FAKULTA INFORMAČNÍCH TECHNOLOGIÍ ČVUT V PRAZE

Zadání bakalářské práce

Název:	Testování bezpečnosti v mobilních sítích
Student:	Simona Lániková
Vedoucí:	Ing. Jiří Dostál, Ph.D.
Studijní program:	Informatika
Obor / specializace:	Bezpečnost a informační technologie
Katedra:	Katedra počítačových systémů
Platnost zadání:	do konce letního semestru 2022/2023

Pokyny pro vypracování

Zabezpečení mobilních sítí, zejména druhé generace (2G), je zastaralé a téměř po třech desetiletích neobstojí novým hrozbám. Prozkoumejte technologie mobilních sítí druhé až páté generace a porovnejte zejména problematiku autentizace uživatelů, zabezpečení datových a hlasových přenosů a technologii SIM karet. Navrhněte a implementujte nástroj pro demonstraci šifrovacích algoritmů a generování hodnot kryptografických dat SIM karet (symetrický klíč, kód operátora, apod.). Proveďte integraci nástroje do stávajícího projektu testovací 4G sítě pro podporu penetračního testování IoT zařízení. Nástroj otestujte na reálných datech a zdokumentujte.

Elektronicky schválil/a prof. Ing. Pavel Tvrdík, CSc. dne 30. ledna 2022 v Praze.

Bakalárska práca

TESTOVANIE BEZPEČNOSTI V MOBILNÝCH SIEŤACH

Simona Lániková

Fakulta informačních technologií
Katedra informační bezpečnosti
Vedúci: Ing. Jiří Dostál, Ph.D.
10. mája 2022

Obsah

Zoznam obrázkov

Zoznam tabuliek

Zoznam výpisov kódu

Chcela by som sa predovšetkým poďakovať vedúcemu mojej práce Ing. Jiřímu Dostálovi, Ph.D. za jeho cenné rady a pripomienky, ktoré mi pomohli pri tvorbe tejto práce. Ďalej by som sa chcela poďakovať svojej rodine a kamarátom za ich podporu pri štúdiu a písaní tejto práce.

Vyhlásenie

Prohlašuji, že jsem předloženou práci vypracovala samostatně a že jsem uvedla veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací. Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů, zejména skutečnost, že České vysoké učení technické v Praze má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 citovaného zákona.

V Prahe dňa 10. mája 2022 .....

Abstrakt

Práca sa zaoberá technológiami využitých v mobilných sieťach druhej až piatej generácie s dôrazom na autentizáciu a zabezpečenie hlasových a dátových prenosov. Predovšetkým sa zameriava na siete štvrtej generácie a implementáciu autentizačných algoritmov MILENAGE a XOR použiteľnú na testovacie účely a účely ladenia. Popisuje projekt srsRAN slúžiaci na vytvorenie mobilnej siete a rozširuje tento projekt o nástroj pre generovanie súboru user_db.csv, ktorý predstavuje databázu užívateľov uloženú v jadre siete (srsEPC).

Klíčová slova mobilné siete, bezpečnosť mobilných sietí, autentizácia, zabezpečenie SIM kariet, implementácia algoritmu MILENAGE, srsRAN

Abstract

This thesis deals with technologies of the 2nd through 5th generation of mobile networks with a focus on authentication and security of voice and data transfers. Above all, it focuses on the 4th generation networks and implementation of MILENAGE and XOR authentication algorithms suitable for testing and debugging. It describes the srsRAN project, which can be used to create a virtual mobile network, and extends this project with a tool for user_db.csv file generation, which represents the user database stored in the network core (srsEPC).

Keywords mobile networks, security in mobile networks, authentication, SIM cards security, MILENAGE implementation, srsRAN

Zoznam skratiek

AKA	Authentication and Key Agreement
AMF	Authentication Management Field
AuC	Authentication Center
AUTN	Authentication Token
BCCH	Broadcast Control Channel
BSC	Base Station Controller
BSS	Base Station Subsystem
BTS	Base Transceiver Station
CDMA	Code Division Multiple Access
CN	Circuit-Switched Network
EHF	Extreme High Frequency
EIR	Equipment Identity Register
EPC	Evolved Packet Core
FDMA	Frequency Division Multiple Access
GGSN	Gateway GPRS Support Node
GMSC	Gateway Mobile Switching Center
GSM	Global System for Mobile Communications
HE	Home Environment
HLR	Home Location Register
IMEI	International Mobile Equipment Identity
IMSI	International Mobile Subscriber Identity
IoT	Internet of Things
LA	Location Area
LAI	Location Area Identity
LTE	Long Term Evolution
MAC	Message Authentication Code
MCC	Mobile Country Code
MNC	Mobile Network Code
MS	Mobile Station
MSC	Mobile Switching Center
NSS	Network Switching Subsystem
OFDMA	Orthogonal Frequency Division Multiple Access
OP	Operator Code
OSS	Operation Support Subsystem
P-GW	Packet Data Network Gateway
PLMN	Public Land Mobile Network
PN	Packet-Switched Network
QoS	Quality of Service
RAN	Radio Access Network
RNC	Radio Network Controller
SGSN	Serving GPRS Support Node
S-GW	Serving Gateway
SHF	Super High Frequency
SIM	Subscriber Identity Module
SQN	Sequence Number
TDMA	Time Division Multiple Access
TMSI	Temporary Mobile Subscriber Identity
UE	User Equipment
UHF	Ultra High Frequency
UMTS	Universal Mobile Telecommunication System
VLR	Visitor Location Register
VoIP	Voice over Internet Protocol
WCDMA	Wideband Code Division Multiple Access

Úvod

Od vzniku druhej generácie mobilných sietí, ktorých prítomnosť sa vo svete už pomaly blíži ku koncu, nastala veľká expanzia mobilných zariadení a ich využití. Stále viac ľudí chce byť v spojení s okolitým svetom. Pre dnešného človeka je to až nutnosť. Už sa nestačí len počuť, je potrebné sa aj vidieť, zdieľať takmer každý okamih svojho života. Ľudia používajú takzvané smart zariadenia, ktoré sú schopné vykonať nepredstaviteľné funkcie, avšak potrebujú k tomu prístup k sieti. Sieť tak využívajú ľudia nielen na komunikáciu, ale aj pre riadenie rôznych zariadení pre zábavu, pre kontrolu svojej domácnosti alebo pre výkon svojho zamestnania. Smart zariadenia uľahčujú ľuďom ich prácu, pomáhajú im byť precíznejší a pohotovejší.

2G technológie mali stále svoje využitie vďaka spoľahlivosti prenosu hlasových dát, napríklad pri používaní starších zariadení. V súčasnej dobe sa však stáva väčším prínosom vypnúť 2G a 3G siete a prenechať ich zdroje sieťam ďalších generácií – 4G a 5G – ktoré dokážu uspokojiť potreby toľkých zariadení, ktorými spoločnosť disponuje. Zároveň poskytujú opatrenia v bezpečnosti proti rizikám, s ktorými predchádzajúce generácie ani len nepočítali.

Práca sa zaoberá predovšetkým základnými rozdielmi medzi jednotlivými generáciami mobilných sietí, ktoré sú vysvetlené a ukázané na štandardoch jednotlivých generácií, kryptografických algoritmoch zabezpečujúcich autentizáciu a samotný prenos dát s dôrazom na štvrtú generáciu, pretože 4G siete budú onedlho pravdepodobne jediné fungujúce siete popri 5G sieťach.

Nezameriava sa na mobilné siete generácií pred 2G, teda na tzv. mobilné siete nultej generácie (0G), ktoré sú úplným predchodcom bunkových a bezdrôtových sietí, a na mobilné siete prvej generácie (1G), ktoré využívali analógové spojenie. Taktiež sa nezaoberá sieťami v „medzigeneračnom“ období (2.5G, 2.75G, 3.5G ap.).

Cieľ práce

Cieľom teoretickej časti práce je preskúmať a porovnať druhú až piatu generáciu mobilných sietí, predovšetkým ich odlišnosti v technológiách a bezpečnosti týkajúcej sa autentizácie užívateľov a hlasových a dátových prenosov. Ďalej sa teoretická časť zameriava na projekt srsRAN využívaný pri tvorbe virtuálnej mobilnej siete a na technológie SIM kariet, ktoré sú potrebným prostriedkom k zabezpečeniu mobilného zariadenia, ako aj siete, v ktorej sa zariadenie vyskytuje.

Cieľom praktickej časti je navrhnúť a implementovať nástroj, ktorý demonštruje bezpečnostné operácie prítomné v sieťach štvrtej generácie, generuje a dopočítava ich potrebné parametre. Ďalším cieľom praktickej časti je integrácia implementovaného nástroja do existujúceho projektu, ktorý sa zaoberá penetračným testovaním IoT zariadenia v testovacej 4G sieti.

Výstup práce bude prínosný pre testovacie účely založené na penetračnom testovaní v mobilných sieťach vďaka vygenerovaným a predpočítaným parametrom potrebných pre autentizáciu a zabezpečenie, pre študentov FIT ČVUT a populáciu zaujímajúcu sa o problematiku bezpečnosti mobilných sietí.

Kapitola 1

Základy mobilných sietí

Mobilná sieť sa stará o pripojenie a správu mobilného zariadenia v nej prostredníctvom rádiových vĺn. Za správny chod siete sú zodpovedné komponenty, ktoré sú súčasťou jadra siete. Komunikáciu medzi mobilným zariadením a jadrom siete zabezpečuje tzv. sieť s rádiovým prístupom (RAN). [1]

1.1 Prvky mobilnej siete

Aj napriek tomu, že sa funkcie, usporiadania a názvy koncového zariadenia s prístupom do mobilnej siete (mobilného zariadenia užívateľa), RAN a jadra siete naprieč generáciami vyvíjajú a menia, sú základom každej generácie mobilných sietí. Ich konkrétne zmeny v jednotlivých generáciach sú predstavené v ďalšej kapitole, najmä na konkrétnych známych štandardoch.

1.1.1 Koncové mobilné zariadenie

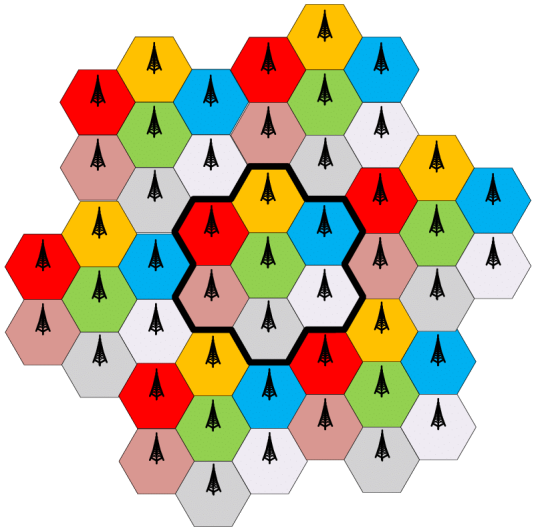
Mobilné zariadenie užívateľa poskytuje užívateľovi rozhranie, prostredníctvom ktorého mu je umožnený prístup k sieti a jej službám. Za prístup do mobilnej siete zodpovedá najmä integrovaný obvod, ktorým je identifikačná karta predplatiteľa mobilných služieb – karta SIM (Subscriber Identity Module).

1.1.1.1 SIM karta

SIM karta je integrovaný obvod, ktorý spracováva hodnoty kryptografických dát pomocou kryptografických funkcií a algoritmov. Z veľkej časti vykonáva funkcie rovnaké ako v autentizačnom centre v jadre mobilnej siete, kde predpočítava ďalšie parametre potrebné predovšetkým pre autentizáciu a zabezpečenie komunikácie. Ďalej obsahuje uloženú hodnotu IMSI (International Mobile Subscriber Identity), ktorá identifikuje užívateľa ako predplatiteľa služieb siete.

Okrem zabezpečenia prístupu k sieti a komunikačného prenosu, slúži aj k zabezpečeniu mobilného zariadenia prostredníctvom PIN kódu. Ide o štvromiestny kód, ktorý je vopred zvolený výrobcom, avšak užívateľ je schopný si ho zmeniť. Niekoľko neúspešných pokusov PIN kódu vedie k dotázaniu sa na tzv. PUK kód, ktorý je osemmiestny a nie je možné ho zmeniť. Jeho neúspešné zadanie vedie k zablokovaniu zariadenia.

Algoritmy, na ktorých sa SIM karty podieľajú v rámci bezpečnosti, sú predstavené v následujúcom texte v jednotlivých generáciách.



Obr. 1.1 Vzorové rozmiestnenie frekvencií pri ich znovupoužití v sieti, aby sa zabránilo vzájomnej interferencii (pri použití 7 frekvenčných kanálov) [2]

1.1.2 RAN

Hlavnou úlohou RAN je tvoriť spojenie medzi užívateľom a sieťou. Za samotné prijímanie a odosielanie dát a signálov od užívateľa k jadru siete (uplink) a naopak (downlink) sú zodpovedné základňové stanice, ktoré RAN tvoria. Práve základňové stanice vysielajú rádiové vlny, ktoré sú nosičmi informácií a signálov. Vďaka rozmiestneniu základňových staníc, ktoré svojím dosahom signálu vytvárajú menšie bunky, je možné presúvať sa medzi týmito bunkami a zostať stále v spojení. Preto sa niekedy mobilná sieť nazýva aj bunková. Aby sa zabránilo k vzájomnému rušeniu vysielaného rádiového signálu – interferencii – medzi bunkami je dôležité, aby základňové stanice v bunkách, ktoré spolu susedia, disponovali odlišnými frekvenčnými kanálmi. [1]

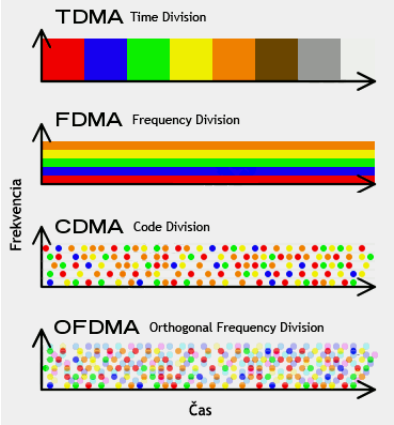
1.1.2.1 Frekvencia

Okrem toho, že výber frekvencie hrá rolu pri zabránení interferencie, správne frekvenčné kanály garantujú tiež silný stabilný signál a veľkú kapacitu prenosu. Rádiové vlnenie je súčasťou elektromagnetického spektra v rozsahu 3 kHz - 300 GHz. Frekvencie využívané v mobilných sieťach sú vyberané z pásem s vyššou frekvenciou tohto rádiového vlnenia, z pásma ultra vysokej frekvencie (UHF), 0,3–3 GHz, super vysokej frekvencie (SHF), 3–30 GHz, a extrémne vysokej frekvencie (EHF), 30–300GHz. Každé z týchto pásem je možné inak využiť, pretože majú rozličné vlastnosti, a teda sú každé vhodné na iný účel. Pásma s vyššou frekvenciou, teda vyšším počtom kmitov za sekundu, majú vyššiu kapacitu prenosu, ale menší dosah. Naopak pásma s nižšou frekvenciou sú schopné pokryť väčšie územie, avšak ich kapacita prenosu je menšia. [3]

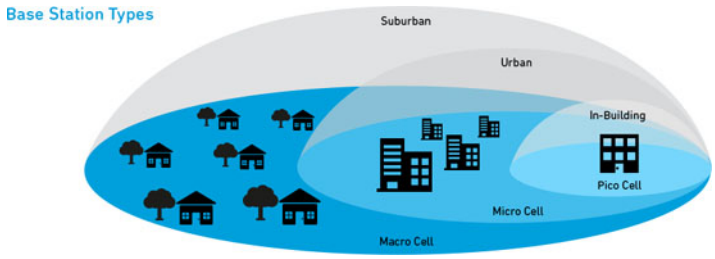
Aby mohlo mobilnú sieť využívať viacero užívateľov súčasne, je potrebné využívať viacnásobný prístup k zdieľanému médiu, tzv. Multiple Access (MA). Mobilné siete využívajú jeden typ alebo kombináciu týchto typov k oddeleniu viacnásobného prístupu.

Frequency Division Multiple Access (FDMA) využíva fakt, že pásmo je zhlukom viacerých frekvencií, a preto je možné celkovú šírku pásma rozdeliť do menších kanálov, ktoré sa dajú rozdeliť medzi jednotlivých užívateľov.[4]

Time Division Multiple Access (TDMA) všetci užívatelia využívajú celé dostupné frekvenčné pásmo, ale dostupnosť pre jednotlivých užívateľov je riadená časom. Časové intervaly pridelené užívateľom sú veľmi malé, a tak to užívateľ ani nezaregistruje.



Obr. 1.2 Typy delenia viacnásobného prístupu k jednému médiu najčastejšie využívané bezdrôtových technológiách [7]



Obr. 1.3 Typy buniek vyobrazené spolu s ich využitím [8]

Code Division Multiple Access (CDMA) komunikácia užívateľov je možná na celej šírke pásme v rovnaký čas, pretože táto technika využíva rozprestretie komunikačného kanálu prostredníctvom určitej pseudonáhodnej kódovej sekvencie, ktorá je vytvorená oveľa vyššou frekvenciou. [5]

Orthogonal Frequency Division Multiple Access (OFDMA) využíva metódu OFDM, ktorá svojím princípom šetrí šírku pásma. OFDM, využívané pri jednom užívateľovi, rozdeľuje frekvencie na menšie kanály rovnako ako FDM, avšak narozdiel od FDM, kde sa nesmú tieto kanály prekrývať, pri OFDM sa tieto subkanály modulujú, aby boli navzájom ortogonálne. To znamená, že v mieste, kde jeden subkanál dosahuje svoj vrchol, ostatné sú v tomto mieste na hodnote nula. Tým sa tiež zabráni interferencii. OFDMA, ako už bolo spomenuté, je delenie prístupu viacerých osôb založené na princípe OFDM, ktoré poskytuje užívateľom možnosť využívať médium aj v rovnakom čase vďaka úspornejšej alokácii kanálov. [6]

1.1.2.2 Bunky mobilnej siete

Ako už bolo spomenuté vyššie, bunkové stanice sú práve tými vysielačmi a prijímačmi rádiového signálu, ktoré prenášajú signály a dáta, a že ich rozmiestnenie a vlastnosti sú veľmi dôležité pre efektívne pokrytie s vysokou kapacitou prenosu. Na základe vlastností frekvencie, poveternostných vplyvov, terénu, prekážok, preťaženia siete a iných faktorov, mobilní operátori poskytujúci svoje služby stoja vždy pred otázkou, aký typ bunky je pre aké využitie vhodný. Správne rozhodnutie v tejto otázke môže vytvoriť rovnováhu medzi nákladmi operátora a uspokojením potrieb užívateľa koncového zariadenia.

Makro bunky sa umiestňujú predovšetkým v riedko osídlených oblastiach, v dedinách, na lúkach a podobne. Ide o bunky, ktoré tvoria rozmerovo veľké základňové stanice na vysokých miestach, napr. na kopcoch, kvôli potrebnému výkonu na vysielanie. Signál makro bunky dosahuje až desiatky kilometrov, teda pokrýva veľmi veľkú oblasť, ale nemá vysokú prenosovú kapacitu.

Mikro bunky sú rozprestreté väčšinou po meste v podobe základňových staníc umiestnených na budovách. Dosah týchto buniek je v stovkách metrov až jednotkách kilometrov. V mestách sa kvôli signálu, hustému osídleniu a zastavaniu vyskytuje hustejšie pokrytie menšími základňovými stanicami, ako je to s makro bunkami v oblastiach s riedkym osídlením.

Pico bunky poskytujú vhodné pokrytie vo veľkých budovách s vysokou hustotou užívateľov, v nákupných centrách, na letisku alebo na železničných staniciach. Ich signál siaha do diaľky jednotiek až stoviek metrov.

Femto bunky sa využívajú pre pokrytie domova alebo kancelárií. Jedna femto bunka vie pokryť približne 10 metrov. Veľkou výhodou je malá spotreba, ale dostatočná kapacita prenosu. Boli navrhnuté najmä ako výplň dier medzi pokrytím iných buniek.

„Dáždnikové“ bunky zabezpečujú pripojenie v miestach, kde by sa užívateľ pripojený k sieti pohyboval veľmi rýchlo medzi bunkami. „Dáždnikové bunky“ sú v skutočnosti viaceré mikro bunky umiestnené vnútri jednej makro bunky. Mikro bunky sú vhodné pre užívateľa, ktorý sa pohybuje po oblasti pomaly. Ak sa užívateľ pohybuje rýchlo, prichádza neustále k odovzdávaniu komunikačného kanálu a môže prísť k prerušeniu spojenia. Avšak tu sa makro bunka postará o to, že užívateľ bude naďalej pripojený, aj keď už opustí priestor so signálom mikro bunky. [9], [10]

1.1.3 Jadro mobilnej siete

Jadro mobilnej siete zabezpečuje služby koncovým užívateľom, riadi výmenu informácií vnútri siete ako aj medzi svojou domovskou a externou sieťou. Uchováva informácie o užívateľoch, zabezpečuje ich prístup k sieti, samotnú komunikáciu a dáta služieb. Je zodpovedné za sledovanie polohy užívateľa, za dodržiavanie politík siete, ale aj kvality služieb.

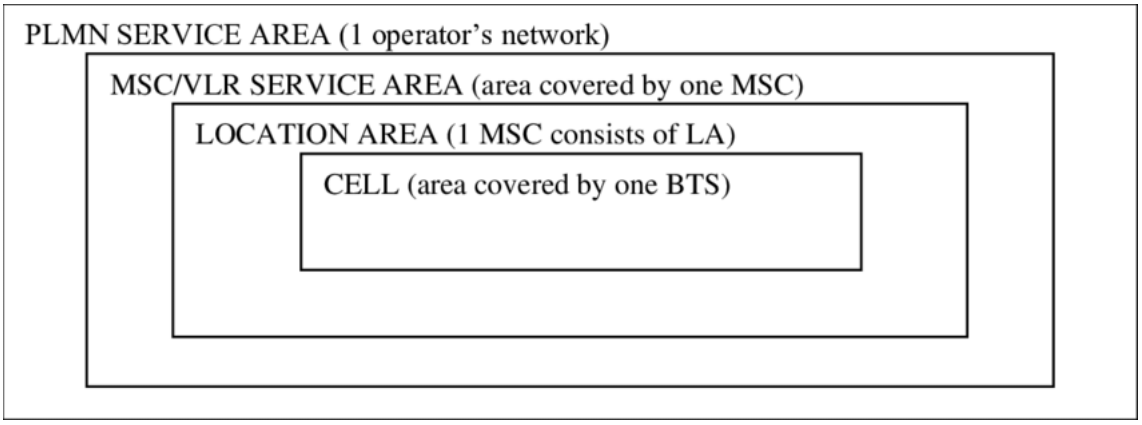
1.2 Mobilita v sieti

Vďaka štruktúre mobilnej siete je umožnený užívateľom pohyb po rozľahlej geografickej oblasti a mať stále prístup k sieti a jej službám takmer bez prerušenia. Preto je najskôr potrebné pochopiť hierarchickú štruktúru siete.

Každý operátor obstaráva jednu tzv. PLMN (Public Land Mobile Network) oblasť, v ktorej sú obsluhujúcim uzlom (v 2G sieti GSM uzol MSC – Mobile Switching Centre) priradené určité územia LA (Location Area). V týchto oblastiach je jedna alebo viacero ovládačov základňových staníc (napr. v GSM ovládač BSC), ktoré obsluhujú jednotlivé bunky. [11]

V rámci jednej LA sa o zmenu polohy zariadenia starajú samotné ovládače a základňové stanice medzi sebou prostredníctvom tzv. handover (odovzdávajúceho) procesu. V prípade zmeny oblasti LA je potrebné informovať zároveň aktuálne obsluhujúci uzol a k nemu priradené registre (v GSM HLR – Home Location Register – a VLR – Visitor Location Register), ktoré uchovávajú polohu zariadenia vo forme LAI (Location Area Identity). LAI je zložený z PLMN identifikátoru, ktorý je tvorený MCC (Mobile Country Code) a MNC (Mobile Network Code). Okrem obsluhujúcej siete je LAI uložené tiež na SIM karte zariadenia.

Každá základňová stanica vysiela v pravidelných intervaloch tzv. BCCH (Broadcast Control



Obr. 1.4 Hierarchia štruktúry siete [11]

Channel). Pomocou neho informuje zariadenia v jej blízkosti o identifikátore oblasti LAI. Zariadenie kontroluje, či sa LAI vysielané základňovou stanicou a LAI uložené na SIM karte zhodujú. V prípade odlišných identifikátorov LAI je zahájená procedúra o aktualizácii polohy (Location Update Procedure), pri ktorej dôjde tiež k aktualizácii polohy v regiistroch obsluhujúcej siete. [12]

Kapitola 2

Generácie mobilných sietí

2.1 2G

2G predstavuje generáciu, ktorá ako prvá priniesla do mobilných sietí digitálny signál. Spolu s tým 2G spôsobila aj obrovský rozkvet mobilných sietí. Od roku 1991, kedy bola spustená prvá sieť druhej generácie s jej najznámejším štandardom – GSM, takmer každých desať rokov prišla na svet ďalšia generácia s novými funkciami, rýchlejším prenosom a nižšou latenciou.

Štandard GSM je však aj napriek tomu najstarší a najdlhšie využívaný štandard. Až s očakávaným príchodom 5G sietí sa siete GSM pomaly blížia ku koncu. Aj keď väčšina ľudí má väčšie a väčšie nároky na rýchlosť, priepustnosť alebo nové funkčnosti, stále sú aj užívatelia využívajúci len funkčnosti GSM, teda hlasové hovory alebo posielanie SMS správ. Aby mohol užívateľ používať siete novších generácií, je potrebné, aby to jeho zariadenie podporovalo, čo najmä pre staršie generácie populácie môže predstavovať problémy.

GSM siete vysielajú na úzkopásmových kanáloch, ktoré sa delia na časové úseky pridelené jednotlivým užívateľom, čiže využívajú metódu TDMA. Konkrétne sú základňovým staniciam priradené kanály o šírke 200 kHz, kde každý takýto kanál je rozdelený po 25 kHz pre každého užívateľa. Takto je jeden frekvenčný kanál schopný obslúžiť 8 užívateľov, pričom každá základňová stanica je zväčša vybavená viacerými úzkopásmovými kanálmi, takže na jednej základňovej stanici môže komunikovať viacero užívateľov.

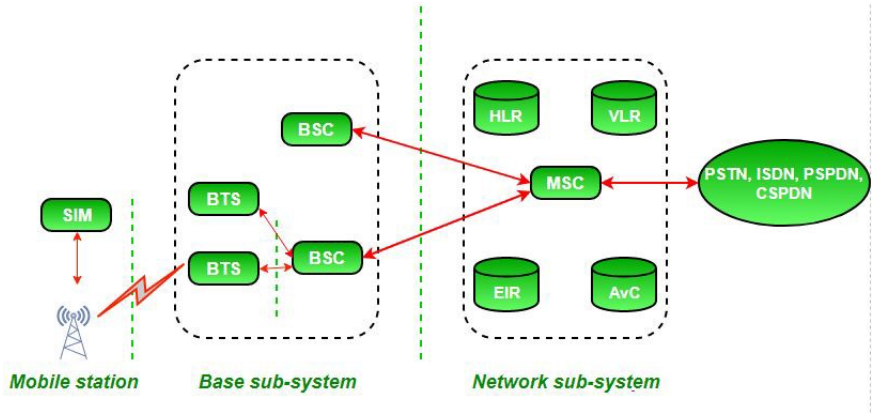
Sieťam GSM štandardu sa priraďujú frekvenčné pásma v okolí 900 MHz a 1800 MHz, ktoré sa mierne líšia v závislosti na krajine alebo dokonca na operátoroch danej krajiny. [13]

2.1.1 Architektúra GSM

Komponenty GSM siete sa dajú rozdeliť do 4 hlavných subsystémov – MS, BSS, NSS a OSS.

MS , alebo koncové zariadenie, je zariadenie, s ktorým sa dostáva do styku užívateľ a môže sa prostredníctvom neho pripojiť do siete a komunikovať s ostatnými koncovými zariadeniami ostatných užívateľov. Okrem samotného mobilného zariadenia, ktoré je vlastne rozhraním medzi sieťou a užívateľom, obsahuje MS ešte SIM kartu. Obe tieto komponenty obsahujú svoj identifikátor, o ktorom užívateľ väčšinou ani nevie a nie je možné ich zmeniť. U MS hovoríme o IMEI (International Mobile Equipment Identity) – slúži na identifikovanie zariadenia, u SIM karty o IMSI.

BSS je v podstate RAN štandardu GSM, ktorú tvoria základňové stanice, tu nazývané skrátene BTS, ktoré kontroluje a ovláda BSC. BTS vytvarajú rádiové spojenie medzi MS a NSS.



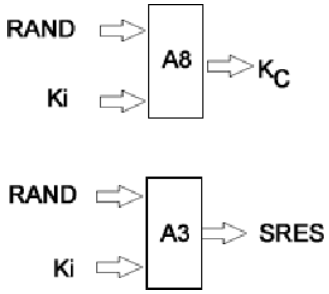
Obr. 2.1 Schéma zobrazuje komponenty architektúry GSM rozdelené do jednotlivých subsystémov [14]

Aktivita niekoľkých BTS je kontrolovaná jednou BSC, ktorá sa stará tiež o odovzdávanie ich obsluhovaných kanálov medzi sebou.

NSS ako subsystém, ktorý predstavuje jadro siete, obsahuje hlavné prvky zodpovedné za riadenie siete. Najdôležitejším riadiacim prvkom je MSC tvoriace prepojenie medzi BSS, HLR, VLR, EIR a GMSC (Gateway Mobile Switching Center).
HLR je z veľkej časti permanentné úložisko obstarávajúce informácie o predplatiteľoch a ich poslednú polohu, ktoré je napojená na AuC starajúce sa o dáta potrebné k autentizácii užívateľa a šifrovaniu komunikácie v sieti. VLR spojená s MSC udržiava informácie o užívateľoch, ktorí sa zdržiavajú v oblasti danej MSC, ktorú táto VLR spravuje. Záznam užívateľa vo VLR vzniká ako kópia z jeho domovskej HLR, zároveň priraďuje pre komunikáciu užívateľovi TMSI (Temporary Mobile Subscriber Identity – dočasný identifikátor mobilného predplatiteľa), aby skutočná IMSI nebola v komunikácii využívaná. EIR je databázou pre identifikačné čísla mobilných staníc, ktoré majú povolenie pripojiť sa k sieti bez špeciálnych podmienok (white list), sú podozrivé, a preto sú sledované (grey list) alebo nemajú povolené pripojenie k sieti (black list). Rozhraním spájajúcim užívateľa mobilnej siete s užívateľom v inej mobilnej sieti je GMSC. [13]

2.1.2 Bezpečnosť GSM

GSM zaručuje bezpečnosť svojím užívateľom prostredníctvom dodržiavania ich anonymity a súkromia, prístupu k službám siete len autorizovaným užívateľom siete a šifrovaného prenosu. Čo však GSM nepodporuje je vzájomná autentizácia, to znamená, že užívateľ sa musí voči sieti preukázať autentizáciou, ale sieť je považovaná za dôveryhodnú bez jej overenia.Ďalšou jej slabou stránkou je používanie slabých funkcií triedy COMP128, ktoré sa považujú za bezpečné hlavne vďaka princípu „Security by Obscurity“. Teda bezpečnosť zaručuje len skrytie a utajenie algoritmov pred verejnosťou. V GSM sa využívajú algoritmy COMP128 na autentizáciu, presnejšie ide o algoritmy A3 a A8, ktorých implementáciu a konkrétne vykonávaný algoritmus si volí samotný operátor. Niekedy môžu byť použité aj pre iných užívateľov iné algoritmy v rámci siete jedného operátora. [15]



Obr. 2.2 Vyobrazenie vstupov a výstupov algoritmov A8 (tvorba šifrovacieho kľúča) a A3 (autentizácia) [16]

2.1.2.1 Autentizácia a tvorba kľúčov – algoritmy COMP128 A3 a A8

Skôr ako môže užívateľ začať využívať služby siete, musí prebehnúť jeho autentizácia. Tá prebieha vytvorením komunikačného kanálu medzi MS a NSS, a následným poslaním IMSI od MS k NSS. V AuC sa ďalej vyhľadá na základe tohto identifikátoru príslušný tajný 128bitový kľúč (Ki), ktorý je uložený aj v SIM karte MS. AuC zároveň vygeneruje 128bitovú náhodnu postupnosť (RAND) a ako tzv. „Challenge“ pošle RAND k MS. Obe tieto hodnoty sú vstupom pre algoritmus A3, ktorý ich spracuje a výstupom je 32bitová hodnota nazývaná SRES. Pošle ju do AuC siete, ktoré medzitým taktiež spracovalo hodnoty Ki a RAND rovnakým algoritmom, aby po prijatí SRES od MS bolo schopné vyhodnotiť, či sa zhodujú. Ak sa tieto hodnoty rovnajú, užívateľ je vpustený do siete, dočasný kanál priradený pre autentizáciu je zrušený a vytvorí sa nový kanál len medzi MS a BSS určený pre následnú šifrovanú komunikáciu.

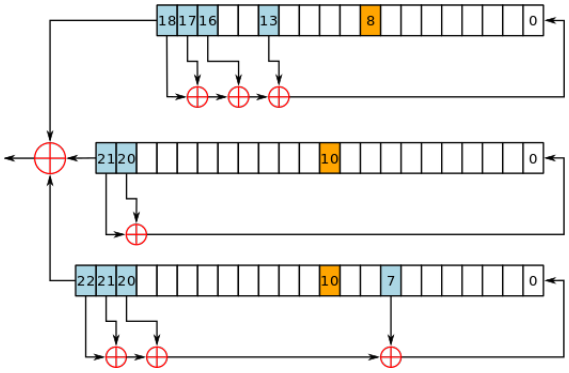
Predtým, ako môžu byť posielané dáta komunikácie, je potrebné ich zašifrovať. Pri každom naviazaní spojenia v sieti, sa pre MS tvorí nový 128bitový kľúč Kc určený k šifrovaniu posielaných dát. Za vygenerovanie Kc je zodpovedná druhá z funkcií COMP128, A8. Tá rovnako ako funkcia A3 prijíma na vstupe Ki a RAND, avšak na výstup posiela 64bitový šifrovací kľúč Kc. [15]

2.1.2.2 Šifrovanie – algoritmus COMP128 A5/1

Po autentizácii a vytvorení Kc, je možné zahájiť komunikáciu, a vysielať tak jej dáta. Tie sa pred odoslaním do BSS šifrujú. Algoritmy na šifrovanie dát GSM sú známe aj konkrétnou implementáciou. Využívajú sa verzie algoritmu A5 – A5/0, A5/1, A5/2 a A5/3. Avšak najznámejšia je verzia A5/1. Pre šifrovanie algoritmus prijíma 22bitový číslo frame-u Fn a Kc. Po ich prijatí vygeneruje 228bitovú pseudonáhodnú postupnosť PRAND, ktorá sa rozdelí na dve 114bitové postupnosti, jedna pre uplink a jedna pre downlink. Konečný výsledok šifrovania je výsledkom operácie XOR s hodnotami PRAND a nešifrovaných dát na vstupe.

Celý proces začína inicializáciou 3 posuvných registrov (R1 s dĺžkou 19 bitov, R2 s dĺžkou 22 bitov a 23bitový register R3) so spätnou väzbou. Táto inicializácia spočíva v 4 krokoch.

1. Všetky registre sa vyplnia nulami
2. V 64 taktoch sa vždy aplikuje operácia xor na počiatočný bit každého registru a nasledovný bit kľúča Kc (po každom takte sa posúvajú všetky registre)
3. V 22 taktoch sa vykoná 22krát operácia xor s počiatočným bitom každého registru a nasledovným bitom čísla Fn (po každom takte sa posúvajú všetky registre)
4. 100 taktov prebehne spôsobom, akým je A5/1 definovaná (vysvetlené nižšie) a výstup sa zahodí



Obr. 2.3 Prúdová šifra A5/1 využívajúca 3 posuvné registre s lineárnou spätnou väzbou [18]

Spôsob posunu registrov je definovaný s pomocou tzv. majoritnej funkcie. Každý register má určený taktovací bit (R1[8], R2[10], R3[10]), ktorý môže mať hodnotu jedna alebo nula. To znamená, že vždy jedna hodnoty musí prevládať. Registre s prevládajúcou hodnotu na taktovacom bite sa posunú a určia sa hodnoty na ich počiatočných pozíciách pomocou operácie xor definovenj na bitoch spätnej väzby ($R1[0] = R1[13] \oplus R1[16] \oplus R1[17] \oplus R1[18]$, $R2[0] = R2[20] \oplus R2[21]$ a $R3[0] = R3[7] \oplus R3[20] \oplus R3[21] \oplus R3[22]$). Výstupom jedného taktu je jeden bit určený bitmi s najvyšším indexom ako $out = R1[18] \oplus R2[21] \oplus R3[22]$. [17]

2.2 3G

S príchodom sietí tretej generácie na začiatku roku 2000, prišiel do života ľudí prostredníctvom mobilných zariadení aj prístup na internet. Vďaka vyššej rýchlosti prenosu dát, tak 3G siete začali umožňovať okrem využívania hlasových služieb aj aplikácie, ktoré poskytujú aj tzv. služby v reálnom čase. Jedná sa o služby, ktoré sa stali neoddeliteľnou súčasťou nášho života, určenie polohy pomocou GPS, video hovory alebo okamžité správy (instant messaging). [19]

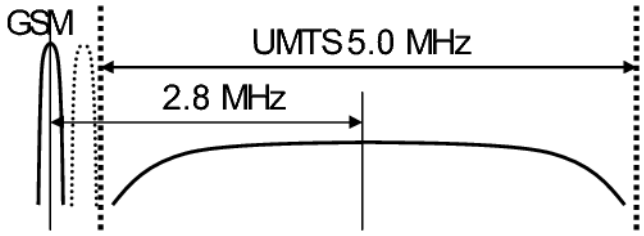
Najpoužívanejším európskym štandardom 3G sietí, ktorý položil základy pre ďalší vývoj v technológiách mobilných sietí, je UMTS využívajúci WCDMA – metódu delenia viacnásobného prístupu rozširujúcu CDMA. [20]

UMTS siete vysielajú vo frekvenčných pásmach 900 MHz a 2100 MHz. Ako bolo vyššie spomenuté v 2.1 GSM sieťam je taktiež priradené pásmo 900 MHz a keďže GSM siete vysielajú aj naďalej po vzniku 3G sietí, je potrebné zaistiť, aby sa príjem a vysielanie signálu týchto dvoch sietí nerušili.

Uvádzajú sa dva základné prípady, ku ktorým môže dôjsť. Ide o prípad nekoordinovanej alebo koordinovanej koexistencie týchto dvoch sietí. V oboch prípadoch je potrebné jednotlivé nosné frekvencie oddeliť ochranným pásmom (guard band). V prípade UMTS kanála širokého 5 MHz a GSM kanála širokého 200 kHz ide pri nekoordinovanom nasadení o vzdialenosť šírky 2,8 MHz (ochranné pásmo o šírke jedného 200 kHz GSM kanála). Pri koordinovanej koexistencii a rovnakých šírkach GSM a UMTS kanála, môže byť táto vzdialenosť menšia ako 2,8 MHz. [21]

2.2.1 Architektúra UMTS

Koncové zariadenia využívajúce služby siete získali nové pomenovanie – z mobilnej stanice MS sa stali užívateľské zariadenia UE – rovnako ako sa premenovala RAN z BSS na UTRAN (UMTS Terrestrial Radio Access Network) a jadro siete sa rozdelilo na domény prepojovania paketov (Packet-Switched Domain) a prepojovania okruhov (Circuit-Switched Domain).



Obr. 2.4 Oddelenie nosnej frekvencie pri spolunasadení GSM 900 a UMTS 900 [21]

UE , ako užívateľské zariadenie, rovnako ako v predchádzajúcej generácii obsahuje identifikačnú kartu na identifikáciu predplatiteľa siete SIM, známu ako USIM. USIM je potrebná najmä k bezpečnosti – k autentizácii, autorizácii a šifrovaniu spojenia.

UTRAN sa skladá z RNC (Radio Network Controller) uzlov obstarávajúce základňové stanice nazývané NB (Node BTS). Vykonáva podobné funkcie ako BSS, avšak RNC sú okrem kontroly základňových staníc zodpovedné za správne preposielanie dát do MSC (riadiaci uzol CS domény) alebo do SGSN (riadiaci uzol PS domény).

CN (CS Network) je časť siete zaobstarávajúca služby, ktoré poskytuje jadro siete GSM (prenos hlasových dát a preposielanie SMS správ). Skladá sa z komponent MSC a GMSC, kde MSC je riadiaci uzol v tejto časti siete a GMSC slúži ako rozhranie k externým sieťam. Okrem týchto komponent zdieľa s PN (PS Network) HLR obsahujúce AuC a VLR, ktoré majú rovnakú funkciu ako v sieťach GSM.

PN zabezpečuje spracovanie a preposielanie paketov obsahujúcich dáta iných služieb než CN. Tu sú komponenty, ktoré pretrvali z „medzigeneračnej siete“ GPRS, SGSN (Serving GPRS Support Node) a GGSN (Gateway GPRS Support Node). Prvý z nich obstaráva rovnako MSC v CN spoluprácu komponent vnútri siete, spravuje relácie, kvalitu služieb a správu mobility zariadenia/užívateľa. Druhý, GGSN, je rozhraním medzi PN sieťou a externými sieťami. Slúži ako smerovač (router) siete. [22]

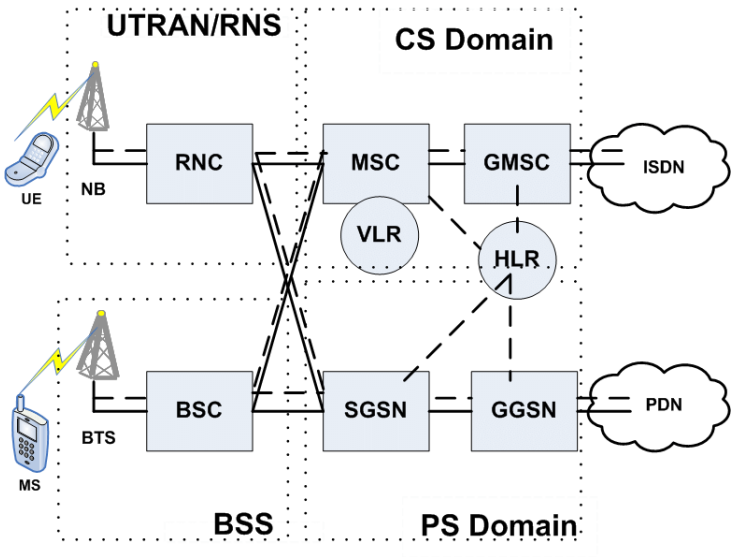
2.2.2 Bezpečnosť UMTS

UMTS štandard narozdiel od 2G zaisťuje vzájomnú autentizáciu, teda pred pripojením zariadenia do siete a zahájením jeho komunikácie, sa neautentizuje len užívateľ voči sieti, ale v rámci siete prebieha autentizácia aj na strane domovského registra. Spolu s autentizáciou sa generujú kľúče pre ďalšie zabezpečenie. Tento proces je tiež označovaný ako AKA (Authentication and Key Agreement). [24]

2.2.2.1 Autentizácia a tvorba kľúčov – algoritmus MILENAGE

AKA začína výzvou obsluhujúcej siete smerom k AuC v HE. Tu sa vygeneruje určitý počet autentizačných vektorov AV, ktoré obsahujú RAND (náhodné číslo vygenerované autentizačným centrom), XRES (očakávaná odpoveď zo strany UE použitá na autentizáciu užívateľa), kľúče CK a IK (na šifrovanie a integritu) a AUTN (autentizačný token vytvorený následujúcim spôsobom $AUTN = SQN \oplus AK||AMF||MAC$). Parametre autentizačného vektoru sú použité pre zachovanie správneho poradia bezpečnostných parametrov (sekvenčné číslo SQN), na utajenie SQN pomocou kľúča pre anonymitu AK, zvolenie správnych funkcií (pole pre správu autentizácie AMF) a pre overenie autentizačného centra (autentizačný kód správy MAC).

Tvorené hodnoty v AV sú vytvorené v AuC algoritmom MILENAGE určeným štandardom.



Obr. 2.5 Komponenty architektúry UMTS prepojené s RAN siete GSM (BSS) rozdelené do jednotlivých subsystémov a domén s prepojovaním okruhov (CS Domain) a paketov (PS Domain) [23]

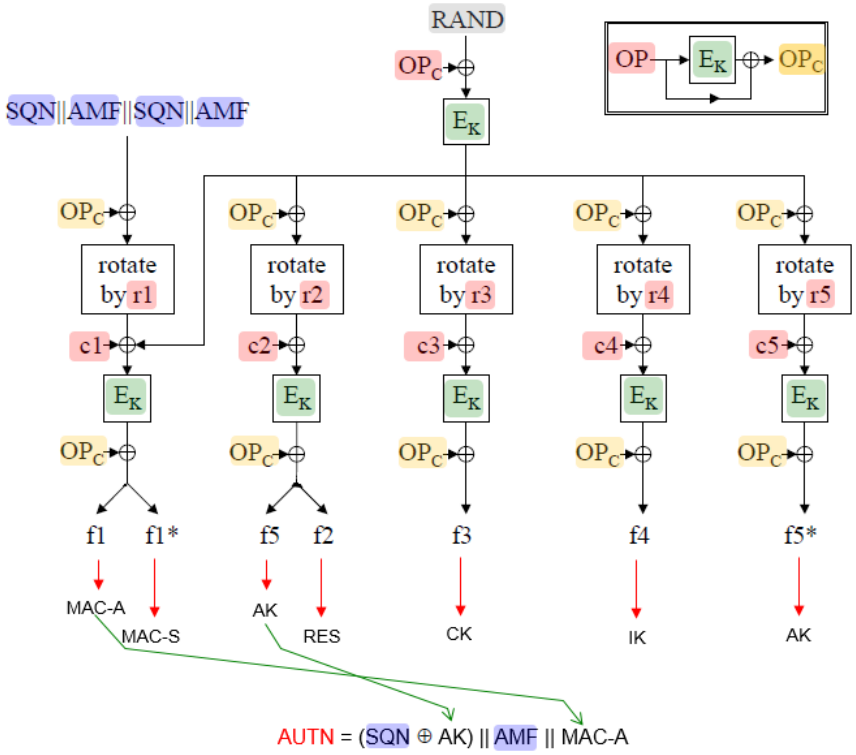
Tabuľka 2.1 Funkcie definované v algoritme MILENAGE

Funkcia	typ funkcie	vstup	výstup
f1	funkcia overenia správy	RAND, K, SQN, AMF	MAC/XMAC
f1*	funkcia overenia správy (pri resynchronizácii)	RAND, K, SQN, AMF	MAC/XMAC
f2	funkcia overenia správy	RAND, K	XRES/RES
f3	funkcia generujúca kľúč	RAND, K	CK
f4	funkcia generujúca kľúč	RAND, K	IK
f5	funkcia generujúca kľúč	RAND, K	AK
f5*	funkcia generujúca kľúč (pri resynchronizácii)	RAND, K	AK

MILENAGE vykonáva autentizačné funkcie aj funkcie generujúce a derivujúce kľúče. V algoritme MILENAGE sa hovorí o siedmich funkciách označovaných ako f1, f1*, f2, f3, f4, f5 a f5*.

Špecifikáciou definovaný MILENAGE používa pre šifrovanie E_k algoritmus Rijndael. Každý operátor je povinný kvôli bezpečnosti zabezpečiť špecifikáciou definované normy, ale aby došlo k určitej modifikácii medzi operátormi, je využívaný kód operátora OP, v praxi častejšie derivovaný kód operátor OPC ($OPC = E_k(OP, K) \oplus OP$), a tiež si môže operátor zvoliť hodnoty konštánt c1 – c5 a r1 – r5, ktoré v tomto algoritme figurujú pre operácie XOR a rotáciu.

Po tom, ako SN prijme n-ticu AV, vyberie vhodný (následujúci) vektor a prislúchajúcu postupnosť RAND a token AUTN pošle smerom k UE. AUTN je hodnota, ktorá je nereplikovateľná vďaka sekvenčnému číslu SQN a nevypočítateľná pre toho, kto nepozná kľúč K. UE po prijatí skontroluje, či je AUTN validné. K tomu je však najskôr potrebné vypočítať kľúč AK rovnako, ako ho počíta AuC siete – funkciou f2. Na vypočítanú hodnotu AK aplikuje USIM operáciu XOR spolu s časťou AUTN – $SQN \oplus AK$, čím dosiahne výslednej hodnoty SQN. Túto hodnotu využije pri spracovaní funkcie f1 k získaniu očakávaného kódu autentizačnej správy XMAC a overí ju s prijatou hodnotou MAC. Po úspešnom overení, overí ešte, či sa SQN nachádza v správnom rozsahu. Pokiaľ nie, pošle žiadosť o resynchronizáciu. V opačnom prípade dopočíta ostatné hodnoty, vrátane RES následne poslanej do SN, ktoré zistí, či sa RES z UE a XRES z HE zhodujú. Ak áno, užívateľ je vpustený do siete a môže využívať jej služieb. [24]



Obr. 2.6 Algoritmus MILENAGE určený pre autentizáciu a tvorbu kľúčov [25]

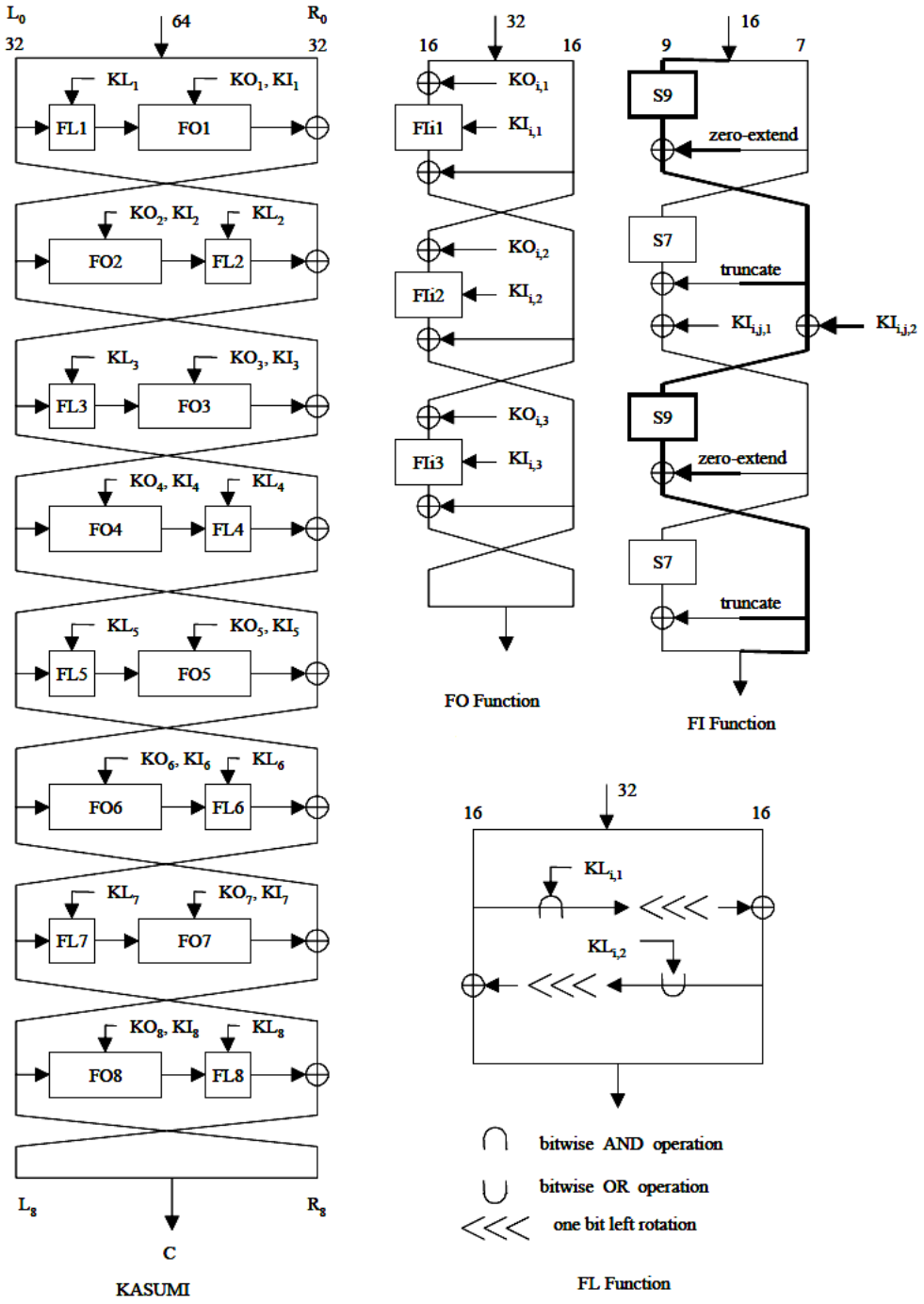
2.2.2.2 Šifrovanie – algoritmus KASUMI

Pre šifrovanie a zachovanie integrity dát sa využíva algoritmus KASUMI. Jedná sa o blokovú šifru založenú na bloku o dĺžke 64 bitov a 128bitovom kľúči. Pôvodné požiadavky stanovujú, že šifrovacia funkcia, f8, má byť prúdovou šifrou a funkcia pre integritu, f9, má byť funkciou MAC, teda obdobou hashovacích funkcií. Aby bola dodržaná prúdovosť funkcie f8, využívajú sa pri spracovaní blokov módy CFB alebo OFB s CTR módom. Funkcia f9 vytvára MAC pomocou CBC-MAC módu. [24]

KASUMI šifruje bloky v ôsmich rundách šifrou Feistelovho typu. To znamená, že pred vykonávaním šifry je blok rozdelený na dve rovnomerné časti – po 32 bitoch – L_0 a R_0, ktoré sú v každej runde spracované definovaným spôsobom ako $R_i = L_{i-1}$ a $L_i = R_{i-1} \oplus f_i(L_{i-1}, RK_i)$ pre $i \in \{1, 2, .., 8\}$ predstavujúce číslo rundy, kde f_i je funkcia závislá na párnosti čísla rundy a RK_i je rundovný kľúč, ktorý označuje trojicu kľúčov KL_i, KO_i, KI_i prislúchajúce k podfunkciám FL, FO, FI. Nepárne rundy (1, 3, 5 a 7) aplikujú najskôr na vstup funkciu FL za pomoci kľúča KL_i a na tento výstup funkciu FO s použitím kľúčov KO_i a KI_i – teda nepárne funkcie majú tvar $f_i(input, RK_i) = FO(FL(input, KL_i), KO_i, KI_i)$. Párne rundy (2, 4, 6 a 8) posielajú vstup do funkcií v opačnom poradí spolu s ich príslušnými kľúčmi – tvar párnych funkcií je $f_i(input, RK_i) = FL(FO(input, KO_i, KI_i), KL_i)$.

Funkcia FL rozdelí 32bitový vstup na dva 16bitové vstupy L a R, rovnako tak 32bitový kľúč KL_i rozdelí na dva 16bitové podkľúče $KL_{i,1}$ a $KL_{i,2}$. Výstupom je reťazec $L'R'$, pričom $R' = R \oplus ROL(L \cap KL_{i,1})$ a $L' = L \oplus ROL(R' \cup KL_{i,2})$, kde operácia ROL je ľavá kruhová rotácia o 1 bit a operácie $\cap$ a $\cup$ sú bitový AND a bitový OR.

Funkcia FO spracováva 32bitový vstup rozdelený na dve polovice L_0 a R_0 s pomocou dvoch 48bitových kľúčov KO_i a KI_i rozdelených do trojíc 16bitových kľúčov $KO_{i,1}$, $KO_{i,2}$, $KO_{i,3}$ a $KI_{i,1}$, $KI_{i,2}$, $KI_{i,3}$. Následne je definové pre každé $j \in \{1, 2, 3\}$ $R_j = FI(L_{j-1} \oplus KO_{i,j}, KI_{i,j}) \oplus$



Obr. 2.7 Pomocné funkcie blokovej šifry KASUMI [26]

R_{j-1} a $L_j = R_{j-1}$. Výstupom je 32bitový reťazec L_3R_3.

Poslednou podfunkciou je funkcia FI spracovávajúca 16bitový vstup rozdelený na 9bitovú časť L_0 a 7bitovú časť R_0 s použitím 16bitového kľúča $KI_{i,j}$ s časťoui $KI_{i,j,1}$ s dĺžkou 7 bitov a 9bitovou časťou $KI_{i,j,2}$. Celá funkcia je popísaná v následujúcich 4 krokoch:

1. $L_1 = R_0$, $R_1 = S9[L_0] \oplus ZE(R_0)$
2. $L_2 = R_1 \oplus KI_{i,j,2}$, $R_2 = S7[L_1] \oplus TR(R_1) \oplus KI_{i,j,1}$
3. $L_3 = R_2$, $R_3 = S9[L_2] \oplus ZE(R_2)$
4. $L_4 = S7[L_3] \oplus TR(R_3)$, $R_4 = R_3$

S7 a S9 sú S-boxy (mapy), ktoré konvertujú 7bitové vstupy na 7bitové výstupy (S7) alebo 9bitové vstupy na 9bitové výstupy (S9). Funkcia $ZE(input)$ vezme $input$ o dĺžke 7 bitov a pridá nulové bity na koniec k najvýznamnejšiemu bitu, a vytvorí tak 9bitový výstup. Naopak $TR(input)$ odstráni z 9bitového vstupu dva najvýznamnejšie bity a vráti výstup o 7 bitoch. [27]

2.3 4G

Prístupom na internet, ktorý zaistila 3G sieť, sa zvýšilo množstvo užívateľov využívajúcich vylepšených funkcií, ako aj služieb v reálnom čase. Avšak s príchodom internetu v mobilných sieťach prichádzajú aj nové nároky na sieť, ktorými sú lepšie pokrytie, širšie pásma, a s tým sa viaže predovšetkým vyššia rýchlosť prenosu. A tak vznikla 4G sieť, ktorá si vzala za cieľ splniť všetky tieto požiadavky. 4G priniesla okrem požadovaných nárokov ďalšie nové funkcie. Zatiaľ čo 3G rozdeľuje svoju architektúru podľa prepojovania paketov a okruhov, 4G sa stala prvou generáciou založenou iba na prepojovaní paketov a všetka komunikácia v sieti je riadená IP protokolmi. Ľuďom umožňuje pozerať videá na internete v HD kvalite, hrať hry s potrebným prístupom k internetu a vďaka IP protokolu tiež služby WWW alebo VoIP. [28], [29]

Siete štvrtej generácie boli operátormi uvádzané v praxi dávno predtým, než sa sieťmi 4G vôbec stali. Príkladom je 4G LTE. Hoci je jej rýchlosť asi 10krát nižšia ako štandardy 4G požadujú, čím je považovaná len za medzigeneráciu medzi 3G a 4G, mala veľké predpoklady k tomu, aby sa štandardom 4G stala. Neustály vývoj LTE nakoniec viedol k vzniku LTE-A, jedného zo štandardu 4G. [30], [31]

2.3.1 Architektúra LTE/LTE-A

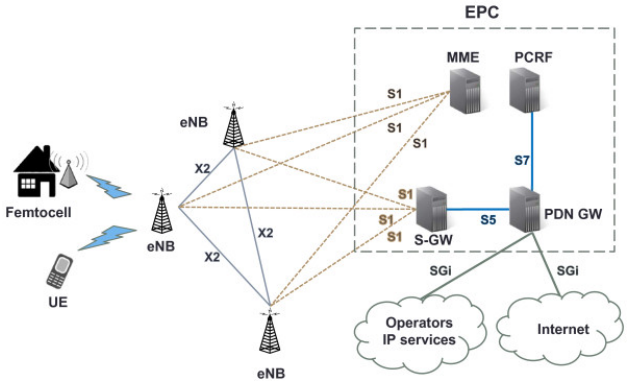
Aj keď sa zdá, že sa vývojom LTE až do príchodu LTE-A muselo veľa zmeniť, opak je pravdou. Teda aspoň navonok. Architektúra LTE siete zostala takmer nezmenená. Hlavný rozdiel medzi týmito verziami spôsobujú najmä nové technológie, ktoré LTE-A využíva. LTE-A, ako sieť patriaca do rodiny LTE sietí je tvorená z UE, E-UTRAN a EPC.

UE poskytuje rovnako prístup k sieti ako UE v 2.2.1 prostredníctvom kryptografických funkcií a parametrov v USIM.

E-UTRAN je tvorená narozdiel od jej predchodcov len prvkami jedného typu, ktorými sú eNodeB, alebo teda eNB. eNB sú vylepšené základňové stanice, ktoré sa okrem komunikácie medzi UE a EPC starajú posielaním signalizačných správ aj o nízkoúrovňové funkcie, akou je napríklad odovzdávanie kanálov medzi sebou.

EPC sa tiež tiež výrazne zjednodušilo. Jadro siete štvrtej generácie obsahuje len 5 prvkov, ktoré sa už navyše nedelia na komponenty pracujúce na prepojovaní okruhov a paketov, keďže celá sieť funguje vďaka IP protokolu a paketom. Hlavnými sú MME a S-GW (Serving Gateway) komunikujúce s E-UTRAN. MME patriace do riadiacej roviny architektúry sa stará o správu

mobilitu, sleduje užívateľovu polohu, aj počas jeho nečinnosti, udržiava spojenie s HSS, s úložiskom informácií o užívateľoch a riadi S-GW spolu s E-UTRAN počas predávania komunikačného kanálu pri zmene užívateľovej polohy. S-GW je riadiacim uzlom pre užívateľskú rovinu. Jeho prvoradou úlohou je preposielanie užívateľských paketov od užívateľa k P-GW (PDN – Packet Data Network – Gateway), ktorá posiela tieto pakety ďalej do iných externých sietí a priraďuje IP adresy. Poslednou je PCRF zodpovedné sa kontrolu a vynucovanie dodržiavania pravidiel siete a účtovania. [29], [32], [28], [33]



Obr. 2.8 Architektúra LTE [34]

2.3.2 Vrstvy v LTE/LTE-A

Vrstvy zabezpečujúce prenos dát v architektúre založenej na IP protokole sa delia na užívateľskú a riadiacu rovinu. V oboch rovinách sa nachádza PHY, MAC, RLC a PDCP. V užívateľskej rovine sa napokon pakety prenášajú IP protokolom. Riadiaca rovina kvôli komunikácii medzi uzlami siete využíva narozdiel od užívateľskej RRC a NAS.

PHY (Physical layer) je fyzická vrstva, ktorá zabezpečuje samotný prenos informácií. Spolu s tým sa stará o moduláciu signálu, synchronizáciu, kontroluje výkon a rôzne iné merania.

MAC (Medium Access Control) prideľuje podvrstve RLC logické kanály, riadi viacnásobný prístup multiplexovaním pre PHY, kontroluje a spravuje opravu chýb prostredníctvom HARQ (Hybrid Automatic Repeat Request) a priraďuje prioritu logickým kanálom užívateľa.

RLC (Radio Link Control) zaznamenáva poradie v doručovaní, kontroluje a spravuje opravu chýb prostredníctvom ARQ (Automatic Repeat Request), detekuje duplikáty a stará sa o segmentáciu pre potreby bloku transportnej vrstvy.

PDCP (Packet Data Convergence Protocol) stará sa o kompresiu hlavičky s ROHC (Robust Header Compression), detekciu duplikátov a taktiež je transportnou vrstvou pre dáta vrstvy RRC, koré tiež šifruje a zabezpečuje ich integritu.

RRC (Radio Resource Control) sprostredkováva komunikáciu medzi NAS a UE, riadi komunikáciu medzi UE a eNB, zodpovedá za funkcie kvality služieb QoS, mobilitu, bezpečnosti a správy bezpečnostných kľúčov, vytvára, udržuje a uvoľňuje RRC spojenie.

NAS (Non-Access Stratum) obsahuje protokoly pre spojenie a správu relácie medzi UE a MME. Táto vrstva je okrem toho zodpovedná za autentizáciu a ovládanie bezpečnosti. [31], [34], [35]

Tabuľka 2.2 Kľúče derivované z nadradeného kľúču KASME

	využitie kľúča
KNASenc	šifrovanie dát pri spojení NAS
KNASint	integrita dát pri spojení NAS
KeNB	derivácia kľúčov v eNB
KUPenc	šifrovanie dát na užívateľskej úrovni
KRRCenc	šifrovanie dát pri spojení RRC
KRRCint	integrita dát pri spojení RRC

2.3.3 Bezpečnosť v LTE/LTE-A

Špecifikácia 3GPP určuje 5 domén, v ktorých je možný zraniteľný vstup a musia byť v LTE zaistené:

Zabezpečenie prístupu k sieti Doména zabezpečujúca prístup k sieti sa zameriava hlavne na E-UTRAN zabezpečenie, kde je potrebné zaistiť bezpečné spojenie medzi UE a EPC. Rieši otázky anonymity užívateľa prostredníctvom dočasnej identity predplatiteľa, dôveryhodnosti siete a užívateľa na základe vzájomnej autentizácie a celkového šifrovania a integrity tohto spojenia.

Zabezpečenie domény siete Sieťová doména zaisťuje bezpečný chod predovšetkým vo vnútri siete, medzi HSS a obsluhujúcou sieťou MME, a medzi E-UTRAN a MME, kde dochádza k prenosu riadiacich a užívateľských dát.

Zabezpečenie užívateľskej domény Ako už napovedá názov, ide o zabezpečenie zariadenia užívateľa, a tiež prístupu k SIM karte tohto zariadenia. Po zapnutí zariadenia so SIM kartou je potrebné zadať štvormiestny číselný kód PIN.

Zabezpečenie aplikačnej domény Aplikačná doména sa týka jednotlivých aplikácií dostupných v sieti a tiež u užívateľa siete. Stará sa o prenos potrebných užívateľských dát a samotných dát aplikácie.

Viditeľnosť a konfigurovateľnosť zabezpečenia Zobrazuje užívateľovi konfiguráciu zabezpečenia, ktorú jeho operátor poskytuje a dostupné bezpečnostné prvky pre jeho zariadenia [36], [37], [38]

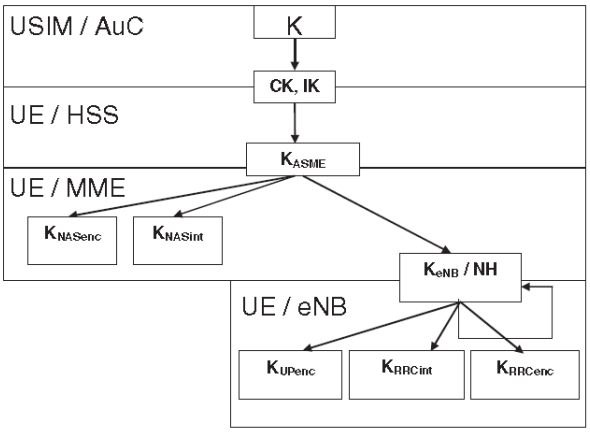
2.3.3.1 Autentizácia a tvorba kľúčov – algoritmus MILENAGE, kľúč KASME

Autentizácia a generovanie kľúčov prebieha podobným spôsobom ako v štandarde UMTS, avšak aby neprišlo k prenášaniu samotných kľúčov CK a IK, využíva sa tzv. kľúč KASME, ktorý je určený pre tvorbu kľúčov KNASenc, KNASint a KeNB, z ktorého sú derivované kľúče KUPenc, KRRCenc, KRRCint. Tento kľúč KASME je derivovaný z CK a IK a je prenášaný v autentizačnom vektore z AuC do MME. [36]

2.3.3.2 Šifrovanie a integrita dát – algoritmy EEA a EIA

Na šifrovanie a integritu dát v LTE sieťach sa využívajú funkcie označujúce sa ako EEA (EPS – Evolved Packet System – Encryption Algorithms) a EIA (EPS Integrity Algorithms). Konkrétne ide o algoritmy EEA1/EIA1 – SNOW 3G, EEA2/EIA2 – AES a EEA3/EIA3 – ZUC. [39]

Šifra **SNOW 3G** vyobrazená na obrázku 2.10 využíva jeden posuvný register so spätnou väzbou LFSR, ktorý obsahuje 16 32bitových prvkov $s_0, s_1 ... s_{15}$ a konečný stavový automat FSM. Posuvný register šifry sa incializuje pomocou 128bitového kľúča $k = k_0||k_1||k_2||k_3$ a 128bitového



Obr. 2.9 Hierarchia tvorby kľúčov so zdieľaným kľúčom K na vrchole [39]

inicializačného vektoru $IV = IV_0||IV_1||IV_2||IV_3$ spôsobom popísaným v tabuľke 2.3. Konečný stavový automat FSM pri inicializácii nastaví registre R_1, R_2 a R_3 na nulovú hodnotu. Po nastavení počiatočných hodnôt sa vykoná 32krát:

1. Najskôr prebehne takt konečného stavového automatu FSM, ktorý vyprodukuje pri jednom takte 32bitové slovo $F = (s_{15} \boxplus R_1) \oplus R_2$, operácia $\boxplus$ je definovaná ako celočíselný súčet modulo 2^{32}, a aktualizujú sa registre $(R_3 = S_2(R_2),\ R_2 = S_1(R_1),\ R_1 = R_2 \boxplus (R_3 \oplus s_5))$, pričom S_1 a S_2 sú substitučné boxy.
2. Dôjde k posunu registru LFSR, pri ktorom sa vypočíta nová hodnota na pozíciu $s_{15} = (s_{0,1}||s_{0,2}||s_{0,3}||0x00) \oplus MUL_\alpha(s_{0,0}) \oplus s_2 \oplus (0x00||s_{11,0}||s_{11,1}||s_{11,2}) \oplus DIV_\alpha(s_{11,3}) \oplus F$. Platí, že $s_0 = s_{0,0}||s_{0,1}||s_{0,2}||s_{0,3}$, rovnako $s_{11} = s_{11,0}||s_{11,1}||s_{11,2}||s_{11,3}$ a funkcie $MUL_\alpha()$ a $DIV_\alpha()$ sú mapovacie funkcie, ktoré zároveň rozširujú 8bitový vstup na 32bitový výstup.

Ďalšia fáza začína jedným taktom automatu FSM, pričom jeho výsledok sa zahodí. Potom sa vyprodukuje n 32bitových slov opakovaním nasledovných 3 krokov $n$krát:

1. FSM spracuje ďalší takt spôsobom ako v inicializačnej fáze, ktorého výstupom je slovo F.
2. Na toto slovo F spolu s prvkom s_0 sa aplikuje operácia xor a použije sa ako 32bitový výstup prúdovej šifry.
3. Prebehne takt registru LFSR, avšak tentokrát už nie s vstupnou hodnotou F, takže sa vypočíta nová hodnota na pozíciu s_{15} ako

$$(s_{0,1}||s_{0,2}||s_{0,3}||0x00) \oplus MUL_\alpha(s_{0,0}) \oplus s_2 \oplus (0x00||s_{11,0}||s_{11,1}||s_{11,2}) \oplus DIV_\alpha(s_{11,3}).$$

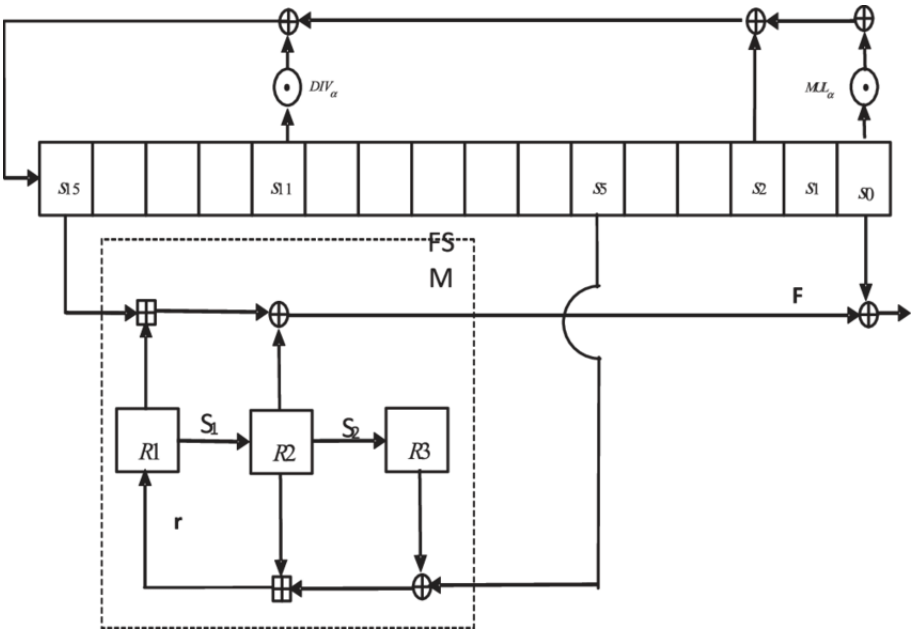
[40]

AES je blokovou šifrou, ktorej kľúč môže nadobúdať dĺžku 128, 192 alebo 256 bitov. Platí, že čím väčší kľúč je, tým je šifra bezpečnejšia, ale zároveň aj pomalšia a náročnejšia na výkon. V mobilných sieťach sa šifruje s pomocou 128bitového kľúča, preto ďalej v texte bude popisovaný tento variant. AES využívajúci 128bitový kľúč sa vykonáva v 10 rundách, pričom každá runda okrem poslednej pozostáva zo 4 podoperácií (SubBytes, ShiftRows, MixColumns a AddRoundkey). V poslednej runde sa nevykonáva operácia MixColumns.

Pred začiatkom šifrovania sa vytvorí 44 (počet 32bitových slov N_b*(počet rúnd $N_r + 1$)) 32bitových slov. Prvá štvorica slov w_0, w_1, w_2, w_3 je vytvorená kópiou šifrovacieho kľúča k. Ďalších

Tabuľka 2.3 Inicializácia posuvného registra šifry SNOW 3G ($\mathbb{1} = 0$xFFFFFFFF)

s_0	$k_0 \oplus \mathbb{1}$
s_1	$k_1 \oplus \mathbb{1}$
s_2	$k_2 \oplus \mathbb{1}$
s_3	$k_3 \oplus \mathbb{1}$
s_4	k_0
s_5	k_1
s_6	k_2
s_7	k_3
s_8	$k_0 \oplus \mathbb{1}$
s_9	$k_1 \oplus \mathbb{1} \oplus IV_3$
s_{10}	$k_2 \oplus \mathbb{1} \oplus IV_2$
s_{11}	$k_3 \oplus \mathbb{1}$
s_{12}	$k_0 \oplus IV_1$
s_{13}	k_1
s_{14}	k_2
s_{15}	$k_3 \oplus IV_0$



Obr. 2.10 Schéma algoritmu SNOW 3G [41]

Tabuľka 2.4 Konštanty R_{con} ($R_{con}[j] = 2 * R_{con}[j-1]$ s násobením nad $GF(2^8)$)

j	1	2	3	4	5	6	7	8	9	10
$R_{con}[j]$	01	02	04	08	10	20	40	80	1B	36

N_r štvorslov je vytvorených vždy pomocou predchádzajúceho štvorslova. Ak sa jedná o slovo w_i, ktorého index je deliteľný štyrmi, predchádzajúce slovo w_{i-1} sa orotuje o jeden bajt a aplikuje sa naň S-Box (substitúcia pomocou substitučného mapovania). Na tento výsledok sa aplikuje xor s konštantou z poľa R_{con} (viď 2.4) na indexe $i/4$.

Operácie v jednotlivých rundách aplikované na maticu 4 slov (uložených do stĺpcov) o veľkosti 4x4:

SubBytes každý bajt matice sa pomocou substitučnej tabuľky S-box namapuje na inú hodnotu bajtu.

ShiftRows prvý riadok matice nemení, ostatné riadky cyklicky posunie vždy o jeden bajt doľava.

MixColumns aplikuje násobenie nad Galoisovým telesom $GF(2^8)$ spolu s operáciou xor na vstupnú maticu a maticu $\mathrm{A} = \begin{pmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{pmatrix}$.

AddRoundKey pridá do šifry operáciou xor na vstup ďalší rundovný kľúč z poľa expandovaného kľúča. [42]

2.4 5G

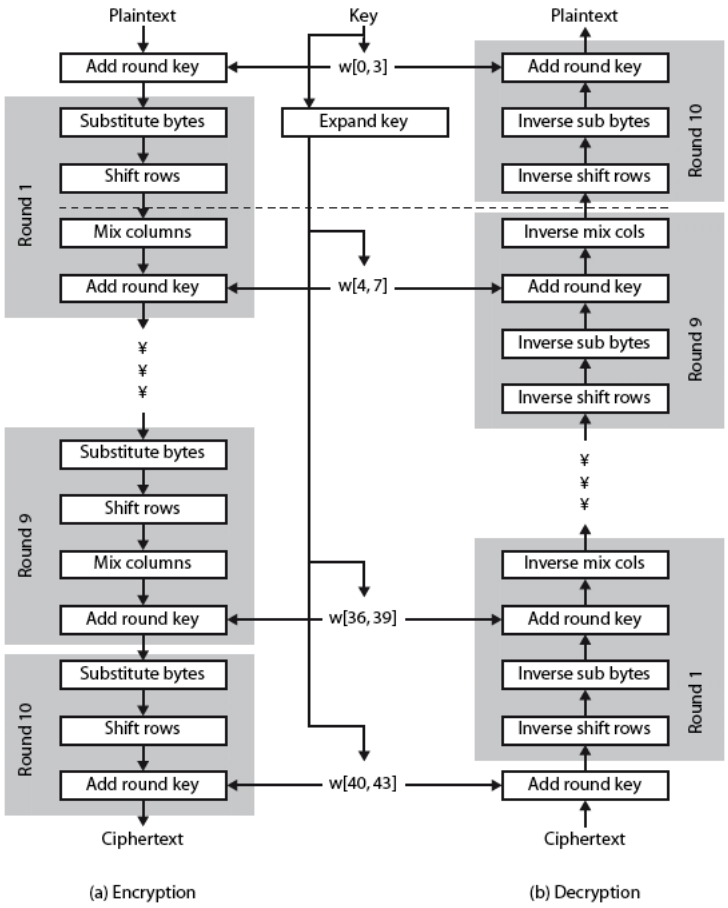
5G siete predstavujú novú generáciu, ktorej éra práve len pomaly začína. Už teraz však operátori sľubujú nespočetné množstvo výhod a prínosov tejto generácie. Tieto výhody môžu nastať len vďaka tomu, že už pri samotnom plánovaní 5G sietí sa myslelo na enormné množstvo zariadení a ich rozličných využití, pre ktoré bude využívaná. Technológie, ktoré sú do návrhov týchto sietí zahrnuté zvyšujú rýchlosť prenosu, znižujú latenciu a podporujú kvalitnejšie zabezpečenie siete.

Pri 5G sieťach sa často hovorí o 5G NSA (Non-Standalone) a 5G SA (Standalone) sieťach. V prvom prípade ide o siete, ktoré zdieľajú komponenty s 4G sieťou, väčšinou ide o EPC jadro siete, ktoré riadi nielen 4G sieť, ale aj komponenty 5G siete. Tento variant šetrí operátorom počiatočné náklady pri nasadení siete, ale nie je možné využiť všetky výhody, ktoré 5G sieť ponúka. Najmä čo sa týka latencie. V druhom prípade ide o 5G sieť, ktorá je schopná fungovať samostatne, a tak poskytnút všetky prínosy, ktoré siete 5G ponúkajú.

Siete piatej generácie využívajú frekvenčné pásma pod 1 GHz, ktoré slúžia najmä pre pokrytie veľkého menej osídleného územia, od 1GHz po 6 GHz poskytujúce väčšiu šírku pásma a stále dostatočne veľké pokrytie, často sú využívané pre tzv. kritickú komunikáciu (vyžadujú veľmi nízku latenciu a rýchly prenos). Ďalej im boli priradené pásma frekvenčného spektra nad 24 GHz, ktoré sa nazývajú aj rozšírené frekvenčné pásma, kde už sa hovorí o milimetrových vlnách. Milimetrové vlny dosahujú malých vzdialeností, avšak poskytujú veľkú šírku pásma, takže aj rýchly prenos dát. Väčšinou ide o pásma, ktoré sú priradené ako doplnkové v prípade vyššieho objemu vysielaných dát. [43]

2.4.1 Jadro 5G sietí

Jadro siete 5G, označované tiež ako NG-Core (New Generation Core), nemá špecifikáciou dané presné presné komponenty. NG-Core je založené na princípe microservice-based architecture



Obr. 2.11 Schéma šifry AES s 10 rundami a podoperáciami SubBytes, ShiftRows, MixColumns a AddRoundkey [42]

(architektúry založenej na mikroslužbách). To znamená, že ide o architektúru založenú na čo najmenších možných funkčných blokoch. Špecifikácia definuje len množinu funkcií, ktoré by mala sieť spĺňať, avšak je už na operátoroch, ako jadro siete navrhnú a rozčlenia do komponent. Jednou z možných architektúr, ktorá takto člení funkcie do jednotlivých komponent, je architektúra jadra obsahujúca komponenty nižšie. Jej schéma je vyobrazená na obrázku 2.12.

AMF (Access and Mobility Management Function) – funkcia správy prístupu a moility – zodpovedá za správu mobility zariadení, prístup k sieťovým funkciám a jeho zabezpečenie prostredníctvom autentizácia a autorizácie.

SMF (Session Management Function) – funkcia správy relácie – obstaráva reláciu užívateľa, prideľuje IP adresy a kontroluje kvalitu služieb.

AUSF (Authentication Server Function) – funkcia autentizačného / overovacieho serveru – stará sa o autentizáciu užívateľa a vykonáva kryptografické funkcie.

NSSF (Network Slicing Selector Function) – funkcia výberu sieťového segmentu – vyberá segment siete určený danému zariadeniu s ohľadom na účel využitia. Priradený segment siete tiež určuje, aká časť AMF je sprístupnená.

PCF (Policy Control Function) – funkcia kontroly politík – spravuje kontrolu práv a povinností v sieti, kontroluje prístup k funkciám zariadenia a vynucuje dodržiavanie politík u ostatných funkcií.

NRF (Network Function Repository Function) – funkcia úložiska sieťových funkcií – vyhľadáva dostupné funkcie siete v úložisku funkcií.

NEF (Network Exposure Function) – funkcia sieťovej expozície – poskytuje dostupné funkcie a služby siete bezpečným spôsobom. Tieto funkcie môžu byť dostupné aj pre softvéry tretích strán.

UDM (Unified Data Management) – jednotná správa údajov – spravuje informácie o užívateľoch a ich zariadeniach, jednotlivých funkciách, ktoré sú im dostupné a uchováva údaje potrebné k autentizácii.

SDSF (Structured Data Storage Function) – funkcia úložiska štruktúrovaných dát – úložisko s definovanou štruktúrou, napr. prostredníctvom databázy.

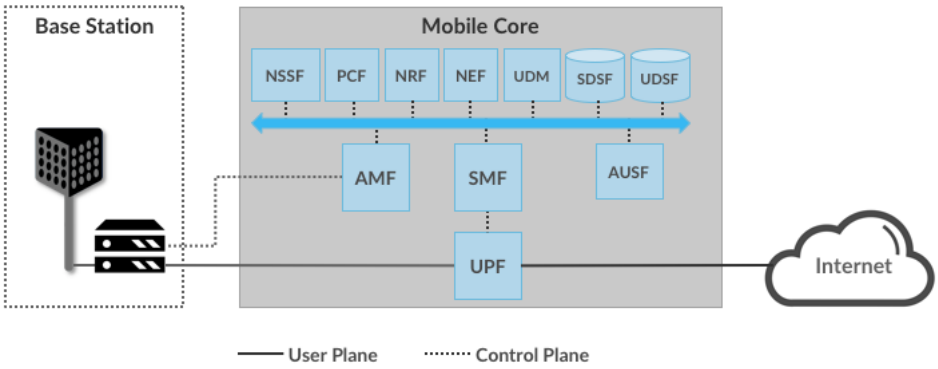
UDSF (Unstructured Data Storage Function) – funkcia úložiska neštruktrúrovaných dát – ukladá funkcie len pomocou kľúča a hodnoty.

UPF (User Plane Function) – funkcie užívateľskej roviny – preposiela dáta užívateľa medzi zariadením a externou sieťou. Zároveň sleduje prevádzku, presadzuje pravidlá siete a dodržiavanie služieb predplatiteľa. [43]

2.4.2 Technológie podporujúce bezpečnosť 5G sietí

Návrh 5G siete je z veľkej časti na samotných operátoroch, pretože nie je definovaná jej implementácia alebo algoritmy, ktoré by musela sieť využívať. Avšak je odporúčané držať sa určitých stavebných blokov, ktoré pomáhajaú dosiahnuť kladených podmienok na sieť.

- Technológia native cloud
- Architektúra založená na mikroslužbách
- Virtualizácia sieťových funkcií NFV (Network Function Virtualisation)



Obr. 2.12 Komponenty jadra siete 5G rozdelené na komunikáciu riadiacej roviny (Control Plane) a užívateľskej roviny (User Plane) [43]

- Softvérovo definovaná sieť SDN (Software Defined Network)
- Decentralizovaný cloud (pomocou edge computing)
- Delenie siete na segmenty (pomocou network slicing)

Technológia native cloud pristupuje k návrhu siete tak, aby sa na zavedennie cloudu myslelo už od samého začiatku. Preto je 5G architektúra postavená na čo najmenších funkčných blokoch definujúcich architektúru založenú na mikroslužbách, ktorá je pre native cloud častá. Vďaka mikroslužbám je možná prípadná rýchla, nezávislá a jednoduchá oprava alebo rozšírenie. Je tiež známe, že čím menšie úlohy funkcie vykonávajú, tým konkrétnejšie je možné určiť prístup k nim, a tak nastaviť práva iba na tie potrebné a ostatné zakázať. Takto prispieva tiež k zabezpečeniu funkcií siete.

Okrem mikroslužieb využíva native cloud aj myšlienok NFV a SDN, ktoré sa často pri výklade často zamieňajú, avšak ich použitie je odlišné. NFV využíva virtualizované sieťové funkcie umožňujúce nad hardvérovými prvkami vykonať softvérovú nadstavbu upravujúcu funkčnosť týchto hardvérových prvkov. Vrtualizácia siete tak umožní spojenie alebo oddelenie funkčnosti jednotlivého hardvéru. Vďaka nej je možné nahradiť drahšie hardvérové zariadenia lacnejším variantom v podobe softvéru, uľahčiť tak rozširiteľnosť a oddeliť od seba hardvérovú a softvérovú zložku. Cieľom SDN je oddeliť riadiacu a dátovú časť a dosiahnuť riadenia celej siete za pomoci jedného kontroléra. Tu je však potrebné, aby obe časti boli dostatočne zabezpečené ďalším bezpečnostnými opatreniami, pretože zlúčenie riadiacej a dátovej časti je bezpečnostné riziko.

Decentralizovaný cloud, častejšie označovaný výrazom edge computing, je formou cloudu, ktorý je ale posunutý „bližšie“ k zariadeniam. Čo znamená, že sieť nie je spracovávaná iba jedným centralizovaným cloudom niekde vnútri jadra siete, ale menšie cloudy sú umiestnené napríklad na základňové stanice. Takto je rozdelená záťaž medzi jednotlivé cloudové uzly, ktoré rozhodujú tiež o priorite a dôležitosti dát a ich preposielaní do centrálneho cloudu, takže neprichádza k jeho zahlteniu.

Delenie siete na segmenty využíva funkciu NSSF, ktorá bola spomenutá v 2.4.1. Sieť rozdelená na segmenty určené hlavným účelom využitia, môžu zvýšiť nielen efektivitu, ale aj zabezpečenie. Takto je možné rozdeliť sieť napríklad na sieť pre smartfóny užívateľov, ktoré využívajú väčšiu šírku pásma, IoT zariadenia používajúce menšiu batériu (je potrebná nižšia spotreba) a autonómne vozidlá, ktoré vyžadujú vysokú spoľahlivosť a nulovú latenciu. Tým bude umožnený prístup zariadeniam iba do takej časti siete, ku ktorej majú prístupové práva.

Kapitola 3
srsRAN

Projekt srsRAN (do roku 2021 známy ako srsLTE) poskytuje softvér, ktorý je schopný spolupracovať s rádiovým prístupom, vďaka čomu je pomocou neho možné vytvoriť si vlastnú mobilnú sieť so všetkými jej prvkami. srsRAN umožňuje vytvoriť sieť štvrtej alebo piatej generácie. K funkčnosti siete poskytuje aplikácie srsUE (4G a 5G NSA / SA UE) a srsENB (4G eNB a 5G NSA / SA gNB). Pre 4G sieť ponúka tiež jadro sieť srsEPC s prvkami MME, HSS, S-GW a P-GW. Pri spustení siete 5G je potrebné využiť softvér tretej strany posyktujúci jadro siete.

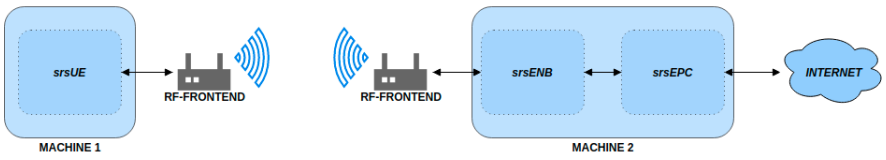
Keďže praktická časť tejto práce sa zaoberá najmä sieťami 4G, ďalší text sa bude zameriavať podobne predovšetkým na 4G sieť.

3.1 Nasadenie 4G siete pomocou srsRAN

Pre nasadenie celej 4G siete LTE pomocou srsRAN je potrebné splniť požiadavky definované projektom srsRAN. Aby mohla byť simulovaná celá sieť takmer v reálnych podmienkach, srsRAN vyžaduje dva počítače s operačným systémom Linux / GNU a dve RF front-end (Radio Frequency front-end) zariadenia. Tieto zariadenia podporujú vysielanie a prijímanie rádiového signálu z a do zariadenia, ku ktorému sú pripojené. Sieť tak s týmto vybavením vyzerá nasledovným spôsobom. Na jednom zariadení s RF front-end bežia v dvoch odlišných konzolách aplikácie srsENB a srsEPC, a srsEPC sa pripojí do internetu. Druhé zariadenie simuluje užívateľské zariadenie prostredníctvom srsUE a druhého RF front-end zariadenia. [44]

Pre správne fungovanie srsRAN uvádza možné ovládače RF front-end, ktoré sú s podporou srsRAN.

- UHD
- SoapySDR
- BladeRF
- ZeroMQ



Obr. 3.1 Architektúra siete nasadená pomocou srsRAN [44]

Pre štandardnú aplikáciu srsRAN s USRP front-end je k dispozícii inštalácia balíčku z príkazového riadku pomocou príkazov 3.1.

Výpis kódu 3.1 Inštalácia sady srsRAN štandardnej aplikácie s USRP front-end

```
sudo add-apt-repository ppa:softwareradiosystems/srsran
sudo apt-get update
sudo apt-get install srsran -y
```

Pre modifikáciu aplikácie alebo použitie iného zariadenia RF front-end je potrebná inštalácia zo zdroja. Nainštalovanie požadovaných knižníc príkazom 3.2, stiahnutie srsRAN z GIT repozitáru 3.3 a následná inštalácia s dostupnými konfiguračnými súbormi 3.4.

Výpis kódu 3.2 Inštalácia potrebných knižníc pre spustenie srsRAN

```
sudo apt-get install build-essential cmake libfftw3-dev libmbedtls-
    ↪ dev libboost-program-options-dev libconfig++-dev libsctp-dev
```

Výpis kódu 3.3 Stiahnutie a kompilácia softvéru srsRAN zo zdroja

```
git clone https://github.com/srsRAN/srsRAN.git
cd srsRAN
mkdir build
cd build
cmake ../
make
make test
```

Výpis kódu 3.4 Inštalácia softvéru srsRAN zo zdroja

```
sudo make install
srsran_install_configs.sh user
```

3.1.1 srsUE

srsUE je softvér, ktorý je schopný pri spustení na počítači s operačným systémom Linux / GNU pripojiť sa do LTE siete a simulovať tak rozhranie mobilnej siete. Na prijímanie a vysielanie rádiových vĺn vyžaduje použitie softvérom definovaného rádia SDR (Software Defined Radio), akým je napríklad Ettus Research USRP.

Aplikácia, ako je zobrazené na obrázku 3.2a, podporuje vrstvy, ktoré boli definované v 2.3.2.

1. L1

   PHY zaisťuje prenos dát z vrstvy MAC, vyhľadávanie buniek, kontroluje a spravuje výkon.

2. L2

   MAC zabezpečuje multiplex dát, ktoré sú prenášané z alebo do PHY vrstvy. Zodpovedá za výmenu informácií s eNB, znovuposielanie a opravu chýb prostredníctvom HARQ a plánovanie prioritného spracovania.

   RLC má na starosti viacero logických kanálov, z ktorých každý môže pracovať v jednom z 3 režimov – transparentný (Transparent Mode) UM, nepotrvdený (Unacknowledged Mode) UM a potvrdený (Acknowledged Mode) AM. TM prenáša jednoducho dáta cez RLC. UM segmentuje a znovu spája segmentované dáta pre transportnú vrstvu, ak je to potrebné, zodpovedá za detekciu duplikátov a preusporiadanie. AM vykonáva rovnaké funkcie ako UM, ale navyše opakuje prenos chýbajúcich dát a resegmentuje sieť.

PDCP je transportnou vrstvou pre dáta z vyššej vrstvy, je zodpovedný za zabezpečenie týchto dát prostredníctvom šifrovania a kontroly integrity. Vyraďuje duplikáty a môže komprimovať hlavičky ROHC.

3. L3

RRC vytvára, udržuje a uvoľnuje RRC spojenie s eNB, cez ktoré vykonáva správu kľúčov, výmenu informácií medzi UE a eNB. Spravuje konfiguráciu nižších vrstiev UE, k čomu využíva sieťou vysielané informácie systému (SIB – System Information broadcast). Riadi výber buniek, mobilitu aj pre odovzdávanie medzi bunkami, ktoré spolu susedia, zabezpečenie spojenia a konfiguráciu rádiových nosičov.

NAS vyberá PLMN, riadi pripojenie k sieti a udržiava spojenie medzi UE a EPC, cez ktoré spravuje výmenu dát riadiacej roviny prostredníctvom signálov.

GW (Gateway) vytvára a udržuje virtuálne sieťové jadro TUN. Vrstva je schopná spustenia aplikácie u užívateľa a pracovať tak s IP paketmi dátovej roviny.

Softvér srsUE je možné nahradiť iným zariadením, ktoré je schopné prijímať a vysielať rádiové signály. Vyžaduje sa však použiť navyše programovateľnú SIM kartu, ktorá umožní pripojenie zariadenia do siete. [44]

3.1.1.1 Konfigurácia srsUE

Nastavenie UE je možné pomocou konfiguračného súboru **ue.conf**. Prednastavené UE využíva virtuálnu USIM kartu s parametrami práve z tohto súboru. Týmito parametrami sú algoritmus ALGO, unikátny identifikátor IMSI, tajný kľúč K a kód operátora OP alebo OPC. K úspešnému pripojeniu je potrebné, aby sa tieto hodnoty zhodovali s hodnotami uložený v HSS databáze vo vnútri jadra EPC. Po zmene parametrov, je pre spustenie s použitím konfiguračného súboru príkaz 3.5.

Výpis kódu 3.5 Spustenie srsUE za použitia konfiguračného súboru ue.conf

```
sudo srsue ~/.config/srsran/ue.conf
```

3.1.1.2 Spustenie srsUE

Spustenie srsUE z príkazového riadku vyžaduje administrátorské práva (3.6) kvôli vláknu s vysokou prioritou a tiež k vytvoreniu jadra TUN. Po spustení sa pokúsi vyhľadať zariadenie RF front-end a pripojiť sa k lokálnej bunke. Po úspešnom pripojení sa spustí TUN rozhranie tun_srsue, ktoré umožňuje posielanie dát z a do siete.

Výpis kódu 3.6 Spustenie srsUE

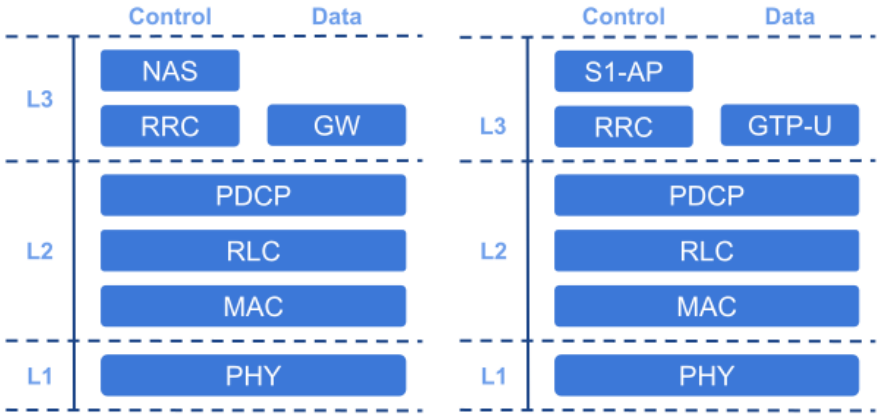
```
sudo srsue
```

Výsledky komunikácie so sieťou sú dostupné v logovacom súbore, ktorý sa vytvára v **/tmp/ue.log**.

3.1.2 srsENB

Je softvérovo vytvorená základňová stanica eNB, ktorá vytvára lokálnu bunku, pomocou ktorej sa pripája zariadenie do LTE siete. K vysielaniu a prijímaniu signálu je potrebné RF front-end zariadenie ako v 3.1.1.

Softvér podporuje rovnaké funkcie vrstiev L1, L2 a RRC v L3 z 3.2a, v tretej vrstve L3 sa líšia najvyššie vrstvy riadiacej (S1-AP – S1 Application Protocol) a dátovej (GTP-U – GPRS Tunnelling Protocol User Plane) roviny.



(a) srsUE **(b)** srsENB

Obr. 3.2 Vrstvy v architektúre softvéru srsRAN [44]

S1-AP poskytuje spojenie v riadiacej rovine medzi eNB a jadrom siete EPC, kde sa pripojí k MME uzlu. Preposiela správy z MME do UE najskôr do vrstvy RRC, kde sa zapuzdria a uložia do zásobníka na prenos. V prípade správ z UE do MME sú zapuzdrené tiež vrstvou RRC, ale pred prijatím uzlom MME sa extrahujú v eNB. Až po extrahovaní ich prevezme S1-AP a prepošle ďalej do siete.

GTP-U sa pripája k uzlu S-GW v EPC. IP pakety dátovej roviny sú zapuzdrené do paketov GTP a vysielané cez vytvorený tunel do siete. Tu sa extrahujú z tunelu a pomocou P-GW sa prepošlú ďalej do internetu. [44]

3.1.2.1 Konfigurácia srsENB

Na konfiguráciu srsENB slúži súbor **enb.conf**. Umožňuje nastaviť parametre ku konfigurácii bunky, frekvenciam vysielania, úrovniam prenosového výkonu a logovania.

Konfigurácia srsENB umožňuje samostatne nastaviť vysielané systémové informácie SIB prostredníctvom súboru **sib.conf**, rádiové zdroje cez **rr.conf** a dátové nosiče **drb.conf**.

Hlavným parametrom je **enb.mme_addr** – IP adresa pridelená uzlu MME. Predvolené nasadenie predpokladá, že srsEPC a srsENB bežia na rovnakom počítači (3.1).

Výpis kódu 3.7 Spustenie srsENB za použitia konfiguračného súboru enb.conf

```
sudo srsenb ~/.config/srsran/enb.conf
```

3.1.2.2 Spustenie srsENB

Spustenie srsENB z príkazového riadku vyžaduje administrátorské práva (3.8) kvôli vláknam s vysokou prioritou. Po spustení sa pokúsi vyhľadať zariadenie RF front-end a pripojiť sa k jadru EPC. Po úspešnom pripojení začne vysielať.

Výpis kódu 3.8 Spustenie srsENB

```
sudo srsenb
```

Výsledky sú dostupné v logovacom súbore, ktorý sa vytvára v **/tmp/enb.log**.

3.1.3 srsEPC

srsEPC je softvérom simulujúcim jadro siete LTE – EPC. Poskytuje dôležité komponenty siete, ktoré sú vyobrazené na obrázku 3.3 a ich funkcie sú popísané v 2.3.1.

Funkcie, ktoré priamo poskytuje implementácia srsRAN sú popísané nižšie.

MME podporuje protokoly NAS a S1-AP vrstiev na dáta posielané v riadiacej rovine medzi jadrom EPC, základňovou stanicou eNB a zariadením UE. Vo vrstve NAS sú zahrnuté funkcie ako procedúra na pripojenie, odpojenie a žiadosť o službu, požiadanie o identitu alebo odpoveď na ňu a autentizácia. S1-AP vrstva vytvára alebo uvoľňuje spojenie s MME a preposiela správy vrstvy NAS.

HSS umožňuje predovšetkým konfiguráciu parametrov potrebných pre autentizáciu UE. Pracuje s databázou vytvorenou vo formáte súboru CSV, podporuje autentizačné algoritmy XOR a MILENAGE, informuje o triede kvalít služieb QCI a konfiguruje dynamickú alebo statickú IP alokáciu.

SPGW preposiela komunikáciu užívateľskej dátovej roviny medzi EPC a eNB. Zabezpečuje rozhranie virtuálnej siete pomocou jadra TUN, preposielanie medzi eNB, S-GW a ďalej cez P-GW do internetu pomocou protokolu GTP-U. Vytvára alebo uvoľňuje GTP-U tunely, cez ktoré sú dáta posielané.

3.1.3.1 Konfigurácia srsEPC

Na konfiguráciu srsEPC slúžia súbory **epc.conf** a **user_db.csv**. **epc.conf** sa týka uzlov siete, kde sa určuje PLMN siete, algoritmy pre šifrovanie a integritu, APN (Access Point Name), IP adresu, adresu protoku GTP-U a iné.

Druhý súbor je databázou pre uzol HSS v jadre. Ide o súbor ktorý udržiava informácie o užívateľoch siete. Každý riadok súboru je záznam jedného užívateľa a hodnoty týkajúce sa parametrov tohto užívateľa sú oddelené čiarkou (CSV formát). Hodnoty sa musia zhodovať s hodnotami uloženými na USIM karte UE zariadenia.

Štruktúra súboru user_db.csv je

$$(ue\_name), (algo), (imsi), (K), (OP/OPc\_type), (OP/OPc\_value), (AMF), (SQN), (QCI), (IP\_alloc)$$

.

ue_name meno zariadenia

algo autentizačný algoritmus (MILENAGE alebo XOR)

imsi medzinárodný identifikátor mobilného predplatiteľa

K zdieľaný tajný kľúč

OP/OPc_type nemodifikovaný typ OP alebo zašifrovaný typ OPC

OP/OPc_value konkrétna hodnota kódu operátora odvíjajúca sa od predchádzajúceho parametra

AMF pole pre správu autentizácie

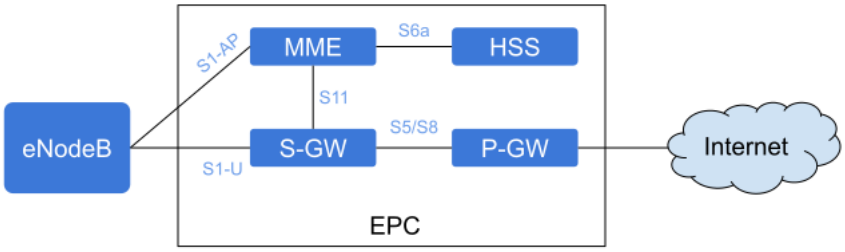
SQN sekvenčné číslo

QCI identifikátor triedy kvalít služieb

IP_alloc dynamická alebo statická alokácia

3.1.3.2 Spustenie srsEPC

Spustenie srsEPC z príkazového riadku vyžaduje administrátorské práva (3.9), aby sa vytvorilo rozhranie TUN, ktoré spustí EPC a bude čakať na pripojenie eNB a zariadení. srsEPC spustí tiež rozhranie TUN nazývané **srs_spgw_sgi**, ktoré zabezpečuje posielanie užívateľských paketov do užívateľského zariadenia.



Obr. 3.3 Komponenty architektúry EPC jadra softvéru srsRAN [44]

Výpis kódu 3.9 Spustenie srsEPC

```
sudo srsepc
```

Pri spustení EPC a eNB na odlišných zariadeniach, musia byť nastavené adresy **mme_bind_addr** a **gtpu_bind_addr**. Prvá hodnota určuje adresu, na ktorej bude spojenie S1-AP medzi eNB a MME, druhá určuje adresu spojenia GTP-U. Tieto hodnoty by mali byť rovnaké, ak užívateľ srsRAN nevyžaduje, aby užívateľká rovina bola v inej podsieti (príklad spustenia s IP adresou 10.0.1.10 v 3.10).

Výpis kódu 3.10 Spustenie srsEPC a srsENB na odlišných zariadeniach

```
sudo srsepc --mme.mme_bind_addr 10.0.1.10 --spgw.gtpu_bind_addr
    ↪ 10.0.1.10
```

Aby bolo možné sieť pripojiť k internetu, je potrebné vykonať IP masquerading. Bez nej by jadro operačného systému Linux nebolo schopné preposielať pakety z jednej podsiete do druhej. Zapnúť IP masquerading je možné pomocou skriptu 3.11, v ktorom sa definuje vonkajšie rozhranie do internetu.

Výpis kódu 3.11 Skript pre IP masquerading

```
sudo srsepc_if_masq <out_interface>
```

Log spustenej siete je možné sledovať v súbore, **/tmp/epc.log**.

Obr. 3.4 Mobilné zariadenia komunikujúce so zariadením využívajúcim USRP [45]

Kapitola 4
Praktická časť

Nástroj pre demonštráciu šifrovacieho algoritmu a generovanie hodnôt kryptografických dát SIM kariet bol rozdelený na dva samostatné nástroje, aby boli tieto nástroje schopné fungovať nezávisle od seba.

4.1 Výber programovacieho jazyka a knižníc

4.1.1 Programovací jazyk Python

Nástroje boli implementované v programovacom jazyku Python, konkrétne Python 3. Práca sa zaoberá kryptografickými hodnotami, ktoré sú väčšieho rozsahu. Väčšinou ide o 128bitové čísla, nad ktorými je potrebné vykonávať operácie. Výhodou tohto programovacieho jazyka je, že od Python 3 je už v základnom datovom type *int* zahrnnutý typ, ktorý umožňuje ukladať do premennej veľké čísla. Preto nebolo potrebné ani importovať ďalšiu knižnicu na spracovanie takýchto čísel.

4.1.1.1 csv

Implementuje triedy, ktoré sú určené k čítaniu a zapisovaniu do súborov formátu csv – súbor s hodnotami oddelenými čiarkou. Ide o najčastejší najjednoduchší formát zápisu hodnôt týkajúci sa tabuliek a databáz. Vďaka modulu csv, je možné definovať podporované oddeľovače a pristupovať k riadku súboru ako k poľu hodnôt. [46]

4.1.1.2 secrets

Mobul secrets umožňuje generovať kryptograficky silné náhodné čísla. Je vhodný pre generovanie v oblasti bezpečnosti a kryptografie narozdiel od pseudonáhodného generátora čísel *random*. V práci je využitý na generovanie hexadecimálneho reťazca, ktorý slúži ako kľúč. [47]

4.1.1.3 Crypto.Cipher.AES

Implementácia blokovej šifry AES od PyCryptodome bola využitá na šifrovanie kódu operátora a na šifrovanie blokov pri algoritme MILENAGE. V práci je šifrovanie pomocou 128bitového kľúča, avšak modul podporuje tiež šifrovanie pomocou 192bitových a 256bitových kľúčov. Umožňuje tiež výber blokového módu šifrovania. [48]

4.1.1.4 click

Click pomáha vytvárať rohranie pre príkazový riadok. Click výrazne uľahčuje spracovanie príkazového riadku, aj s podporou prepínačou. Za použitia balíku click je možné definovať typ, ktorý má byť prijatý,

a tak ošetriť vstupy. Zároveň pri jeho použití, sa automaticky vygeneruje krátky manuál, ktorý definuje dostupné funkcie a prepínače. [49]

4.2 Generovanie hodnôt kryptografických dát SIM kariet

Na implementovaný nástroj bola kladená podmienka využitia v testovacej 4G sieti vytvorenej pomocou projektu srsRAN. Tento projekt, ako bolo spomenuté v 3.1.3.1, využíva súbor user_db.csv ako databázu užívateľov, kde jeden riadok súboru patrí záznamu jedného užívateľa. Preto bol nástroj navrhnutý ako skript určený na generovanie súboru user_db.csv.

Skript sim.py, ktorý poskytuje generovanie súboru vo formáte súboru user_db.csv, implementuje dve funkcie *empty* a *add*. Obe funkcie prijímajú ako vstupný parameter názov generovaného súboru. Príkaz *empty* vytvára nový súbor s prázdnou databázou obsahujúci len popis parametrov, zatiaľ čo *add* umožňuje pridanie jedného užívateľa s predvolenými a generovanými hodnotami alebo pomocou prepínačou s hodnotami od užívateľa.

- Záznam užívateľa začína menom jeho zariadenia, v prípade generovaného súboru, ide buď o meno zariadenia zadané užívateľom alebo reťazec **ue** s číslom, ktoré následuje po maximálnom čísle už v existujúcej databáze.
- Ďalej následuje autentizačný algoritmus, kde sa zadáva *mil* pre algoritmus MILENAGE alebo *xor* pre XOR. Predvolený algoritmus je MILENAGE, avšak je možné túto voľbu zmeniť pomocou prepínaču.
- Hodnotu IMSI, ktorá následuje, je väčšinou možné získať z programovateľných kariet (napr sysmoUSIM od sysmocom), avšak pre testovacie účely je možné hodnoty generovať na základe už použitých hodnôt IMSI v existujúcom súbore.
- Tajný kľúč sa generuje pomocou modulu secrets ako hexadecimálny reťazec, pomocou prepínaču je možné zadať požadovanú hodnotu.
- Typ kódu operátora sa priraďuje predvolene „opc“, kedže jeho použitie je bezpečnejšie. Prepínač - *-insecure* vygeneruje záznam s „op“.
- Hodnota kódu operátora je u každého záznamu nastaviteľná prepínačom pri zadávaní z príkazového riadku. Bez jej nastavenia sa priraďuje hodnota uvedená v skripte, ako predvolená.
- Pole AMF generuje predvolenú hodnotu 0x9000, ktorá bola zvolená na základe analýzy ukážkových súborov user_db.csv. Podobne je hodnotu možné zmeniť, avšak najbežnejšie hodnoty sú 0x8000 alebo 0x9000. Uvádza sa, že hodnoty AMF by sa mali pohybovať nad hodnotou 0x8000
- Sekvenčné číslo SQN má predvolenú hodnotu nastavenú na 000000000001, pričom uvádzajú sa aj hodnoty 000000000000.
- QCI, ktoré identifikuje triedu kvality služieb, uvádza väčšinou hodnoty 7 alebo novšie aj 9.
- Alokácia IP adresy, je možná pomocou statickej adresy, ktorú skript umožňuje, no jej použitie sa takmer nikde neuvádza, preto sa bez nastavenia generuje reťazec *dynamic*.

4.2.1 Generovanie mena užívateľa a IMSI

Ak nie je zadané meno užívateľa alebo IMSI, je potrebné tieto hodnoty generovať. Implementovaný skript sim.py generuje tieto hodnoty jednoduchou formou pomocou dvoch funkcií.

1. Funkcia *__getInfo__* otvorí v režime čítania súbor, do ktorého má byť nový užívateľ zapísaný, odfiltruje riadky, ktoré popisujú štruktúru súboru a nastaví csv reader. Ďalej prejde tento súbor po jednotlivých riadkoch hodnoty mena zariadenia a IMSI, a vráti maximálnu hodnotu (u mena užívateľa vyberie len tie, ktoré začínajú reťazcom **ue**, vyberie z nich číslo, ktoré tento reťazec následuje, a z týchto hodnôt vráti maximum). 4.1
2. Vo funkcii *add* sa overí, že hodnoty mena užívateľa a IMSI nie sú nastavené a priradia sa maximálne hodnoty z funkcie *__getInfo__*, ku ktorým sa pričíta 1.

Výpis kódu 4.1 Získanie informácií už z existujúceho súboru pomocou funkcie *__getInfo__*

```
    def __getInfo__(filename):
        with open(filename, 'r') as file:
            file = filter(lambda ln: ln[0] != '#', file)
            reader = csv.reader(file)
            max_imsi = 0
            max_ue = 0

            for line in reader:
                max_imsi = max(max_imsi, int(line[2]))
                if re.fullmatch("ue[0-9]+", line[0]):
                    max_ue = max(max_ue, int(line[0][2:]))

            return max_ue, max_imsi
```

Výpis kódu 4.2 Nastavenie mena užívateľa a IMSI vo funkcii *add*

```
        if name is None or imsi is None:
            max_ue, max_imsi = __getInfo__(filename)
        else:
            max_ue, max_imsi = 0, 0

        if name is None:
            name = f"ue{max_ue + 1}"

        if imsi is None:
            imsi = max_imsi + 1
```

4.3 Spracovanie algoritmov MILENAGE a XOR

Aj napriek už existujúcim implementáciám algoritmu MILENAGE, ktoré sú dostupné aj ako online kalkulačky, tieto implementácie nespĺňali požiadavky umožňujúce demonštráciu pre učebné alebo ladiace (debug) účely.

Skript mil.py môže slúžiť ako už existujúce dostupné riešenia, kedy pri zadaní parametrov vráti len potrebné parametre. Navyše však umožňuje aj režim, pri ktorom vypisuje kroky po jednotlivých operáciách. Okrem toho, zadávanie parametrov je možné z príkazového riadku alebo aj z dostupného súboru spĺňajúceho formát súboru user_db.csv.

Príkaz *compute* pracuje s parametrami zadanými na príkazovom riadku, kde je potrebné zadať hodnoty SQN, AMF, OP a KEY. Umožňuje zmeniť algoritmus z MILENAGE na XOR alebo určiť konkrétnu hodnotu parametru RAND, s ktorou MILENAGE počíta. Pokiaľ táto hodnota nie je zadaná, MILENAGE hodnotu RAND vygeneruje prostredníctvom modulu secrets. Príkazom *use* je možné zadať meno súboru a meno zariadenia. mil.py podľa toho priradí potrebné hodnoty k výpočtu, ak nie je zadané, vygeneruje postupnosť RAND a vráti výsledné parametre.

U oboch príkazov platí, že pri zvolení režimu *verbose* dôjde k výpisu jednotlivých krokov, pri *quiet* vypíše skript iba autentizačný toke AUTN popísaný v 2.2.2.1. Naopak pri nezvolenom režime, funguje mil.py, ako ostatné kalkulačky a vráti ostatné požadované parametre (obrázok 4.1).

4.3.1 Implementácia MILENAGE

Väčšina implementácií, ako sú v ukážkach 4.3 a 4.4, funkcie f2, f3, f4, f5 a f5* uvádzajú v jednej spoločnej funkcii, kde sa iba špecifikujú jednotlivé vstupy z týchto funkcií. Avšak nástroj *mil.py*, ktorý bol implementovaný v rámci tejto práce, by v takomto prípade obsahoval veľa redundantného kódu. Je to spôsobené výpismi krokov, kvôli ktorým tiež nie je možné implementovať jednoduché operácie spôsom, akým sú funkcie zobrazené vo vyššie spomenutých ukážkach. Aby bol kód uhladený, funkcie f2–f5* boli

```
RAND: 123456789abcdef0
SQN||AMF||SQN||AMF: 00000000000190000000000000019000
SQN||AMF||SQN||AMF ⊕ OPC: 5c4f6d9dc744e7e64ba7106157a585fa
rot(SQN||AMF||SQN||AMF ⊕ OPC, r1): 4ba7106157a585fa5c4f6d9dc744e7e6
rot(SQN||AMF||SQN||AMF ⊕ OPC, r1) ⊕ c1 ⊕ Eₖ(RAND ⊕ OPC, k): f08b7223f507b2646dc14b8676fd0941
Eₖ(rot(SQN||AMF||SQN||AMF ⊕ OPC, r1) ⊕ c1 ⊕ Eₖ(RAND ⊕ OPC, k), k): b6c734f882d2b54a9e357a9d05c64e11
Eₖ(rot(SQN||AMF||SQN||AMF ⊕ OPC, r1) ⊕ c1 ⊕ Eₖ(RAND ⊕ OPC, k), k) ⊕ OPC: ea8859654597c2acd5926afc52625beb
MAC-A (f1): ea8859654597c2ac
MAC-S (f1*): d5926afc52625beb

Eₖ(RAND ⊕ OPC, k) ⊕ OPC: e7630fdf65e740787a29367ae61dfb5d
rot(Eₖ(RAND ⊕ OPC, k) ⊕ OPC, r2): e7630fdf65e740787a29367ae61dfb5d
rot(Eₖ(RAND ⊕ OPC, k) ⊕ OPC, r2) ⊕ c2: e7630fdf65e740787a29367ae61dfb5c
Eₖ(rot(Eₖ(RAND ⊕ OPC, k) ⊕ OPC, r2) ⊕ c2, k): beff756e3863fdb43ee55d08adc58919
Eₖ(rot(Eₖ(RAND ⊕ OPC, k) ⊕ OPC, r2) ⊕ c2, k) ⊕ OPC: e2b018f3ff268a5275424d69fa619ce3
AK (f5): e2b018f3ff26
RES (f2): 75424d69fa619ce3

Eₖ(RAND ⊕ OPC, k) ⊕ OPC: e7630fdf65e740787a29367ae61dfb5d
rot(Eₖ(RAND ⊕ OPC, k) ⊕ OPC, r3): 65e740787a29367ae61dfb5de7630fdf
rot(Eₖ(RAND ⊕ OPC, k) ⊕ OPC, r3) ⊕ c3: 65e740787a29367ae61dfb5de7630fdd
Eₖ(rot(Eₖ(RAND ⊕ OPC, k) ⊕ OPC, r3) ⊕ c3, k): 6f0cc98bbd0b9f4e6e7a1e269f1f2856
Eₖ(rot(Eₖ(RAND ⊕ OPC, k) ⊕ OPC, r3) ⊕ c3, k) ⊕ OPC: 3343a4167a4ee8a825dd0e47c8bb3dac
CK (f3): 3343a4167a4ee8a825dd0e47c8bb3dac

Eₖ(RAND ⊕ OPC, k) ⊕ OPC: e7630fdf65e740787a29367ae61dfb5d
rot(Eₖ(RAND ⊕ OPC, k) ⊕ OPC, r4): 7a29367ae61dfb5de7630fdf65e74078
rot(Eₖ(RAND ⊕ OPC, k) ⊕ OPC, r4) ⊕ c4: 7a29367ae61dfb5de7630fdf65e7407c
Eₖ(rot(Eₖ(RAND ⊕ OPC, k) ⊕ OPC, r4) ⊕ c4, k): 5af491d1c3c8215258105b226bb9d584
Eₖ(rot(Eₖ(RAND ⊕ OPC, k) ⊕ OPC, r4) ⊕ c4, k) ⊕ OPC: 06bbfc4c048d56b413b74b433c1dc07e
IK (f4): 06bbfc4c048d56b413b74b433c1dc07e

Eₖ(RAND ⊕ OPC, k) ⊕ OPC: e7630fdf65e740787a29367ae61dfb5d
rot(Eₖ(RAND ⊕ OPC, k) ⊕ OPC, r5): e61dfb5de7630fdf65e740787a29367a
rot(Eₖ(RAND ⊕ OPC, k) ⊕ OPC, r5) ⊕ c5: e61dfb5de7630fdf65e740787a293672
Eₖ(rot(Eₖ(RAND ⊕ OPC, k) ⊕ OPC, r5) ⊕ c5, k): 83d8e4e3d5ae7b5ad13478f41aa79cc7
Eₖ(rot(Eₖ(RAND ⊕ OPC, k) ⊕ OPC, r5) ⊕ c5, k) ⊕ OPC: df97897e12eb0cbc9a9368954d03893d
AK* (f5*): df97897e12eb

e2b018f3ff279000ea8859654597c2ac
```

(a) verbose

```
MAC-A: ea8859654597c2ac
MAC-S: d5926afc52625beb
AK: e2b018f3ff26 | df97897e12eb
RES: 75424d69fa619ce3
CK: 3343a4167a4ee8a825dd0e47c8bb3dac
IK: 06bbfc4c048d56b413b74b433c1dc07e

e2b018f3ff279000ea8859654597c2ac
```

(b) standard

```
e2b018f3ff279000ea8859654597c2ac
```

(c) quiet

Obr. 4.1 Ukážky výstupu skriptu mil.py

rozdelené do samostatných funkcií, podľa využitých konštánt c1–c5 a r1–r5 z 2.2.2.1.

Na spoločné časti bola implementovaná funkcia *__fproto__* 4.5, ktorá vykonáva hlavné časti algoritmu. Ako je vidieť na schéme algoritmu MILENAGE 2.6, podľa počiatočného vstupu (u f1 a f1* je ním $SQN||AMF||SQN||AMF$, u f2–f5* $E_k(RAND \oplus OPC, k)$) a tiež podľa operácií xor s konšantami c1–c5 (u konštanty c1 sa okrem hlavnej vetvy algoritmu pripája ešte aj vstup $E_k(RAND \oplus OPC, k)$), je možné rozdeliť funkcie podobne ako u ukážkových implementácií na f1 spolu s f1* (funkcia *__f1_f1star__*) a f2–f5* (funkcia *__f2345__* 4.6), avšak z týchto funkcií je volaná spoločná funkcia *__fproto__*, u ktorej je možné špecifikovať, aký vstup sa má použiť a o aké konštanty c1–c5 a r1–r5 sa jedná (pre f1 a f1* sa ako konštanta c prijíma $c1 \oplus E_k(RAND \oplus OPC, k)$), spolu s ich popismi kvôli jednotlivým výpisom.

Ku príkladu z funkcie *__f5_f2__* 4.7 je volaná funkcia *__f2345__* so spoločnými parametrami *rand, opc, key, encrypt, verbose* (s funkciami *__f3__*, *__f4__* a *__f5__*) a parametrom $idx = 2$, ktorý bude pre tieto funkcie odlišný. Vo funkcii *__f2345__* sa pridá spoločný vstup týchto funkcií ($E_k(RAND \oplus OPC, k)$), pripravia sa konštanty r a c s ich indexom a definujú sa ich výpisy. Takto pripravené parametre sú vstupnými parametrami funkcie *__fproto__*, ktorá s dodanými hodnotami vykoná samotný algoritmus.

Výpis kódu 4.3 Ukážka z implementácie funkcie *f2* algoritmu MILENAGE v programovacom jazyku C [50]

```
void f2345 ( u8 k[16], u8 rand[16], u8 res[8], u8 ck[16], u8 ik[16],
    ↪  u8 ak[6], u8 op[16] )
    {
    // ...
    //vypocet OPc (op_c)

    for (i=0; i<16; i++)
        rijndaelInput[i] = rand[i] ^ op_c[i];
    RijndaelEncrypt( rijndaelInput, temp );

    for (i=0; i<16; i++)
        rijndaelInput[i] = temp[i] ^ op_c[i];
    rijndaelInput[15] ^= 1;

    RijndaelEncrypt( rijndaelInput, out );
    for (i=0; i<16; i++)
        out[i] ^= op_c[i];

    for (i=0; i<8; i++)
        res[i] = out[i+8];
    for (i=0; i<6; i++)
        ak[i]  = out[i];
    // ...
    return;
    }
```

Výpis kódu 4.4 Ukážka z implementácie funkcie *f2* algoritmu MILENAGE v programovacom jazyku Python [51]

```
def f2345(self, K, RAND, OP=None):
    # osetrenie vstupu
    # vypocet OPc

    cipher = AES_ECB(K)
    K_OPc_RAND_OPc = xor_buf(cipher.encrypt(
                             xor_buf(OPc, RAND)),
                             OPc)
    out2 = xor_buf(OPc,
                   cipher.encrypt(
                   xor_buf(rot_buf16(K_OPc_RAND_OPc,
                                     self.r2),
```

```
                                self.c2)))
        # ...
        return out2[8:16], out3, out4, out2[:6]
```

Výpis kódu 4.5 Funkcia *__fproto__* (spoločná časť funkcií f1–f5*)

```
    def __fproto__(inp, opc, r, c, key, encrypt, inpName, rName, cName,
        ↪ verbose = True)
        result = inp ^ opc
        # ...
        result ^= opc
        if verbose:
            print("Ek(rot(%s ⊕ OPC, %s) ⊕ %s, k) ⊕ OPC: %032x" % (
                ↪ inpName, rName, cName, result))

        return result
```

Výpis kódu 4.6 Funkcia *__f2345__* (spoločná časť funkcií f2–f5*)

```
    def __f2345__(rand, opc, key, encrypt, idx, verbose = True):
        return __fproto__(__enc__(rand, opc, key, encrypt), opc, __R__[
            ↪ idx], __C__[idx], key, encrypt, "Ek(RAN ⊕ OPC, k)", "r%i"
            ↪ % idx, "c%i" % idx, verbose)
```

Výpis kódu 4.7 Funkcia *__f5_f2__* (funkcia f2 a f5)

```
    def __f5_f2__(rand, opc, key, encrypt, verbose = True):
        result = __f2345__(rand, opc, key, encrypt, 2, verbose)

        res = result & 0xffffffffffffffff
        ak = (result >> 80)

        if verbose:
            print("AK␣(f5):␣%012x␣\nRES␣(f2):␣%016x" % (ak, res))
        return ak, res
```

Kapitola 5
Záver

V práci boli preskúmané technológie sietí druhej až piatej generácie, s dôrazom na autentizáciu a zabezpečenie hlasových a dátových prenosov. Ďalej bol v rámci práce vytvorený nástroj, ktorý demonštruje kryptografické algoritmy MILENAGE a XOR, a nástroj, ktorý generuje hodnoty kryptografických parametrov využitých v sieťach vytvorených prostredníctvom srsRAN.

Práca popisuje základné pojmy, s ktorými sa pri mobilných sieťach stretávame, fungovanie mobilnej siete, rozdiely medzi jednotlivým generáciami, ich kryptografické algoritmy alebo technológie, ktoré môžu k zabezpečeniu prispieť.

Po preskúmaní technológií využívaných v mobilných sieťach štvrtej generácie boli skriptom mil.py naimplementované algoritmy MILENAGE a XOR, ktoré boli navrhnuté najmä pre účely ladenia. Tiež bol naimplementovaný nástroj sim.py, ktorý umožňuje generovať súbory s kryptografickými parametrami používané ku konfigurácii softvéru srsEPC projektu srsRAN.

Výsledky práce budú slúžiť pre testovacie a učebné účely vďaka konkrétnemu spracovaniu algoritmov MILENAGE a XOR a skriptu, ktorý umožňuje generovanie súboru využívaného v projekte vytvoreného pre testovanie sietí vo virtuálnom prostredí.

V budúcnosti by sa nástroj sim.py mohol rozšíriť o funkcie, prostredníctvom ktorých by sa generovaný súbor dal upravovať. Napríklad o funkciu na automatické generovanie viacero záznamov užívateľov a tiež vymazanie záznamu užívateľa. Nástroj mil.py, ktorý je v súčasnej dobe dostupný z príkazového riadku, by sa dal rozšíriť o grafické užívateľské rozhranie pre väčšiu prehľadnosť a jednoduchšiu manipuláciu.

Dodatok A

Užívateľská príručka

A.1 Inštalácia

Stiahnite a nainštalujte si interpreter pre Python 3 (v3.9+) a správcu balíčkov Pip (v22.0+). Následne pomocou správcu balíčkov Pip nainštalujte potrebné balíčky:

- click (v8.1+)
- pycryptodome (v3.14+)

Z priloženého média skopírujte adresár *src/impl* do svojho pracovného prostredia.

A.2 Použitie

A.2.1 sim.py

Pre vytvorenie prázdneho súboru user_db.csv použite príkaz A.1.

Výpis kódu A.1 Vytvorenie prázdneho súboru user_db.csv

```
python3 <cesta k skriptu sim.py> empty <cesta k user_db.csv>
```

Pre pridanie nového užívateľa do súboru user_db.csv použite príkaz A.2.

Výpis kódu A.2 Pridanie užívateľa do user_db.csv

```
python3 <cesta k skriptu sim.py> add [PREPINACE] <cesta k user_db.
    ↪ csv>
```

Pre detail prepínačov viď prepínač –help.

A.2.2 mil.py

Pre výpočet algoritmu MILENAGE so zadanými parametrami použite príkaz A.3.

Výpis kódu A.3 MILENAGE so zadanými parametrami

```
python3 <cesta k skriptu mil.py> compute [PREPINACE] SQN AMF OP KEY
```

Pre výpočet algoritmu MILENAGE s parametrami zo súboru user_db.csv A.4.

Výpis kódu A.4 MILENAGE s parametrami zo súboru user_db.csv

```
python3 <cesta k skriptu mil.py> use [PREPINACE] <cesta k user_db.
    ↪ csv> UE
```

Pre detail prepínačov viď prepínač –help.

Bibliografia

1. *What is a cellular network or mobile network?: Samsung Support India* [online]. 2020 [cit. 2022-04-24]. Dostupné z : https://www.samsung.com/in/support/mobile-devices/what-is-a-cellular-network-or-mobile-network/.
2. HEMACHANDRA, Kasun. *Performance Evaluation and Spectral Efficiency Improvements for Wireless Networks With Interference*. 2015. Dostupné z DOI: 10.13140/RG.2.2.27925.73443. Diz. pr.
3. UGWEJE, Okechukwu. Radio Frequency and Wireless Communications. In: *The Internet Encyclopedia*. 2004. ISBN 9780471482963. Dostupné z DOI: 10.1002/047148296X.tie151.
4. POPOOLA, Segun; NDUJIUBA, Charles. *Multiple Access Techniques: Design Issues in FDMA/TDMA*. 2016. Dostupné z DOI: 10.13140/RG.2.2.27152.81929.
5. *Multiple Access Schemes for Cellular Systems* [online]. 2016 [cit. 2022-04-24]. Dostupné z : https://www.electronics-notes.com/articles/connectivity/cellular-mobile-phone/multiple-access-schemes-technology-techniques.php.
6. *OFDMA* [online]. 2020 [cit. 2022-04-25]. Dostupné z : https://www.gta.ufrj.br/ensino/eel879/trabalhos_vf_2014_2/rafaelreis/ofdma_scfdma.html.
7. VOXO, Ivo Balaj. *Wi-Fi* [online]. 2020 [cit. 2022-04-25]. Dostupné z : https://elektrosmog-info.voxo.eu/wi-fi.
8. ALPER, C. Emre; MIKTUS, Michal; CLEMENTS, Mr. Benedict J. *Bridging the Mobile Digital Divide in Sub-Saharan africa: Costing under demographic change and urbanization* [online]. International Monetary Fund, 2019 [cit. 2022-04-25]. Dostupné z : https://www.elibrary.imf.org/view/journals/001/2019/249/article-A001-en.xml.
9. SALMAN, Hesham; ALDABBAGH, Ghadah; TAHA, Zaki; FATTOUH, Lamia. *Topological Planning and Design of Heterogeneous Mobile Networks in Dense Areas*. 2015. Dostupné z DOI: 10.1109/CSCI.2015.151.
10. USMAN, Muhammad Rehan; IQBAL, Johar; RAZZAQ, Fahad. *Performance Analysis of Channel Allocation Schemes in WiMAX*. 2009. Diz. pr.
11. MOSES, Kuboye. Mobile Communication Evolution. *International Journal of Modern Education and Computer Science*. 2014, roč. 6, s. 25–33. Dostupné z DOI: 10.5815/ijmecs.2014.01.03.
12. MEHROTRA, A.; GOLDING, L.S. Mobility and security management in the GSM system and some proposed future improvements. *Proceedings of the IEEE*. 1998, roč. 86, č. 7, s. 1480–1497. Dostupné z DOI: 10.1109/5.681375.

13. UR RAHMAN, Zia. GSM Technology: Architecture, Security and Future Challenges. *International Journal of Science Engineering and Advance Technology*. 2017, roč. 5, s. 70–74.
14. LAKSHAN, Manura. *GSM architecture* [online]. Medium, 2021 [cit. 2022-04-26]. Dostupné z : https://manura-lakshan.medium.com/gsm-architecture-db5edef8204d.
15. BAKAUL, Masum; ISLAM, Md; AHAD, H.M.; RAHMAN, Shayma. Security in GSM Networks. *International Journal of Computer Applications*. 2020, roč. 177, s. 36–39. Dostupné z DOI: 10.5120/ijca2020919774.
16. ROMIL GANDHI, Amitha Nair. GSM Networks: Substantiation of GSM Stationed algorithm. *International Journal of Scientific & Engineering Research* [online]. 2014, roč. 5 [cit. 2022-04-26]. ISSN 2229-5518. Dostupné z : https://www.ijser.org/paper/GSM-Networks-Substantiation-of-GSM-Stationed-algorithm.html.
17. IBRAHEM, Mahmood K. *GSM Security: Threats And Challenges* [online]. 2008 [cit. 2022-04-24].
18. *A5/1* [online]. Wikimedia Foundation, 2022 [cit. 2022-04-26]. Dostupné z : https://en.wikipedia.org/wiki/A5/1.
19. KARJALUOTO, Heikki. An Investigation of Third Generation (3G) Mobile Technologies and Services. *Contemporary Management Research Pages*. 2006, roč. 2, s. 91–104. Dostupné z DOI: 10.7903/cmr.653.
20. DETHAN, Jacob. *Wireless Broadband (UMTS)*. 2015. Dostupné z DOI: 10.13140/RG.2.1.1842.3766.
21. HOLMA, Harri; AHONPÄÄ, Timo; PRIEUR, Elisa. UMTS900 Co-Existence with GSM900. *2007 IEEE 65th Vehicular Technology Conference - VTC2007-Spring*. 2007, s. 778–782.
22. NOTES, Electronics. *3G UMTS Network Architecture* [online]. 2019 [cit. 2022-05-06]. Dostupné z : https://www.electronics-notes.com/articles/connectivity/3g-umts/network-architecture.php.
23. MONTILLET, Jean-Philippe. *Precise Positioning in Urban Canyons: Applied to the Localisation of Buried Assets* [online]. 2008 [cit. 2022-04-27]. Dostupné z DOI: 10.13140/RG.2.2.25999.69287. Diz. pr.
24. PÜTZ, Stefan; SCHMITZ, Roland; MARTIN, Tobias. Security Mechanisms in UMTS. *Datenschutz und Datensicherheit* [online]. 2001, roč. 25 [cit. 2022-04-24].
25. *LTE Authentication* [online]. 2017 [cit. 2022-04-27]. Dostupné z : https://www.sharetechnote.com/html/Handbook_LTE_Authentication.html.
26. SHAKER, Nabil; ISSA, Hanady; SHEHATA, K.A.; HASHEM, Somaia. Design of F8 Encryption Algorithm Based on Customized Kasumi Block Cipher. *International Journal of Computer and Communication Engineering* [online]. 2013, s. 398–402 [cit. 2022-04-27]. Dostupné z DOI: 10.7763/IJCCE.2013.V2.213.
27. *Kasumi (block cipher)* [online]. 2020 [cit. 2022-04-28]. Dostupné z : https://cryptography.fandom.com/wiki/KASUMI_(block_cipher).
28. OCHANG, Paschal A.; IRVING, Philip J. Evolutionary Analysis of GSM, UMTS and LTE Mobile Network Architectures. *World Scientific News*. 2016, roč. 54. ISSN 2392-2192.
29. IKKELÄ, Kalle; MYLLYNEN, Marko; HEINANEN, Juha; MARTIKAINEN, Olli. 4G Mobile Network Architecture. In: *Emerging Personal Wireless Communications*. Springer US, 2002, s. 183–195. ISBN 978-0-306-47001-1. Dostupné z DOI: 10.1007/0-306-47001-2_12.
30. *The standards of the 4G technology* [online]. 2021 [cit. 2022-04-24]. Dostupné z : https://whatsag.com/4g-and-lte-standards/understanding_4g.php.

31. ABED, Dr.Ghassan; ISMAIL, Mahamod; JUMARI, Kasmiran. The Evolution to 4G Cellular Systems: Architecture and Key Features of LTE-Advanced Networks. *International Journal of Computer Networks and Wireless Communications*. 2012, roč. 2.
32. *LTE network architecture* [online]. 2020 [cit. 2022-04-24]. Dostupné z : `https://www.tutorialspoint.com/lte/lte_network_architecture.htm`.
33. KUMAR, Ashish; ASWAL, Ankit; SINGH, Lalit. 4G Wireless Technology: A Brief Review. *International Journal of Engineering and management Research*. 2013, roč. 3, s. 35–43.
34. TRAN, Thien-Toan; SHIN, Yoan; SHIN, Oh-Soon. Overview of enabling technologies for 3GPP LTE-advanced. *EURASIP Journal on Wireless Communications and Networking*. 2012, roč. 2012. Dostupné z DOI: `10.1186/1687-1499-2012-54`.
35. DEEPTI; KUMAR, Dr Mukesh. Research Paper Based On E-UTRAN Architecture for LTE/ LTE-A Network Based On OMNET++. 2016.
36. VINTILĂ, Cristina-Elena; PATRICIU, Victor-Valeriu; BICA, Ion. *Security Analysis of LTE Access Network*. 2011.
37. *LTE, LTE-Advanced and WiMAX: Towards IMT-Advanced Networks* [online]. Wiley, 2021 [cit. 2022-04-24]. Dostupné z : `https://www.oreilly.com/library/view/lte-lte-advanced-and/9781119970453/ch14-sec002.html`.
38. *How to find the pin and PUK code?* [Online]. 2020 [cit. 2022-04-24]. Dostupné z : `https://www.vodafone.cz/pece/en/my-number/pin-and-puk/pin-and-puk-codes/`.
39. HUSSEIN, Soran. Lightweight Security Solutions for LTE/LTE-A Networks. 2014.
40. ORHANOU, Ghizlane; YOUSSEF, Bentaleb. SNOW 3G Stream cipher Operation and Complexity Study. *Contemporary Engineering Sciences*. 2010, roč. 3, s. 97–111.
41. MUTHALAGU, Raja; JAIN, Subeen. Modifying the structure KASUMI to improve its resistance towards attacks by inserting FSM and S-Box. *Journal of Cyber Security Technology*. 2018, roč. 2, s. 1–14. Dostupné z DOI: `10.1080/23742917.2018.1485415`.
42. ABDULLAH, Ako. Advanced Encryption Standard (AES) Algorithm to Encrypt and Decrypt Data. 2017.
43. PETERSON, Larry; SUNAY, Oğuz. *5G Mobile Networks: A Systems Approach*. 2020. ISBN 9781681738895. Dostupné z DOI: `10.2200/S01021ED1V01Y202006NSY001`.
44. *SrsRAN 22.04 documentation*. 2021. Dostupné tiež z: `https://docs.srsran.com/en/latest/index.html`.
45. PARK, Shinjo; SHAIK, Altaf; BORGAONKAR, Ravishankar; MARTIN, Andrew C.; SEIFERT, Jean-Pierre. White-Stingray: Evaluating IMSI Catchers Detection Applications. In: *WOOT*. 2017.
46. *CSV - csv file reading and writing* [online]. 2022 [cit. 2022-05-09]. Dostupné z : `https://docs.python.org/3/library/csv.html`.
47. *Secrets - generate secure random numbers for managing secrets* [online]. 2022 [cit. 2022-05-09]. Dostupné z : `https://docs.python.org/3/library/secrets.html`.
48. *AES* [online]. 2021 [cit. 2022-05-09]. Dostupné z : `https://pycryptodome.readthedocs.io/en/latest/src/cipher/aes.html`.
49. *Welcome to click* [online]. 2021 [cit. 2022-05-09]. Dostupné z : `https://click.palletsprojects.com/en/8.1.x/`.
50. *Milenage.c* [online]. 2007 [cit. 2022-05-10]. Dostupné z : `http://gull.sourceforge.net/dev/milenage_8c-source.html`. [Source code].
51. MICHAU, Benoit. *CryptoMobile/Milenage.py* [online]. 2013 [cit. 2022-05-10]. Dostupné z : `https://github.com/mitshell/CryptoMobile/blob/master/CryptoMobile/Milenage.py`. [Source code].

Obsah priloženého média

- `readme.txt` — stručný popis obsahu média
- `src` — zdrojové súbory
  - `impl` — zdrojové kódy implementácie
    - `lib` — adresár s modulmi implementovanej knižnice
      - `__init__.py` — inicializácia balíku
      - `common.py` — spoločné knižnicové funkcie
      - `enc.py` — šifrovacie funkcie
      - `mil.py` — autentizačné funkcie
      - `sim.py` — funkcie na generovanie súboru user_db.csv
      - `types.py` — konverzné funkcie pre spracovanie vstupu
    - `res` — ďalšie zdroje (nie zdrojový kód)
      - `template.csv` — šablóna súboru user_db.csv
    - `mil.py` — autentizačný skript
    - `sim.py` — skript na generovanie súboru user_db.csv
  - `thesis` — zdrojová forma práce vo formáte LaTeX
- `text` — text práce
  - `thesis.pdf` — text práce vo formáte PDF