GSM funkčnost nestačí pro oklamání mobilního telefonu. I když je možné extrahovat KI ze starých SIM karet, telefon stále rozezná rozdíl. Telefon odmítne komunikovat s kartou ještě před zavoláním autentizační funkce RUN GSM ALGORITHM. Můžeme se pouze domnívat, jak se mobilnímu telefonu podařilo odhalit, že se jedná o falešnou SIM kartu.

Z důvodu zjišťování příčiny byla implementována veškerá funkčnost dostupná pro Java karty. Ale ani touto metodou se nám nepodařilo odhalit příčinu. Potom bylo vyzkoušeno, zdali se některé příkazy nedostanou do našeho debuggeru. Do SIM karty byly zaslány všechny možné kombinace hodnot CLA a INS (255*255 možností). Příkazy, které byly známé SIM kartě, poté byly odeslány do námi vytvořené SIM karty. Všechny příkazy se dostali až do debuggeru.

Z tohoto zjištění usuzuji, že karta byla odhalena podle hodnoty ATR (Answer to reset), kterou nelze zcela změnit a umožňuje identifikovat výrobce a typ karty. Nebo z jiné obdobné příčiny, která mně není dodnes známa.

5.1.7 Pokusy o komunikaci s neautentizovanou SIM/USIM kartou

V průběhu implementace bylo zjištěno, že pro veškerou komunikaci s Android telefonem, se SIM/USIM v prvé řadě musí úspěšně autentizovat v telefonní GSM síti.

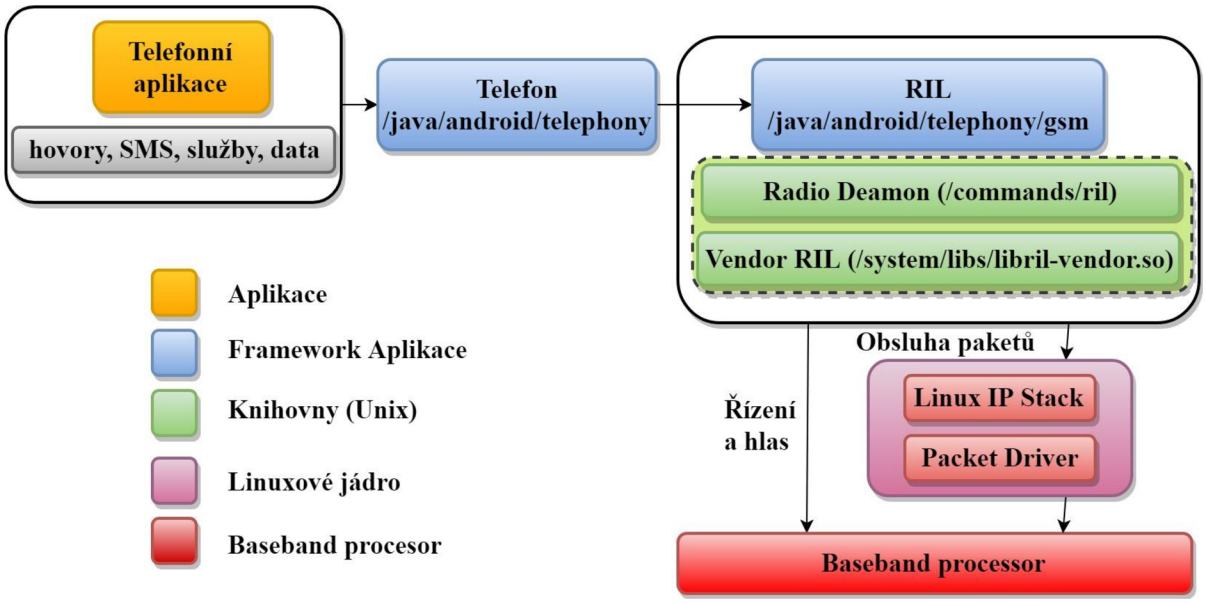
Telefon se karty začne ptát, zdali je něco nového pomocí příkazu FETCH, až po úspěšné autentizaci. Z tohoto důvodu není možné využít menu v Sim Application Toolkit pro neautentizované karty.

Pro komunikaci nelze využít ani RIL knihovnu. Ta také umožňuje komunikaci pouze s autentizovanými kartami.

Veškeré tvrzení výše bylo otestováno.

5.2 Využití originální SIM/USIM a knihovny RIL

Následující obrázek zobrazuje komunikaci mezi aplikacemi a SIM/USIM kartou. Baseband procesor je zařízení, které nám spravuje veškerou funkčnost rádia. Tento procesor přímo komunikuje se SIM/USIM kartou. Můžete vidět, že aplikace komunikují s Baseband procesorem přes RIL (Radio Interface Layer) knihovnu.



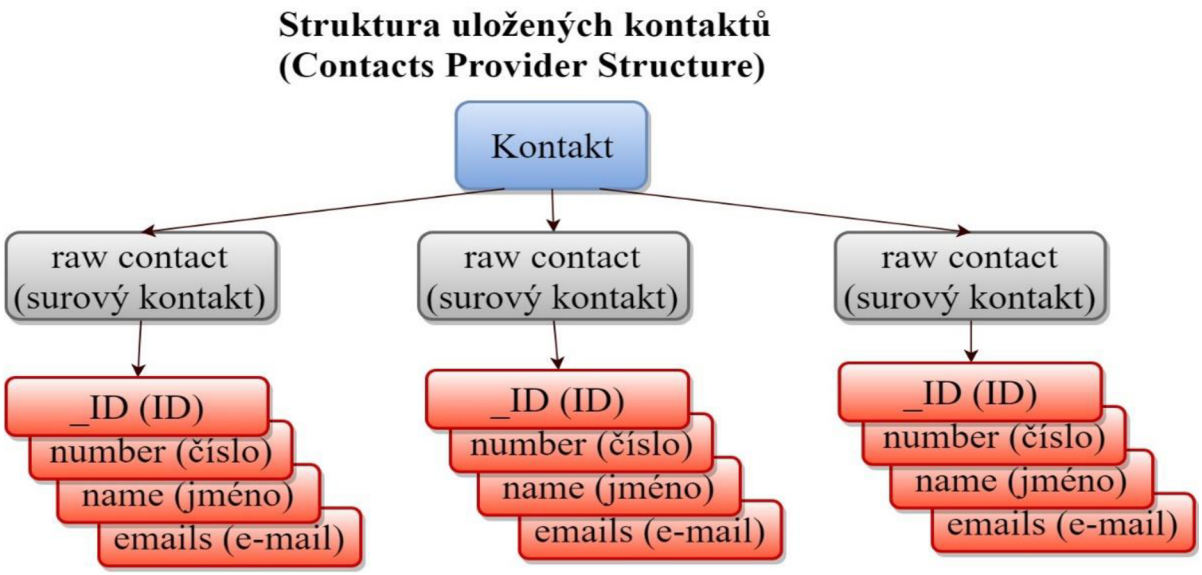
Obrázek 5.1: Schéma komunikace mezi MS a Baseband procesorem [11][41][25]

V této části se budeme zabývat využitím originální SIM/USIM karty k bezpečnému přístupu a ukládání dat. Budeme se snažit implementovat výše uvedený návrh číslo 4.1.4.

Implementujeme šifrování dat do kontaktů uložených na SIM/USIM kartě. Budeme-li úspěšní, data budou zabezpečená nejen pomocí CHV 1, ale i pomocí námi vytvořeného šifrovacího algoritmu. Data budou jednoduše přenosná, díky absolutní kompatibilitě SIM/USIM karet s telefonními zařízeními.

5.2.1 Implementace komunikace

Veškeré kontakty na SIM/USIM kartě, jsou uložené v příslušném souboru. Kontakty jsou ukládány jako záznamy. Pro jednodušší práci s kontakty, nám Android knihovny umožňují pracovat s abstraktní strukturou kontaktů. V obrázku níže si můžete prohlédnout abstraktní strukturu uložených kontaktů.



Obrázek 5.2: Struktura uložených kontaktů [6]

Pro získání povolení přistupovat a modifikovat kontakty, je nutné nastavit aplikaci speciální povolení. Jedná se konkrétně o user-permission (uživatelská oprávnění). Bylo třeba povolit následující oprávnění READ_CONTACTS a WRITE_CONTACTS. Ty můžete nalézt v xml souboru AndroidManifest.

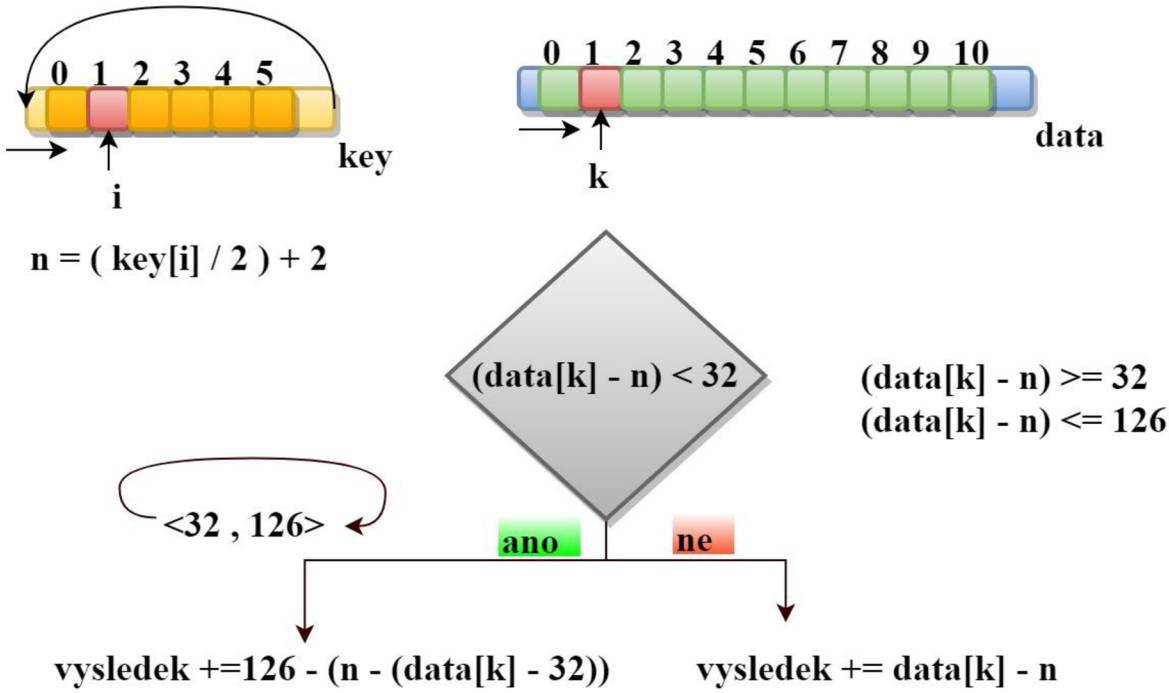
Pro komunikaci byla využita třída Uri. Pomocí této třídy si otevřeme ukazatel na dostupný zdroj a to konkrétně naši SIM/USIM kartu. A to všechno pomocí příkazu Uri.parse("content://icc/adn").

Pro přístup k obsahu jednotlivých kontaktů, jsme využili příkaz getContentResolver().query(). Ve vstupních parametrech jsme mu poskytli získaný Uri odkaz na SIM/USIM kartu. Jako výstup jsme obdrželi Cursor, který nám umožňuje jednoduchý průchod dvou dimenzionální tabulkou. Každý kontakt v tabulce je reprezentován telefonním číslem, jménem, e-mailem, ID. Pro získání příslušných informací, slouží příkaz cursor.getString(cursorSim.getColumnIndex(data)). Parametr data lze nahradit textovým řetězcem, například number, name, emails, _id. Specifikuje, o kterou informaci máme právě zájem.

Pro vytváření nového kontaktu jsme použili třídu ContentValues. Pomocí příkazu put přidáváme jednotlivé hodnoty. Hodnota tag reprezentuje jméno, number telefonní číslo, emails příslušný e-mail. Takto připravenou strukturu přidáme do kontaktů na SIM/USIM kartě, užitím příkazu getContentResolver().insert(). Ve vstupních parametrech mu poskytneme vzniklou strukturu a Uri odkaz na SIM/USIM kartu. Poté informujeme kartu o změnách, které jsme udělali, použitím příkazu getContentResolver().notifyChange().

5.2.2 Šifrovací algoritmus

V následujícím obrázku si můžete prohlédnout, jak celé šifrování dat probíhá. Pod obrázkem jsou dostupné bližší informace spojené s touto problematikou.



Obrázek 5.3: Použitý šifrovací algoritmus

Do kontaktů lze ukládat pouze tisknutelné znaky. Z tohoto důvodu musíme šifrování omezit na znaky s ascii hodnotou <32-126>.

Bylo třeba navrhnout a implementovat šifrovací algoritmus, který dodržuje příslušné rozmezí. Pro šifrování byl použit následující algoritmus.

Šifrovací algoritmus

Pro šifrování používáme bezpečnostní klíč, uložený v privátní globální proměnné key. V této proměnné se pohybujeme pomocí indexu směrem vpřed. Dojdeme-li na konec klíče, začneme znovu od začátku.

Od každé ascii hodnoty znaku určeného k zašifrování, odečteme hodnotu získanou z bezpečnostního klíče. Hodnota je získána užitím jednoduchého algoritmu, konkrétně se jedná o algoritmus (key[i] / 2) + 2. Překročíme-li při odečítání povolené rozmezí hodnot, odečteme zbývající hodnotu od maximální povolené hodnoty. Jedná se o cyklický algoritmus. Pro dešifrování se používá opačný algoritmus.

5.2.3 Interakce s uživatelem

Interakce s uživatelem probíhá pomocí jednoduchého GUI, vycházejícího z návrhu 4.1.4.

Obrázek 5.4: Výsledná aplikace

Pod úvodem se nachází textový vstup, který umožňuje zadat až 57 znaků. Poté následují tři tlačítka, která umožňují řídit aplikaci.

Tlačítko uložit slouží k ukládání zadaných dat. Data mohou být uložena v šifrované formě nebo v nezašifrované. O nastavení rozhoduje zaškrtávací políčko pod tlačítkem.

Tlačítko zrušit nám zruší specifikované uložené bezpečnostní data. Zadaná data jsou hledána v zašifrované i nezašifrované formě automaticky. Zaškrtávací políčko pod daným tlačítkem, nám umožňuje zrušit veškerá uložená data.

Tlačítko načíst nám vyhledá a zobrazí uložená data. O rozhodování, zdali mají býti data zobrazeny v dešifrované formě, rozhoduje políčko níže. Výstupní data jsou zobrazována v neviditelném textovém poli, které se nachází pod tlačítky.

Výsledná aplikace byla rozšířená o následující funkčnost oproti návrhu. Byla přidána zaškrtávací políčka, která umožňují ovládat šifrování a dešifrování. Bylo přidáno i políčko umožňující rychlé mazání veškerých uložených dat.

5.2.4 Vytvořené funkce

V této části naleznete funkce, které umožňují správný běh aplikace.

Funkce loadButton, saveButton a deleteButton nám specifikují chování vyvolané jednotlivými tlačítky.

Pro ukládání nových kontaktů nám slouží funkce insertSIMContact, na vstupu přijímá zabezpečená data a telefonní číslo nově vznikajícího kontaktu.

Funkce contactNumber vyhledá příslušné zabezpečené data a vrátí telefonní číslo kontaktu, jemuž náleží. Číslo je navráceno v textové formě.

Pro kontrolu existence kontaktu slouží funkce contactExist, která podle vstupních dat zkontroluje, zdali již stejná data nebyla uložena. Při nálezu vrátí true, jinak false.

Funkce deleteContact nám odstraní příslušný kontakt specifikovaný bezpečnostními daty a telefonním číslem.

Pro zobrazování všech uložených dat slouží funkce readContacts. Jednotlivé data jsou zobrazovány na nové řádky.

Pro řízení šifrováni a dešifrování slouží funkce algEncrypt, encryptData a decryptData.

5.2.5 Omezení a způsob implementace

Zabezpečená data jsou ukládána do jména kontaktu i do příslušného e-mailu kontaktu. Maximální velikost jména je 14 a e-mail dovoluje bezpečně uložit až 43 znaků. Z tohoto důvodu je možné uložit do jednoho kontaktu až 57 znaků. Nejprve jsou data ukládány do jména kontaktu, poté se ukládají do e-mailu.

Z důvodu bezpečnosti byla omezena maximální velikost vstupního řetězce na 57 znaků. Při překročení, je uživatel vhodným způsobem informován.

5.3 Program použitý pro komunikaci s kartou

Pro komunikaci se všemi kartami byl použit program GPShell (Global Platform Shell). Ten nám umožňuje nahrávání knihoven, instalaci aplikací a běžnou komunikaci s kartou pomocí APDU příkazů.

Pro danou funkčnost je třeba i čtečka karet. Pro realizaci byla použita čtečka karet značky gemalto s P/N HWP117685E.

Pro kontrolu výstupů z GPShell byla vytvořena Java aplikace, sloužící taktéž ke komunikaci s čtečkou karet a přítomnými kartami.

5.4 Použité Java a SIM/USIM karty

Pro implementaci byly použity Java karty typu TOP IM GX4, maska čipu MSA081. V průběhu práce padly 3 karty tohoto typu za oběť. Kdy byly nadobro zablokovány, či znehodnoceny.

Pro výzkum byly použity USIM karty od výrobců O2, T-mobile a Tesco Mobile. I jedna SIM karta z roku 2002 od výrobce T-mobile. Celkem 2 karty byly nadobro zničeny.

5.5 Informace o použitých mobilních zařízeních

V této části naleznete informace o zařízeních, na kterých byla výsledná aplikace testována a vyvíjena. Vzniklá aplikace byla realizována na starším mobilním telefonu HTC Desire S S510e s androidem 2.3.5. I na telefonu značky Motorola XT615 s androidem 2.3.7.

5.6 Informace o použitých IDE

Pro vývoj aplikací pro telefony s operačním systémem Android, byla využita nejnovější verze Android studia 3.1.2. Aplikace by měla být kompatibilní se zařízeními s API levelem >= 9.

Applet pro Java karty s technologií SIM byl vytvářen v nejnovější verzi JCID, která je součástí JavaCOS Panel. Pro překlad byl použit JDK 2.2.2.

Applet pro Java karty s technologií USIM byl vytvořen v NetBeans IDE 8.2. Pro překlad byl použit JDK 2.2.2. Ten se musel externě přidat do tohoto IDE.

6 Závěr

Účelem bakalářské práce bylo navrhnout a implementovat využití druhého SIM slotu v telefonu, jako bezpečnostního modulu.

Bylo zjištěno, že nelze využít obyčejnou Java kartu podporující GSM funkčnost k obelhání telefonu. Po verifikaci pomocí pinu, je falešná SIM karta vždy odhalena.

Telefon s operačním systémem Android, neumožňuje komunikovat s neautentizovanou SIM/USIM kartou. Bylo by nutné nabourat originální SIM/USIM kartu k získání bezpečnostního klíče. I když lze získat bezpečnostní klíč KI ze starých SIM karet, jednalo by se o trestnou činnost. A z tohoto důvodu, by se to dalo využít pouze ke studijním účelům. I kdyby se nám povedlo získat potřebný bezpečnostní klíč, stále bychom nebyli schopni využít obyčejnou Java kartu k řešení našeho problému.

K vyřešení našeho problému, lze ale použít pravou SIM/USIM kartu. Výzkumem a testováním, jsme přišli na způsob, jak bezpečně ukládat data na originální SIM/USIM kartu.

Data jsou zašifrovány a uloženy na kartu, jako data nově vzniklého kontaktu. Zašifrované informace jsou ukládány do jména a e-mailu kontaktu. Tato metoda nám umožňuje uložit až 57 tisknutelných znaku do jednoho jediného kontaktu.

Takto uložené informace, lze snadno a jednoduše přemístit do jiného zařízení, přendáním použité SIM/USIM karty. K dešifrování dat, nám stačí znát použitý algoritmus a šifrovací klíč.

Ze získaných znalostí a zkušeností usuzuji, že lze využít nejen druhý SIM slot v telefonu k bezpečnému ukládání dat, ale i samostatný SIM slot. Takto zabezpečená data, lze využít nikoli jen v mobilních telefonech typu android, ale v každém zařízení, které umožňuje komunikaci se SIM/USIM kartou.

Tato metoda je vynikajícím řešením, nechcete-li nechat uživatele přímo používat bezpečnostní data. Stačí, aby uživatel zadal správný šifrovací kód a k zabezpečeným datům se ani nepřiblíží. S každým novým zařízením, není nutné generovat nové bezpečnostní data každému uživateli, postačí přenést SIM/USIM kartu. Správnou metodou lze kontakty spojovat a ukládat díky tomu i více než 57 symbolů. Touto metodou, lze například identifikovat uživatele v každém používaném zařízení, podporující SIM/USIM karty. Jedná se o inovativní metodu, která jistě v budoucnu nalezne více než jedno využití.

Literatura

[1] 3GPP TS 11.11. Sophia-Antipolis Cedex: 3GPP Mobile Competence Centre, 1999 verze 5.3.0 [Online; navštíveno 28.10.2017].
URL http://www.etsi.org/deliver/etsi_gts/11/1111/05.03.00_60/gsmts_1111v050300p.pdf

[2] European Telecommunications Standards Institute: Specifications of the SIM-ME interface, 1994 verze 3.16.0 [Online; navštíveno 7.11.2017].
URI http://www.etsi.org/deliver/etsi_gts/11/1111/03.16.00_60/gsmts_1111sv031600p.pdf

[3] 3GPP TS 11.14. Sophia-Antipolis Cedex: 3GPP Mobile Competence Centre, 1999 [Online; navštíveno 15.11.2017].
URL http://www.etsi.org/deliver/etsi_gts/11/1114/05.09.00_60/gsmts_1114v050900p.pdf

[4] REDL, S., WEBER M., and OLIPHANT, M. V.: GSM and Personal Communications handbook London: Artech House Publishers, 1998 [Online; navštíveno 15.02.2018].
URL https://gctjaipur.files.wordpress.com/2015/08/gsm-and-personal-communications.pdf

[5] HARRINGTON, M.: SIM card protocols, 2007 [Online; navštíveno 28.02.2018].
URL https://mobileforensics.files.wordpress.com/2007/03/sim-card-protocols.pdf

[6] Android developer guide (Průvodce pro vývojáře Android) [Online; navštíveno 28.04.2018].
URL https://developer.android.com/guide/

[7] Java SE - Downloads | Oracle Technology Network. Oracle | Integrated Cloud Applications and Platform Services [Online; navštíveno 20.11.2017].
URL http://www.oracle.com/technetwork/java/javase/downloads/index.html

[8] JCID for JavaCardoOS (vývojové prostředí pro Java karty) [Online; navštíveno 22.11.2017].
URL https://javacardos.com/javacardforum/

[9] Taimur Akmal, Team Android: 2014 [Online; navštíveno 24.04.2018].
URL http://www.teamandroid.com/2014/02/19/install-android-442-sdk-try-kitkat-now/

[10] 3GPP: Universal Subscriber Identity Module Application Toolkit, 2011 verze 31.111 [Online; navštíveno 09.03.2018].
URL http://www.3gpp.org/ftp/Specs/html-info/31111.htm

[11] ALLEN, Grant. Android 4: průvodce programováním mobilních aplikací. Brno: Computer Press, 2013. ISBN 978-80-251-3782-6. [kniha; 17.03.2018].

[12] 3GPP: Mobile radio interface Layer 3 specification 24.008, 2018 verze 15.2.0 [Online; navštíveno 24.04.2018].
URL http://www.3gpp.org/ftp/Specs/html-info/24008.htm

[13] European Telecommunications Standards Institute: Smart Cards: Card Application Tookit, 2011 [Online; navštíveno 09.02.2018].
URL http://www.etsi.org/deliver/etsi_ts/102200_102299/102223/10.05.00_60/ts_102223v100500p.pdf

[14] European Telecommunications Standards Institute: Smart Cards: UICC-Terminal Interface, Physical and logical characteristics, 2011 verze 10.0.0 [Online; navštíveno 09.02.2018].
URL http://www.etsi.org/deliver/etsi_ts/102200_102299/102221/10.00.00_60/ts_102221v100000p.pdf

[15] Digital cellular telecommunications system (Phase 2+): Specification of the SIM Application Toolkit (SAT) for the Subscriber Identity Module - Mobile Equipment (SIM-ME) interface 3GPP TS 11.14, 1999 verze 8.18.0 [Online; navštíveno 09.02.2018].
URL http://www.etsi.org/deliver/etsi_ts/101200_101299/101267/08.18.00_60/ts_101267v081800p.pdf

[16] European Telecommunications Standards Institute: Smart Cards: Card Application Toolkit, 2011 verze 9.0.0 [Online; navštíveno 10.02.2018].
URL http://www.etsi.org/deliver/etsi_ts/102200_102299/102223/09.00.00_60/ts_102223v090000p.pdf

[17] Universal Mobile Telecommunications System: LTE: Characteristics of the Universal Subscriber Identity Module (USIM) application 3GPP TS 31.102, 2018 verze 14.4.0 [Online; navštíveno 10.02.2018].
URL http://www.etsi.org/deliver/etsi_ts/131100_131199/131102/14.04.00_60/ts_131102v140400p.pdf
[18] Universal Mobile Telecommunications System: USIM Conformance Test Specification 3GPP TS 31.122, 1999 verze 3.1.0 [Online; navštíveno 16.02.2018].
URL http://www.etsi.org/deliver/etsi_ts/131100_131199/131122/03.01.00_60/ts_131122v030100p.pdf
[19] Universal Mobile Telecommunications System: LTE: Universal Subscriber Identity Module (USIM) conformance test specification 3GPP TS 31.122, 2017 verze 14.0.0 [Online; navštíveno 23.02.2018].
URL http://www.etsi.org/deliver/etsi_ts/131100_131199/131122/14.00.00_60/ts_131122v140000p.pdf#page=94
[20] Nikolay Elenkov: Using the SIM card as a secure element in Android, 2013 [Online; navštíveno 25.02.2018].
URL https://nelenkov.blogspot.cz/2013/09/using-sim-card-as-secure-element.html
[21] Computer Network | Circuit Switching VS Packet Switching [Online; navštíveno 27.02.2018].
URL https://www.geeksforgeeks.org/computer-network-circuit-switching-vs-packet-switching/
[22] Karl Koscher, Eric Butler: Defcon 21 - The Secret Life of SIM Cards, 2013 [Online; navštíveno 28.02.2018].
URL https://www.youtube.com/watch?v=31D94QOo2gY
[23] Research Article: Hindawi Publishing Corporation: EURASIP Journal on Wireless Communications and Networking, 2011 [Online; navštíveno 01.03.2018].
URL https://jwcn-eurasipjournals.springeropen.com/track/pdf/10.1155/2011/867315
[24] Smart card standard iso7816-4: section 6 basic interindustry commands [Online; navštíveno 03.03.2018].
URL http://cardwerk.com/smart-card-standard-iso7816-4-section-6-basic-interindustry-commands/
[25] Source Android: RIL Refactoring [Online; navštíveno 23.04.2018].
URI https://source.android.com/devices/tech/connect/ril
[26] Oracle help center: Java™ Platform Standart Ed. 7 [Online; navštíveno 25.04.2018].
URL https://docs.oracle.com/javase/7/docs/api/java/security/Provider.html#getServices()
[27] Simalliance: Open Mobile API Specification, 2015 [Online; navštíveno 27.04.2018]
URL http://simalliance.org/wp-content/uploads/2015/03/SIMalliance_OpenMobileAPI3_1_.pdf
[28] Oracle help center: specifikace třídy APDU [Online; navštíveno 28.04.2018]
URL https://docs.oracle.com/javacard/3.0.5/api/javacard/framework/APDU.html#isCommandChainingCLA()
[29] Oracle help center: specifikace třídy Dispacher [Online; navštíveno 28.04.2018]
URI https://docs.oracle.com/javacard/3.0.5/api/javacard/framework/service/Dispatcher.html
[30] Javatips: Java Examples for javacard.framework.JCSystem [Online; navštíveno 28.04.2018]
URL https://www.javatips.net/api/javacard.framework.jcsystem
[31] Java Card v2.2.2.: specifikace třídy RMIService [Online; navštíveno 28.04.2018]
URL https://www.win.tue.nl/pinpasjc/docs/apis/jc222/javacard/framework/service/RMIService.html
[32] Java Card v2.2.2: specifikace třídy JCSystem [Online; navštíveno 28.04.2018]
URL https://www.win.tue.nl/pinpasjc/docs/apis/jc222/javacard/framework/JCSystem.html
[33] Application Programming Notes Java Card Platform: Working with Logical Channels, verze 2.2.1 [Online; navštíveno 29.04.2018]
ULR https://askra.de/software/jcdocs/app-notes-2.2.1/logchan.html
[34] Oracle Technology Network: C. Enrique Ortiz: An Introduction to Java Card Technology - Part 1, 2003 [Online; navštíveno 20.12.2017]
URL http://www.oracle.com/technetwork/java/javacard/javacard1-139251.html

[35] Runtime Environment Specification for the Java Card Platform: Version 2.2.2 [Online; navštíveno 24.01.2018]
URL https://anon.inf.tu-dresden.de/svn/JavaCardStudents/ANONCard/trunk/java_card_kit-2_2_2/jc_specification/specs/jcre/html/JCRESpec04selection.html
[36] GUTHERY: Scott B. a Mary J. CRONIN.: Mobile application development with SMS and the SIM toolkit. New York: McGraw-Hill, 2002. ISBN 0-07-137540-6. [ekniha; navštíveno 03.02.2018]
[37] GSM (Global System for Mobile communications): Wikipedia: the free encyclopedia, [Online; navštíveno 24.04.2018].
URL https://en.wikipedia.org/wiki/GSM
[38] Subscriber identity module (SIM): Wikipedia: the free encyclopedia, [Online; navštíveno 24.04.2018].
URL https://en.wikipedia.org/wiki/Subscriber_identity_module
[39] Android (operating system): Wikipedia: the free encyclopedia, [Online; navštíveno 25.04.2018].
URL https://en.wikipedia.org/wiki/Android_(operating_system)
[40] Java Card: Wikipedia: the free encyclopedia, [Online; navštíveno 25.04.2018].
URL https://en.wikipedia.org/wiki/Java_Card
[41] Radio Interface Layer (RIL): Wikipedia: the free encyclopedia, [Online; navštíveno 28.04.2018].
URL https://en.wikipedia.org/wiki/Radio_Interface_Layer
[42] TelecomPedia: User Equipment and SIM/USIM, [Online; navštíveno 29.04.2018].
URL http://telecompedia.net/user-equipment-and-sim-usim/
[43] Gemato: OTA (Over-The-Air) [Online; navštíveno 01.05.2018].
URL https://www.gemalto.com/companyinfo/digital-security/techno/ota
[44] Baseband processor: Wikipedia: the free encyclopedia, [Online; navštíveno 02.05.2018].
URL https://en.wikipedia.org/wiki/Baseband_processor
[45] Home Location Register: Wikipedia: the free encyclopedia, [Online; navštíveno 02.05.2018].
URL https://en.wikipedia.org/wiki/Network_switching_subsystem#Home_location_register_.28HLR.29
[46] CHANDRA, Praphul: Bulletproof wireless security: GSM, UMTS, 802.11 and ad hoc security. Oxford: Newnes, 2005. Communications engineering series. [ekniha; navštíveno 05.05.2018].

Přílohy

Příloha A

Obsah CD

Na přiloženém CD naleznete veškeré zdrojové soubory, dokumentaci, hlavní a pomocné aplikace, manuál pro přeložení výsledné aplikace. Zde naleznete adresářovou strukturu, která se nachází na přiloženém CD disku:

- **doc/** Zdrojový soubor použitý pro vytvoření PDF souboru
- **doc/obrazky** Všechny obrázky použité v této práci
- **pdf/** Výsledný PDF soubor práce
- **javaCard/sim** Přeložený applet SIM pro Java karty
- **javaCard/usim** Přeložený applet USIM pro Java karty
- **javaCard/logg** Obsahuje výpis z debuggeru
- **javaCard/src/sim** Zdrojové soubory pro SIM applet
- **javaCard/src/usim** Zdrojové soubory pro USIM applet
- **android/aplikace/** Výsledná přeložená aplikace na Android
- **android/src/** Zdrojové soubory pro Android aplikaci
- **parser/** Webová pomocná aplikace pro parsování příkazů z debuggeru
- **reader/** Zdrojové soubory pomocné aplikace pro komunikaci s kartou

Příloha B

Manuál k Android aplikaci

Následující text slouží jako jednoduchý návod k instalaci a překladu aplikace.

B.1 Překlad aplikace

1) Stáhneme si nejnovější verzi Android Studia
Aplikace je volně dostupná na stránkách: https://developer.android.com/studio/
2) Otevřeme si projekt v android/src/ na přiloženém CD
3) Jdeme do Build --> Generate Signed APK (Nemáme-li podepisovací klíč, je třeba ho vytvořit)

B.2 Instalace aplikace

1) Nahrajeme aplikaci, .apk soubor do mobilního telefonu
2) Spustíme instalaci

Příloha C

Manuál k instalaci a řízení Appletů

Následující text slouží jako jednoduchý návod k instalaci a řízení appletů.

C.1 Postup k instalaci .cap souboru

1) Stáhněte si nejnovější verzi GPShell.
Aplikace je volně dostupná na stránkách: https://sourceforge.net/projects/globalplatform/
2) Zkopírujte si .cap soubor do adresáře s gp.exe (Je třeba zkopírovat i debuger.cap)
3) Otevřete si příkazový řádek
4) Pomocí příkazového řádku přejděte do adresáře s gp.exe
5) Připojte čtečku karet k počítači a vložte do ní Java kartu
6) Pomocí příkazového řádku a uvedeného příkazu nainstalujete applet jako výchozí applet pro připojenou Java kartu (Máte-li jiný bezpečnostní klíč, než základní z výroby, nejprve ho musíte korektně zadat)
gp –load debuger.cap
gp –install <soubor.cap> -default

C.2 Řízení appletu

Applet je nastaven, jako výchozí. Pro komunikaci s appletem postačí zasílat APDU příkazy do Java karty. Pro zadávání příkazů lze použít například přiložený pomocný program, který naleznete na CD v adresáři reader/. Applet podporuje většinu výše uvedených příkazů.

Příloha D

Návratové kódy

Zde jsou uvedeny všechny návratové kódy technologie SIM a ty nejdůležitější z USIM.

SIM			Druhy odpovědí	USIM	
SW1	**SW2**	**Č.**	**Vykonáno úspěšně**	**SW1**	**SW2**
'90'	'00'	**01.**	úspěšně vykonáno	'90'	'00'
'91'	'XX'	**02.**	'XX' dostupných dat v GET RESPONSE	'91'	'XX'
'9F'	'XX'	**03.**	'XX' dat v odpovědi	'92'	'XX'
			Příkaz byl odložen		
'93'	'00'	**04.**	zaneprázdněno, příkaz momentálně nelze vykonat	'93'	'00'
			Správa paměti		
'92'	'0X'	**05.**	úspěšně vykonáno, ale po aktualizaci opakujete 'X' krát	'63'	'CX'
'92'	'40'	**06.**	problém s pamětí	'65'	'81'
			Správa ukazatelů		
'94'	'00'	**07.**	žádný vybraný EF	'69'	'86'
'94'	'02'	**08.**	mimo platný rozsah	'6A'	'88'
		09.	záznam nenalezen	'6A'	'83'
'94'	'04'	**10.**	soubor nenalezen	'6A'	'82'
'94'	'08'	**11.**	vybraný soubor je nekompatibilní s daným příkazem	'6A'	'81'
			Správa bezpečnosti		
'98'	'02'	**12.**	CHV (PIN) nebyl inicializován		
'98'	'04'	**13.**	neúspěšná verifikace CHV, alespoň jeden pokus zbývá	'63'	'CX'
'98'	'08'	**14.**	v rozporu se stavem CHV (PIN)	'69'	'84'
'98'	'10'	**15.**	v rozporu se stavem, soubor byl zneplatněn		
'98'	'40'	**16.**	neúspěšná verifikace, CHV zablokován	'69'	'83'
		17.	chyba autentifikace	'98'	'62'
		18.	relace vypršela	'98'	'63'
'98'	'50'	**19.**	dosažena maximální hodnota	'98'	'50'
			Chyby nezávislé na aplikaci		
'67'	'XX'	**20.**	špatný parametr P3, má se rovnat 'XX'	'67'	'XX'
'6B'	'XX'	**21.**	špatný parametr P1 nebo P2, má mít hodnotu 'XX'	'6B'	'XX'
'6D'	'XX'	**22.**	neznámý INS kód příkazu	'6D'	'00'
'6E'	'XX'	**23.**	špatná třída příkazu	'6E'	'00'
'6F'	'XX'	**24.**	technický problém	'6F'	'00'

Tabulka D.1: Důležité návratové kódy SIM/USIM [1][19]

Níže naleznete tabulku jednotlivých příkazů a jejich návratových kódů.

	Úspěšné				**Neúspěšné ukončení**																			
Jednotlivé příkazy	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
SELECT			X			X				X										X	X	X	X	X
STATUS	X	X				X														X	X	X	X	X
UPDATE BINARY	X	X			X	X	X				X		X		X					X	X	X	X	X
UPDATE RECORD	X	X			X	X	X	X	X		X		X		X					X	X	X	X	X
READ BINARY	X	X				X	X				X		X		X					X	X	X	X	X
READ RECORD	X	X				X	X	X	X		X		X		X					X	X	X	X	X
SEEK	X		X			X	X		X	X	X		X		X					X	X	X	X	X
INCREASE			X		X	X	X				X		X		X				X	X	X	X	X	X
VERIFY CHV	X	X			X	X						X	X	X		X				X	X	X	X	X
CHANGE CHV	X	X			X	X						X	X	X		X				X	X	X	X	X
DISABLE CHV	X	X			X	X						X	X	X		X				X	X	X	X	X
ENABLE CHV	X	X			X	X						X	X	X		X				X	X	X	X	X
UNBLOCK CHV	X	X			X	X						X	X	X		X				X	X	X	X	X
INVALIDATE	X	X			X	X	X						X		X					X	X	X	X	X
REHABILITATE	X	X			X	X	X						X		X					X	X	X	X	X
RUN GSM ALG.			X			X					X		X							X	X	X	X	X
SLEEP	X																			X	X	X	X	X
GET RESPONSE	X	X				X														X	X	X	X	X
TERMINAL PROF.	X	X			X	X														X	X	X	X	X
ENVELOPE	X	X	X	X	X	X														X	X	X	X	X
FETCH	X					X														X	X	X	X	X
TERMINAL RES.	X	X			X	X														X	X	X	X	X
AUTHENTICATE			X			X		X						X			X			X	X		X	X
MANAGE CH.	X		X								X							X		X	X		X	X

Tabulka D.2: Podpora návratových kódů jednotlivých příkazů [1][19]