VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV INFORMAČNÍCH SYSTÉMŮ
DEPARTMENT OF INFORMATION SYSTEMS

SS7 HONEYPOTY – PROAKTIVNÍ OCHRANA PROTI PODVODŮM V MOBILNÍCH SÍTÍCH

SS7 HONEYPOTS – PROACTIVE MOBILE NETWORKS FRAUD PROTECTION

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE **Bc. JURAJ KUBIŠ**
AUTHOR

VEDOUCÍ PRÁCE **Ing. JAN PLUSKAL**
SUPERVISOR

BRNO 2020

Vysoké učení technické v Brně
Fakulta informačních technologií

Ústav informačních systémů (UIFS) Akademický rok 2019/2020

Zadání diplomové práce

Student: **Kubiš Juraj, Bc.**
Program: Informační technologie Obor: Počítačové a vestavěné systémy
Název: **SS7 Honeypoty - proaktivní ochrana proti podvodům v mobilních sítích**
SS7 Honeypots - Proactive Mobile Networks Fraud Protection
Kategorie: Počítačové sítě

Zadání:

1. Prostudujte principy mobilních telefonních sítí se zaměřením na páteřní část, s protokoly rodiny SS7/SIGTRAN a jejich implementacemi. Dále prostudujte tzv. honeypoty a jejich základní dělení. Seznamte se s kybernetickými útoky na mobilní telefonní sítě.
2. Diskutujte různé typy útoků a navrhněte honeypot, jenž bude schopen na tyto útoky reagovat.
3. Navržený systém implementujte.
4. Otestujte funkčnost implementovaného systému.
5. Vyhodnoťte implementovaný systém a diskutujte jeho další možná vylepšení.

Literatura:

- Pužmanová, R.: Moderní komunikační sítě od A do Z, studijní opora, Computer Press 1998, ISBN 80-7226-098-7
- Schiller, J.: Mobile Communications (2nd edition), Addison-Wesley 2003, ISBN 0-321-12381-6

Při obhajobě semestrální části projektu je požadováno:

- Body 1 a 2 a rozpracování bodu 3.

Podrobné závazné pokyny pro vypracování práce viz https://www.fit.vut.cz/study/theses/

Vedoucí práce: **Pluskal Jan, Ing.**
Vedoucí ústavu: Kolář Dušan, doc. Dr. Ing.
Datum zadání: 1. listopadu 2019
Datum odevzdání: 20. května 2020
Datum schválení: 21. října 2019

Abstrakt

Diplomová práca sa zaoberá útokmi a podvodmi v mobilných sieťach. Cieľom práce je implementácia nástroja typu honeypot, ktorý je na tieto útoky schopný reagovať. Obsahom práce je preto základné predstavenie mobilných sietí, ich topológie, používaných sieťových protokolov a rozbor ich bezpečnosti. Ďalej je v práci ozrejmený význam pojmu honeypot, sú taktiež predstavené motivácie jeho nasadenia v sieti spolu s výhodami a nevýhodami. Zvyšok práce sa venuje samotnej implementácii nástroja, konkrétne jej návrhu, realizácie a testovania. V práci je predstavený spôsob reagovania na podporované podvody, detailný popis implementácie, konfigurácie a výstupov nástroja. Je tu opísaný proces testovania, či implementácia odpovedá predstavenému návrhu. Implementovaný nástroj je vyhodnotený a sú diskutované jeho ďalšie možné vylepšenia.

Abstract

This diploma thesis deals with the issue of attacks and fraud against mobile networks, with the main aim being implementation of a honeypot-type tool possessing the ability to respond to these accordingly. Thus, this thesis contains a basic introduction into mobile networks, their topology and commonly used protocols, along with analysis of their general security. This is followed by a clarification of the term honeypot itself, with an explanation of motivations for its deployment into the networks, together with listing of advantages and disadvantages such deployment may bring. The rest of the thesis deals with the actual implementation of such tool, specifically with its design, realisation and testing. This thesis presents a method for responding to the supported frauds, a detailed description of the implementation, configuration and outputs of the tool. The process of testing whether the implementation corresponds to the presented design is described here. The implemented tool is evaluated and its further possible improvements are discussed.

Kľúčové slová

GSM, SS7, honeypot, podvod

Keywords

GSM, SS7, honeypot, fraud

Citácia

KUBIŠ, Juraj. *SS7 Honeypoty – proaktivní ochrana proti podvodům v mobilních sítích.* Brno, 2020. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Jan Pluskal

SS7 Honeypoty – proaktivní ochrana proti podvodům v mobilních sítích

Prehlásenie

Prehlasujem, že som túto diplomovú prácu vypracoval samostatne pod vedením pána Ing. Jana Pluskala. Ďalšie informácie mi poskytol odborný konzultant Ing. Ondřej Pančocha. Uviedol som všetky literárne pramene, publikácie a ďalšie zdroje, z ktorých som čerpal.

.....
Juraj Kubiš
30. júla 2020

Poďakovanie

Rád by som poďakoval vedúcemu svojej práce Ing. Janu Pluskalovi a odborným konzultantom Ing. Ondřeji Pančochovi a Mgr. Bc. Hane Pluháčkovej za odbornú pomoc poskytnutú pri realizácii tejto práce.

Nakoľko vypracovanie tejto práce predstavuje zavŕšenie môjho štúdia na vysokej škole, rád by som za mentálnu a materiálnu podporu poďakoval svojej matke a otcovi a celej svojej rodine, ktorá ma po celú dobu môjho štúdia podporovala.

V neposlednom rade by som rád poďakoval svojim priateľom. Ďakujem Ing. Jurajovi Korčekovi, Ing. Viktoru Kovaříkovi, Ing. Tomáši Goldmanovi, Ing. Adriane Blaškovej, Bc. Pavlu Dohnalíkovi, Bc. Filipu Jaškovi, Bc. Miroslavovi Kažimírovi, Bc. Ondrejovi Dubajovi, Ing. Janu Bartoni, Adéle Bartoňové Slováčkové, Bc. Martine Grzybowskej, Ing. Kataríne Grešovej a Bc. Patriku Velemu za ich priateľstvo a skvelé zážitky, bez ktorých by som sa vo svojom štúdiu nikdy nedostal až sem.

Obsah

Zoznam použitých skratiek

3GPP The 3rd Generation Partnership Project. 8, 16

ATI Any Time Interrogation. 1, 20, 25, 28, 29, 41, 45

AuC Authentication Center. 11

BSC Base Station Controller. 10, 15

BSS Base Station Subsystem. 10

BTS Base Transceiver Station. 10, 13, 14

CC Country Code. 13

CdPA Called Party Address. 41

CGI Cell Global Identity. 13, 14, 48

CgPA Calling Party Address. 39, 41, 45

CID Cell ID. 13, 14, 28, 40

CSV Comma Separated Values. 2, 25, 33, 35–37, 39, 40, 44, 46–48

EDGE Enhanced Data rates for GSM Evolution. 8, 12

EIR Equipment Identity Register. 11

ETSI European Telecommunications Standards Institute. 8

GGSN Gateway GPRS Support Node. 11

GMSC Gateway Mobile Switching Centre. 10, 15, 18, 45

GPRS General Packet Radio Service. 5, 8, 11, 12

GSM Global System for Mobile Communications. 6–15, 18

GT Global Title. 16, 25, 31, 39, 41, 45, 46

HLR Home Location Register. 10, 11, 13, 18

IMEI International Mobile Equipment Identity. 10, 11, 27–29, 40

IMSI International Mobile Subscriber Identity. 6, 10, 12, 13, 16, 18, 19, 26–30, 40, 41, 45, 47, 48

IP Internet Protocol. 8, 11, 16, 17, 39

ISDN Integrated Services Digital Network. 10

ITU-T ITU Telecommunication Standardization Sector. 12, 15, 16

LA Location Area. 10, 27, 29

LAC Location Area Code. 13, 14, 28, 40

M2PA MTP2 Peer to Peer Adaptation Layer. 17, 33

M3UA MTP3 User Adaptation Layer. 17, 31, 33, 46

MAP Mobile Application Part. 16, 18, 19, 25–28, 30, 31, 36, 41, 42, 44–48

MCC Mobile Country Code. 12–14, 28, 39

ME Mobile Equipment. 10

MNC Mobile Network Code. 12–14, 28, 39

MOFWSM Mobile Originated Forward Short Message. 1, 20, 21, 25, 29, 30, 41, 42, 45

MS Mobile Station. 10, 11, 18, 21, 26–30

MSC Mobile Switching Centre. 10, 11, 15, 18, 20

MSIN Mobile Subscription Identification Number. 12, 13

MSISDN Mobile Subscriber ISDN Number. 10, 13, 16, 19, 26, 29, 30, 41, 45

MTFWSM Mobile Terminated Forward Short Message. 1, 20, 21, 25, 30, 41

MTP Message Transfer Part. 15–17, 31

NDC National Destination Code. 13

NSS Network Switching Subsytem. 9, 10, 16

OMC Operation and Maintenance Center. 11

OSS Operational Subsystem. 9

PSI Provide Subscriber Info. 1, 19, 20, 25, 27, 28, 41, 45

PSTN Public Switched Telephone Network. 10

RSS Radio Subsystem. 9, 12

SCCP Signaling Connection Control Part. 16

SCTP Stream Control Transmission Protocol. 17, 39

SGSN Serving GPRS Support Node. 11

SIF Signalling firewall. 31, 46

SIGTRAN Signaling Transport. 16, 17, 33, 42, 47

SIM Subscriber Identity Module. 10, 13

SIMSI Send IMSI. 1, 19, 25, 26, 41, 45

SM Short Message. 11, 20, 21, 29, 30, 45

SM-RP-UI Short Message Relay Protocol Unit. 29, 30, 41

SMSC Short Message Service Centre. 11, 20, 21, 29–31, 45

SN Subscriber Number. 13

SPC Signaling Point Code. 16, 17, 41

SRI Send Routing Information. 1, 18, 19, 25, 26, 41, 45, 47

SRIGPRS Send Routing Information for GPRS. 47

SRILCS Send Routing Information for Location Service. 47

SRISM Send Routing Information for Short Message. 47

SS7 Signaling System No. 7. 1, 2, 15–18, 20, 25, 31–33, 35, 42, 46, 47, 51

SSN Subsystem number. 16

TCAP Transaction Capabilities Application Part. 1, 16, 25, 31, 39, 41, 45

TPDU Transfer Protocol Data Unit. 29, 30, 45

UMTS Universal Mobile Telecommunications System. 8, 12

VLR Visitor Location Register. 10, 18, 19, 26, 27, 29, 40, 45

Kapitola 1

Úvod

V roku 1991 bola fínskym operátorom *Radiolinja* uvedená do prevádzky prvá komerčná GSM sieť. Na konci roku 2018 bolo v sieťach GSM druhej a tretej generácie celosvetovo aktívnych viac ako 4 miliardy zariadení[1]. Na základe tohoto počtu je teda možné konštatovať, že aj dnes je táto technológia stále masovo používaná a je pravdepodobné, že ešte nejakú dobu aj bude. Čo je však na tomto stave znepokojujúce, je to, že táto technológia je stará už viac ako 30 rokov.

Prvá verzia štandardu GSM bola zverejnená v roku 1987, teda v dobe, kedy nebol na bezpečnosť či ochranu súkromia a osobných údajov užívateľov braný taký zreteľ, ako je tomu dnes. Kvôli tomuto obsahujú aj dnešné GSM siete určité zraniteľnosti, ktoré sú priamo dané pôvodným návrhom štandardu a ich odstránenie by znamenalo porušenie spätnej kompatibility so staršími zariadeniami.

V súčasnosti je známe veľké množstvo rôznych útokov, či podvodov v týchto mobilných sieťach. Jedným z typických predstaviteľov je zneužitie služieb siete s cieľom sledovania polohy konkrétneho účastníka [17]. Nakoľko podobné typy útokov často vyžadujú ako prerekvizitu identifikáciu účastníka vo forme IMSI, vedú podobné útoky k ďalším útokom zneužívajúcim iné služby siete.

Iným typom útokov, ktoré nenarušujú súkromie jednotlivých účastníkov, ale ich dopady sú taktiež závažné, môže byť rozosielanie nevyžiadaných textových správ [24]. Takáto aktivita nielenže neprimerane obťažuje jej adresátov, ale spôsobuje výrazné finančné škody samotnému operátorovi.

Spoločnou charakteristikou útokov, ktorým sa táto práca venuje je, že jednotlivé scenáre útokov predstavujú, z pohľadu štandardov definujúcich fungovanie siete, plne legitímnu interakciu s jednotlivými prvkami siete. Tento fakt výrazne komplikuje odhaľovanie týchto útokov a vyžaduje detailnú analýzu sieťovej prevádzky.

Jednou z možností, ako túto prevádzku bezpečne získať, je aj nasadenie nástrojov typu honeypot v sieťach mobilných operátorov. Takýto nástroj navodzuje útočníkovi dojem, že interaguje s reálnou sieťou a že jednotlivé interakcie sú úspešné. Hoci v skutočnosti dostáva vymyslené údaje a teda súkromie účastníkov zostáva nenarušené, zdanlivosť úspešnej komunikácie s prvkami siete ho motivuje k dokončeniu celého zamýšľaného scenára útoku. Honeypot zaznamenáva detaily jednotlivých interakcií, čím predstavuje veľmi cenný zdroj dát o sieťovej prevádzke. Implementácia práve takéhoto nástroja je cieľom tejto diplomovej práce.

[1] zdroj https://www.statista.com/statistics/300014

Text tejto práce je členený do niekoľkých kapitol. Prvé dve kapitoly predstavujú teoretický úvod práce. Je to kapitola 2, v ktorej je čitateľ oboznámený s historickým vývojom, architektúrou a princípom činnosti GSM sietí a sú tu taktiež detailnejšie predstavené scenáre niektorých útokov. V kapitole 3 je zasa vysvetlený význam pojmu honeypot, je tu predstavené ich základné delenie, či ozrejmená motivácia pre nasadzovanie týchto nástrojov spolu s pozitívami a negatívami, ktoré to prináša.

Ostatné kapitoly sa venujú vytvorenému honeypotu a popisujú jednotlivé fázy prác. Kapitola 4 sa podrobne venuje návrhu implementovaného honeypotu, požiadavkám na neho kladených a je v nej detailne opísané spracovávanie podporovaných správ, či tvorba odpovedí. Nasledujúca kapitola 5 čitateľa podrobne zoznámi s implementačnými detailmi vytvoreného honeypotu. Sú tu rozobrané použité technológie, či opis jednotlivých tried implementácie. Nadväzujúca kapitola 6 potom pojednáva o fáze testovania implementovaného honeypotu, či o jeho experimentálnom nasadení. Celá práca je zakončená kapitolou 7 predstavujúcou jej záver, v rámci ktorej sú rekapitulované dosiahnuté výsledky.

Kapitola 2

Štandard GSM

V tejto kapitole sú prezentované základné informácie o historickom vývoji, či o architektúre GSM sietí. Konkrétne sú tu uvedené informácie o delení siete na rôzne logické podsystémy a informácie o jednotlivých sieťových elementoch, spolu s popisom funkcie, ktorú v sieti plnia.

Okrem toho sú tu predstavené dôležité sieťové protokoly, ktoré slúžia na signalizáciu a riadenie GSM siete. Sú tu spomenuté ako protokoly, na ktorých boli tieto siete založené, tak aj protokoly, na ktorých stoja tieto siete dnes. Je tu taktiež načrtnutý trend migrovania GSM sietí na IP infraštruktúru a motivácie k tomuto prechodu.

Posledná časť kapitoly je potom venovaná otázke bezpečnosti GSM sietí. V rámci tejto časti je čitateľ zoznámený zo základnou premisou, na ktorej bolo ich zabezpečenie založené a v neposlednom rade tu sú ukázané typické podvody v dnešných GSM sieťach.

2.1 História a počiatky štandardu

Na začiatku 80-tych rokov minulého storočia koexistovalo v Európe množstvo analógových mobilných sietí, ktoré síce boli na založené podobných štandardoch, ale boli prevádzkované na rozličných frekvenciách [21]. Dôsledkom bolo, že jednoduché používanie mobilných zariadení naprieč týmito sieťami nebolo možné. Preto bola v roku 1982 založená výskumná skupina *Groupe Spécial Mobile*, ktorej poslaním bolo navrhnúť druhú generáciu mobilných sietí, v ktorých by tento nedostatok absentoval. Tento systém dostal názov *Globálny systém pre mobilnú komunikáciu* – Global System for Mobile Communications (GSM) a bol vyvinutý inštitútom ETSI, s cieľom definovať protokoly pre druhú generáciu bunkových mobilných sietí.

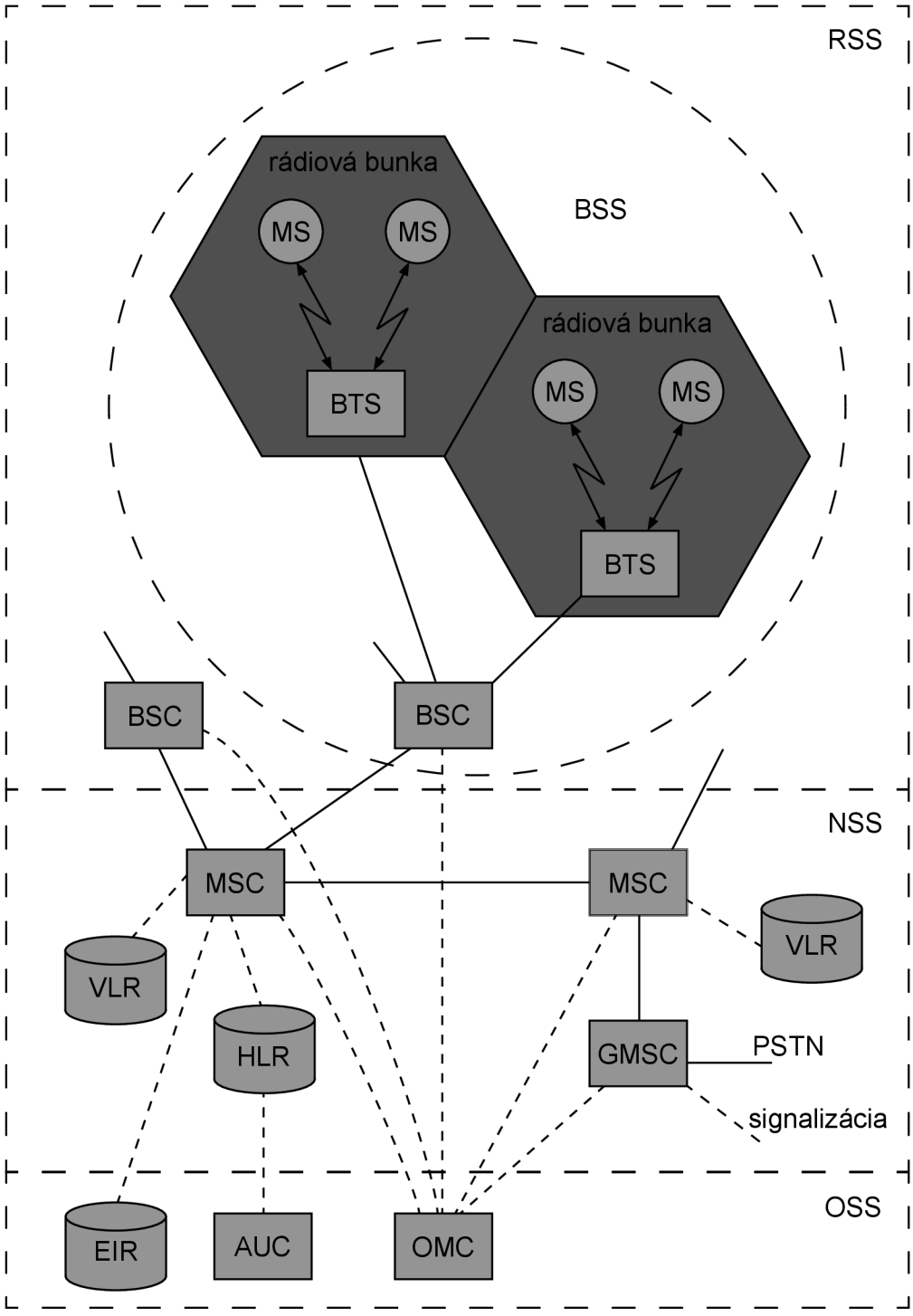
V nasledujúcich rokoch bol systém GSM viackrát rozšírený a modernizovaný. Hlavné zlepšenia spočívali najmä v zvyšovaní prenosových rýchlostí dátového pripojenia. V roku 1998 bol predstavený štandard GPRS, označovaný aj ako 2.5G, ktorý rozširoval pôvodnú architektúru GSM o nové sieťové prvky. O rok neskôr bol zase predstavený štandard EDGE, označovaný ako 2.75G – ďalší evolučný stupeň vychádzajúci z pôvodného štandardu.

V roku 1998 bol založený projekt 3GPP s cieľom definovať štandardy pre mobilné siete tretej generácie (3G). Výsledkom bol štandard UMTS, ktorý na rozdiel od minulých modernizácií vyžadoval aj nové frekvenčné rozsahy.

Nasledujúce generácie mobilných sietí nie sú priamo postavené na pôvodnom štandarde GSM a nesúvisia s témou tejto práce.

2.2 Architektúra a základné prvky siete

Celý systém GSM sa dá rozdeliť na tri základné podsystémy a to konkrétne na **rádiový podsystém** (RSS), **sieťový spojovací podsystém** (NSS) a **prevádzkový podsystém** (OSS) [21]. Toto rozdelenie je znázornené na obrázku 2.1.



Obr. 2.1: Architektúra siete GSM. Založené na [21].

Rádiový podsystém

Do rádiového podsystému zaraďujeme všetky entity, ktoré majú čo dočinenia s rádiovým spojením. Radíme sem nasledovné entity:

- **Mobilná stanica** (MS) predstavuje všetko užívateľské vybavenie potrebné pre komunikáciu so sieťou GSM. Skladá sa z užívateľsky nezávislého hardvérového a softvérového vybavenia (ME) a tzv. **modulu identity predplatiteľa** (SIM), ktorý obsahuje všetky potrebné údaje predplatiteľa.

  V súvislosti s MS je dôležité ozrejmiť ešte dva pojmy:

  - **IMEI** – 15-miestne jedinečné číslo identifikujúce ME,
  - **IMSI** – 15-miestne (väčšinou) jedinečné číslo identifikujúce predplatiteľa, ktoré je uložené v SIM module.

- **Základňová stanica** (BTS) je všetko rádiové vybavenie na strane operátora (antény, spracovanie signálov, zosilňovače, ...).

- **Kontrolér základňových staníc** (BSC) riadi prislúchajúce BTS, rezervuje a spravuje rádiové frekvencie, zabezpečuje handover[1], ...

BSC spolu s množinou ním kontrolovaných BTS tvoria logický celok nazývaný **podsystém základňových staníc** (BSS).

Sieťový spojovací podsystém

NSS je srdcom celého systému GSM. Zabezpečuje služby ako handover, účtovanie služieb, autorizáciu prístupu do siete, roaming[2], lokalizáciu a mnoho ďalších. Všetky tieto služby sú poskytované pomocou nasledujúcich prvkov:

- **Ústredňa verejnej mobilnej siete** (MSC) je vysokovýkonná digitálna ISDN ústredňa, ktorá spája BSC a ostatné MSC. Typicky jedna MSC obsluhuje viacero BSC v geografickej lokalite. Špeciálny typ (GMSC) má navyše pripojenie na iné siete (napríklad PSTN).

- **Domovský register** (HLR) je najdôležitejšia databáza v sieti GSM. Obsahuje informácie o užívateľoch siete, ktoré sú rozdeľované na statické a dynamické.

  Medzi statické radíme informácie ako MSISDN (celosvetovo unikátne číslo identifikujúce užívateľa vo verejnej telefónnej sieti – tj. telefónne číslo v bežnom chápaní), zoznam predplatených služieb, prípadne IMSI.

  Z dynamických informácii stojí určite za zmienku aktuálna geografická poloha MS v sieti – tzv. Location Area (LA).

- **Návštevnícky register** (VLR) je lokálna databáza každého MSC, v ktorej sú uchovávané iba informácie potrebné na obslúženie MS nachádzajúcej sa v LA asociovanej s daným MSC.

  Pri registrácii novej MS v LA sú z HLR stiahnuté všetky potrebné informácie, ktoré sa uložia do VLR. Týmto sa eliminuje častá aktualizácia údajov v HLR a signalizácia užívateľských informácii na dlhé vzdialenosti.

[1] handover je automatické predanie obsluhy spojenia inému BSC
[2] roaming je poskytovanie mobilných služieb v sieti iného operátora

Pri roamovaní užívateľa v cudzej sieti sú všetky potrebné informácie kopírované z HLR nachádzajúceho sa v domovskej sieti jeho operátora.

- **Stredisko krátkych textových správ** (SMSC) je prvok siete GSM, ktorého úlohou je doručovanie textových správ (SM). Každá odoslaná SM je cez MSC odosielateľa zaslaná práve do SMSC, odkiaľ je zase preposlaná adresátovi (skrz jeho MSC).

  Dôvodom takéhoto nepriameho zasielania je, že adresát nemusí byť v dobe odosielania správy dostupný, alebo nemôže prijať správu z iných dôvodov (napríklad plná pamäť v jeho MS). V takomto prípade je správa dočasne uchovaná v SMSC, ktoré správu doručí neskôr.

  Odosielateľom, alebo adresátom SM, nemusí byť len iný účastník siete GSM, ale môžu to byť aj iné entity či služby, nakoľko doručovanie správ vždy prebieha skrz SMSC (notifikácia o zmeškanom hovore, SMS hlasovanie, SMS autentifikácia, ...).

Prevádzkový podsystém

- **Prevádzkové a údržbové stredisko** (OMC) monitoruje a kontroluje všetky ostatné sieťové prvky. Zabezpečuje služby ako monitorovanie sieťovej prevádzky, účtovanie a fakturovanie služieb a mnohé iné.

- **Autentifikačné stredisko** (AuC) zabezpečuje ochranu identity užívateľa a ochranu prenášaných dát, šifrovanie prevádzky a uchovávanie šifrovacích kľúčov. Z podstaty veci môže byť AuC patrične zabezpečená súčasť HLR.

- **Register mobilných staníc** (EIR) je databáza obsahujúca IMEI blokovaných MS (tzv. blacklist), alebo naopak povolených MS (tzv. whitelist) v sieti operátora. Najčastejšie použitie je blokovanie odcudzených mobilných telefónov.

Siete vyšších generácií

Pôvodné siete GSM boli primárne navrhované k hlasovým prenosom. Dátové prenosy v nich síce boli možné, nakoľko aj prenos samotného hlasu je prenosom dátovým (sieť GSM je digitálna), ale neoptimálnym spôsobom. Siete GSM totiž používajú prepínanie okruhov, takže pri dátovom prenose bol užívateľovi vytvorený komunikačný kanál s veľmi malou prenosovou rýchlosťou (v pôvodných GSM sieťach 9,6 kbit/s [21]) a samotný prenos bol účtovaný podľa doby trvania spojenia.

Riešením tohoto problému bolo pridanie podpory pre paketovo orientovaný prenos dát [21]. Z pohľadu topológie siete bola sieť rozšírená o dva nové prvky:

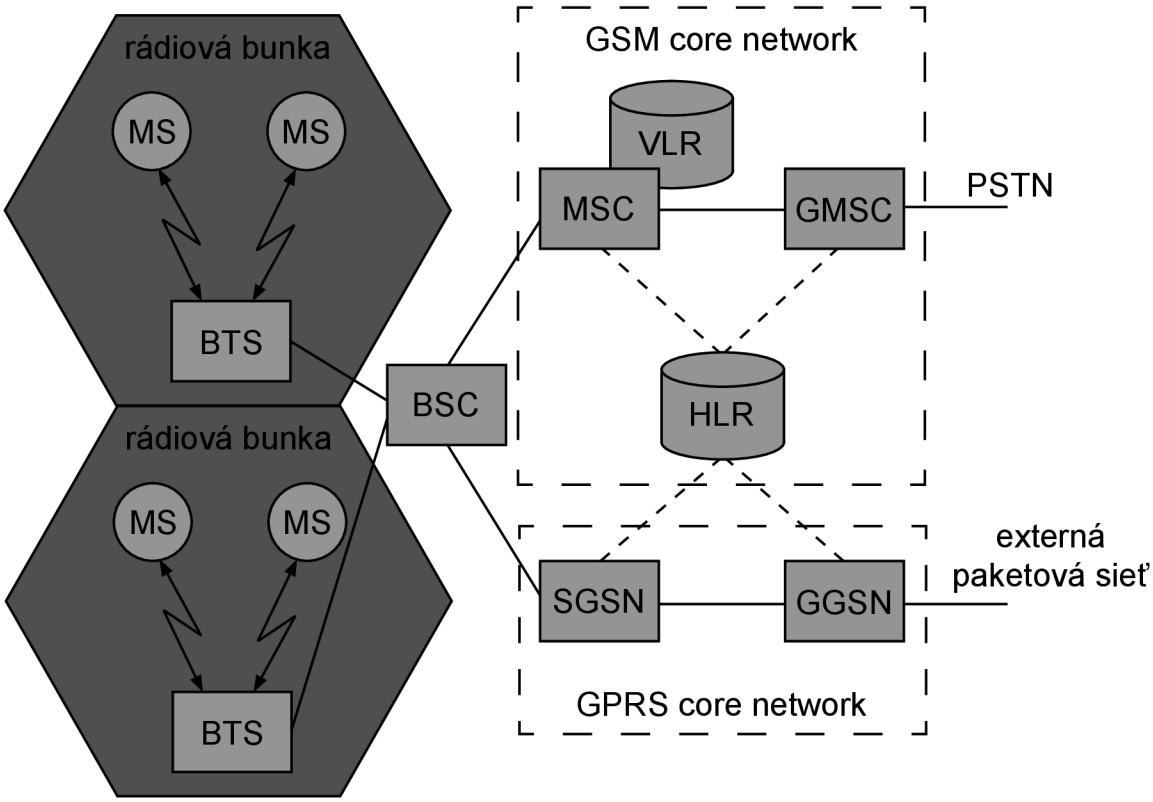
- **Obslužný uzol podpory GPRS** (SGSN) je zodpovedný za obsluhovanie MS, sleduje polohu užívateľa v sieti, vykonáva účtovanie služieb, a iné. Hierarchicky je na rovnakej úrovni ako HLR [21].

- **Bránový uzol podpory GPRS** (GGSN) je uzol, ktorý prepojuje sieť GPRS s externou paketovou sieťou (napríklad IP sieťou). Obsahuje informácie pre smerovanie dát k GPRS účastníkom, stará sa o konverziu adries, tunelovanie a zapuzdrovanie dát, a iné [21].

Dva vyššie spomenuté uzly boli predstavené spolu so vznikom štandardu GPRS. Z tohoto dôvodu býva táto časť siete niekedy označovaná aj ako **GPRS core network** a pô-

vodná časť ako **GSM core network**, prípadne ako **packet-switched** a **circuit-switched core network** [11].

Po štandarde GPRS (2.5G) boli predstavené ešte ďalšie technológie, ktoré ďalej zvyšovali prenosové rýchlosti. Boli to napríklad štandard EDGE (2.75G), či štandard UMTS (3G). Inovácie predstavené v týchto štandardoch sa však predovšetkým týkali RSS (nové spôsoby modulácie signálu, frekvenčné pásma, šírky kanálov, ...), ale z pohľadu signalizačnej časti siete neprišlo k výraznejším zmenám. Na obrázku 2.2 je možné vidieť štruktúru takejto mobilnej siete.



Obr. 2.2: Architektúra siete GSM s podporou paketového prenosu dát. Založené na [19].

2.3 Základné identifikátory a ich formáty

Sieť GSM obsahuje veľké množstvo rôznorodých prvkov či entít, ktoré nesú určité označenie alebo identifikáciu. Nakoľko sa v tejto práci bude s týmito termínmi operovať, je vhodné čitateľa s niektorými bližšie oboznámiť.

- **International Mobile Subscriber Identity (IMSI)** je dekadické číslo s maximálnou (aj typickou) dĺžkou 15 číslic. Toto číslo je jedinečný globálny identifikátor účastníka v sieti GSM, ktorý je zložený z troch častí: MCC, MNC a MSIN. Formát IMSI a jeho častí je definovaný v štandarde ITU-T E.212 [4].
- **Mobile Country Code (MCC)** je trojmiestne dekadické číslo, ktoré jednoznačne identifikuje domovskú krajinu operátora. Kódy jednotlivých krajín nie sú úplne náhodné, prvé číslo identifikuje geografickú oblasť, v ktorej daná krajina leží (napríklad 2 označuje oblasť Európy, alebo 3 región Severnej Ameriky a Karibiku).

- **Mobile Network Code (MNC)** je dvojmiestne alebo trojmiestne dekadické číslo, ktoré jednoznačne identifikuje sieť konkrétneho operátora v rámci jednej krajiny. Prideľovanie a spravovanie týchto kódov je v réžii jednotlivých krajín. Dvojica MCC a MNC predstavuje jednoznačný globálny identifikátor GSM siete operátora.

- **Mobile Subscription Identification Number (MSIN)** je posledná časť IMSI. Je to opäť dekadické číslo, ktoré je dlhé maximálne 10 číslic (jeho dĺžka je závislá na dĺžke MNC, ako aj celého IMSI). Toto číslo identifikuje účastníka v rámci siete jeho operátora, ktorý taktiež spravuje ich prideľovanie.

- **Mobile Subscriber ISDN Number (MSISDN)** slúži, podobne ako IMSI, k identifikácii účastníka na globálnej úrovni. Na rozdiel od IMSI, s ktorým bežný užívateľ neprichádza do kontaktu a slúži na interné účely siete, MSISDN predstavuje telefónne číslo účastníka, s ktorým užívateľ bežne pracuje. Ďalšou odlišnosťou je, že zatiaľ čo IMSI je pevne späté so SIM, mapovanie MSISDN na SIM (konkrétne na jej IMSI) je vykonávané sieťou (uložené v HLR), a teda je možné ho meniť – jedna SIM môže byť asociovaná s viacerými MSISDN.

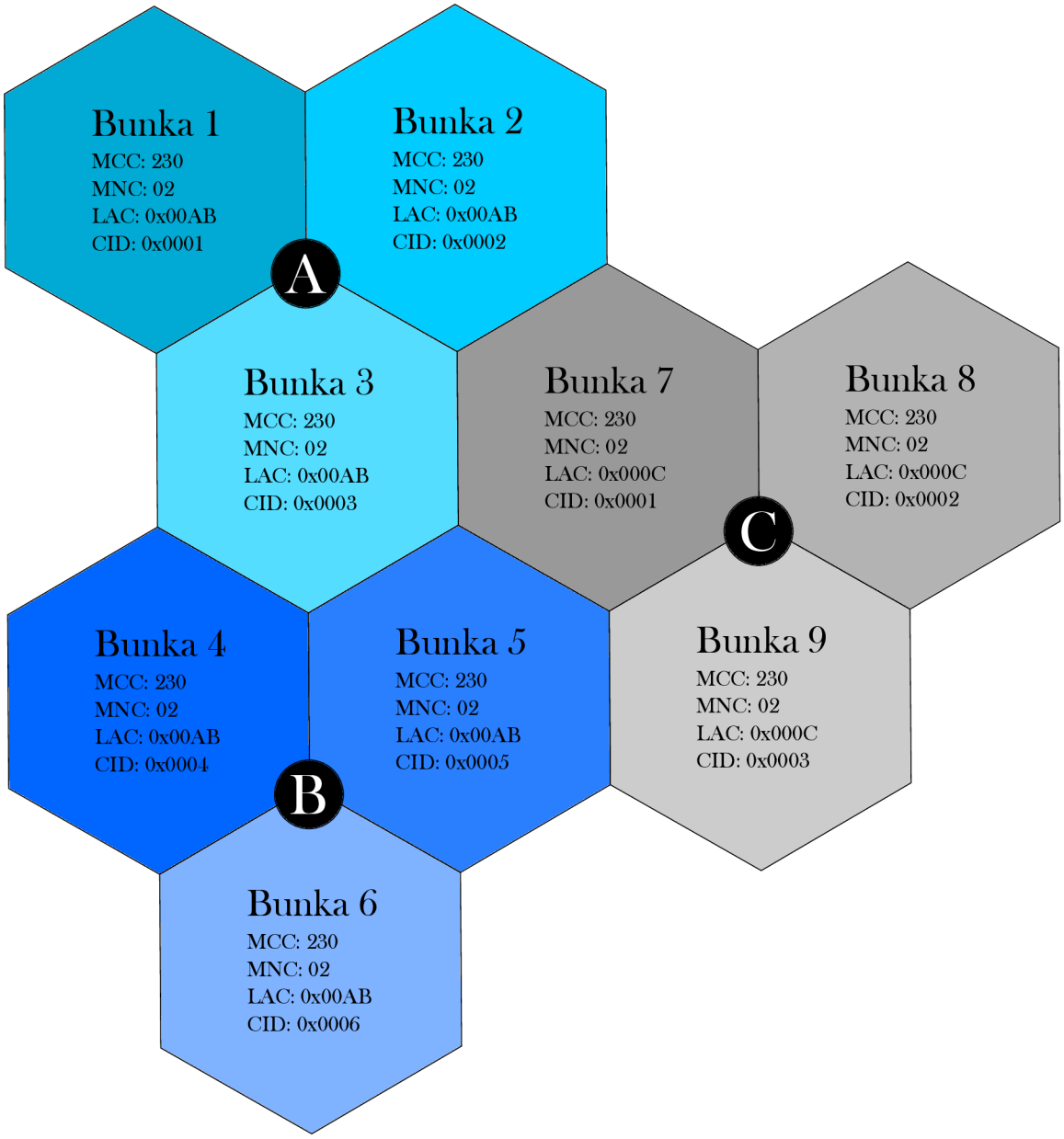
  MSISDN je taktiež zložený identifikátor, ktorý v sebe zahrňuje Country Code (CC), ktorý sa bežne označuje ako medzinárodné smerové číslo (alebo predvoľba) danej krajiny (420 pre ČR, či 421 pre SR). Druhá časť MSISDN je National Destination Code (NDC), ktorý identifikuje sieť operátora, alebo jej časť a Subscriber Number (SN) identifikujúce samotného účastníka. Tento formát je definovaný v štandarde E.164 [3].

- **Cell Global Identity (CGI)** je jedinečný globálny identifikátor konkrétnej bunky. Opäť sa jedná o zložený identifikátor a to z MCC, MNC, LAC a CID.

- **Location Area Code (LAC)** je označenie skupiny (tzv. *location area*) BTS, združených s cieľom optimalizovania smerovania signalizácie v sieti. Je to 16-bitové číslo.

- **Cell ID (CID)** je identifikátor bunky v rámci LAC. K jednej BTS môže prislúchať jedno CID (ak BTS obsluhuje jednu bunku), alebo viacero (ak BTS obsluhuje viacero sektorov). Je to 16-bitové číslo.

Na obrázku 2.3 je znázornená bunková štruktúra GSM siete. Konkrétne tu je zobrazených 9 buniek, ktoré sú obsluhované tromi BTS (A, B a C). Bunky 1 až 6 patria do jednej *location area*, preto majú totožné LAC a zároveň odlišné CID. Bunky 7 až 9, ktoré sú obsluhované BTS C, patria do inej *location area*, preto majú iné LAC, ale CID v nich používané sú jedinečné iba medzi sebou navzájom.



Obr. 2.3: Bunková štruktúra siete GSM.

V súvislosti s bunkovou štruktúrou siete GSM a identifikátormi jednotlivých buniek je vhodné zmieniť komunitnú službu *OpenCellID*[3]. Je to voľne prístupná databáza (podľa správcov služby aj najväčšia na svete), ktorá zhromažďuje GPS pozície všetkých BTS na svete spolu s identifikátormi prislúchajúcich buniek (CGI). Tieto informácie sú primárne zbierané členmi komunity, ktorí majú vo svojom smartfóne nainštalovanú aplikáciu, alebo pomocou špecializovaného hardvéru. Služba, mimo iné, umožňuje priamo na jej webových stránkach lokalizovať konkrétnu bunku na mape pomocou jej CGI (t.j. pomocou štvorice MCC, MNC, LAC a CID).

[3] http://www.opencellid.org/

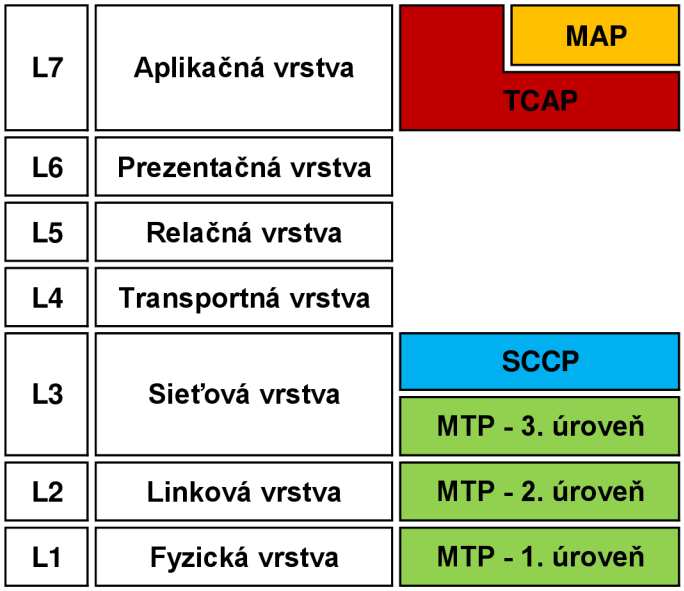
2.4 Signalizácia a protokoly

Signalizácia je súhrnné označenie tej sieťovej prevádzky, ktorá slúži na správu a réžiu siete a nie na samotnú komunikáciu účastníkov v nej. Z pohľadu siete GSM prebieha všetka účastnícka komunikácia (až na malé výnimky) medzi BSC, MSC, prípadne GMSC, a ostatnú prevádzku preto možno označiť za komunikáciu signalizačnú.

Štandard GSM podrobne špecifikuje, ako má táto komunikácia vyzerať, či aké protokoly sa pri nej majú použiť. Všetky tieto informácie sú definované v štandarde SS7, ktorý je spravovaný organizáciou ITU-T.

Signalizačný systém č. 7 (SS7)

Štandard SS7 definuje protokoly naprieč všetkými vrstvami modelu OSI/ISO, čo znamená, že špecifikuje nie len aplikačné protokoly, ale aj samotný sieťový hardvér, smerovanie v sieti, garanciu doručenia, a podobne. Na obrázku 2.4 je znázornený referenčný model OSI/ISO signalizačnej siete SS7. Ako je možné pozorovať, jednotlivé SS7 protokoly nie úplne korešpondujú s jednotlivými vrstvami ISO/OSI modelu. Podľa T. Russel [20] je to čiastočne kvôli tomu, že protokoly SS7 vznikli skôr ako ISO/OSI model.



Obr. 2.4: Sieťový zásobník SS7. Prevzaté z [20].

- **Message Transfer Part** (MTP) je transportný protokol používaný všetkými ostatnými SS7 protokolmi. Podľa štandardu ITU-T Q.701 [12] je protokol rozdelený na tri úrovne, ktoré zabezpečujú služby 1., 2. a čiastočne aj 3. vrstvy modelu ISO/OSI.
  - 1. úroveň špecifikuje fyzikálne, elektrické a funkčné vlastnosti dátového spojenia a spôsob prístupu k nemu,
  - 2. úroveň zabezpečuje spoľahlivý prenos dát, či detegovanie a korekciu chýb,

  - 3. úroveň poskytuje 4 funkcie – smerovanie (iba v rámci jednej siete na základe Signaling Point Code (SPC)), diskrimináciu[4] a distribúciu[5] signalizačných správ a správu siete.

- **Signaling Connection Control Part** (SCCP) je protokol sieťovej vrstvy modelu ISO/OSI. Podľa štandardu ITU-T Q.711 [13], v ktorom sú opísané jeho funkcie, je protokol SCCP nadstavbou protokolu MTP a rozširuje jeho možnosti, aby dokázal zabezpečiť nespojovanú aj spojovanú komunikáciu medzi uzlami SS7 siete, čím implementuje aj služby transportnej vrstvy.

  Protokol, mimo iné, umožňuje aj smerovanie signalizačných správ naprieč sieťami – smerovanie prebieha na základe IMSI, či MSISDN, ktoré sú súčasťou štruktúry nazývanej Global Title (GT).

  Okrem GT môže adresa príjemcu obsahovať SPC (smerovanie iba v rámci siete), alebo tzv. Subsystem number (SSN), pomocou ktorého je možné rozlíšiť jednotlivé služby bežiace na rovnakom uzle. Tieto a všetky ostatné parametre sú definované v štandarde ITU-T Q.713 [14].

- **Transaction Capabilities Application Part** (TCAP) je protokol, ktorý podľa štandardu Q.771 [15] umožňuje, mimo iné, vytvárať nezávislé dialógy medzi tými istými podsystémami (identifikovanými SSN) na rovnakom uzle. Každý dialóg je potom identifikovaný vlastným ID.

- **Mobile Application Part** (MAP) je protokol aplikačnej vrstvy, v súčasnej dobe spravovaný organizáciou 3GPP a špecifikovaný v špecifikáciách 3GPP TS 09.02 [1] a 3GPP TS 29.002 [2], v ktorých sú definované tzv. MAP služby (services) a MAP procedúry.

  MAP služba je primitívna funkcionalita, ktorú daný sieťový prvok poskytuje. Pre každú službu špecifikácia definuje štruktúru MAP správy s povinnými a voliteľnými parametrami, a ktoré sieťové prvky sú legitímnymi odosielateľmi takejto správy. Týmto spôsobom sú definované komunikačné rozhrania jednotlivých sieťových prvkov NSS.

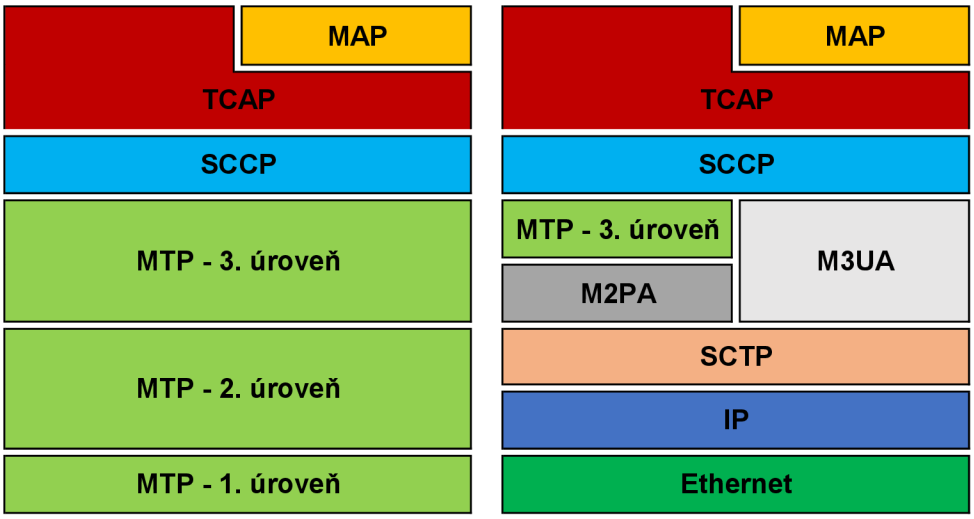
  MAP procedúra je komplexnejšia služba poskytovaná NSS, ktorá typicky pozostáva z výmeny viacerých MAP správ. Zjednodušene sa dá povedať, že to je sled MAP služieb, ktoré treba v danom poradí a medzi určenými prvkami zavolať, aby sa dosiahol požadovaný výsledok.

SIGTRAN

Signaling Transport (SIGTRAN) je sada protokolov umožňujúcich prenášanie SS7 signalizácie pomocou IP infraštruktúry. Tieto protokoly nahradzujú všetky úrovne MTP a sú spätne kompatibilné s protokolmi vyšších vrstiev pôvodného SS7 zásobníku (stacku). Tento vzťah je znázornený na obrázku 2.5.

[4] diskriminácia správy je rozhodnutie, či je správa určená pre uzol, ktorý ju práve spracúva

[5] distribúcia správy je rozoznanie služby uzlu, pre ktorú je správa určená a zaslanie správy tejto službe



Obr. 2.5: Porovnanie tradičného sieťového zásobníka SS7 (vľavo) a sieťového zásobníka SIGTRAN (vpravo). Založené na [20].

Hlavnou motiváciou k prechodu na IP infraštruktúru je podľa M. Immonen [6] odľahčenie SS7 sietí a umožnenie ich lepšej škálovateľnosti. Technológia SIGTRAN taktiež znižuje náklady na budovanie a rozširovanie sietí (napríklad na prepojenie geograficky odľahlých oblastí nie je nutné budovať drahú SS7 infraštruktúru).

Rodina protokolov SIGTRAN obsahuje viacero protokolov, ale za zmienku stoja predovšetkým nasledovné protokoly:

- **Stream Control Transmission Protocol** (SCTP) je transportný protokol, ktorý podľa normy RFC 4960 [16] poskytuje nasledovné služby:
  - spoľahlivý prenos dát s potvrdzovaním a so správnym poradím doručovania segmentov,
  - fragmentáciu dát pre splnenie maximálnej MTU,
  - prenos dát vo viacerých logických prúdoch (streamoch),
  - zlučovanie správ z viacerých prúdov do jedného paketu,
  - zvýšenú spoľahlivosť spojenia vďaka podpore multihomingu[6],
  - a ďalšie.

  SCTP je spojovaný protokol, kde každý koniec spojenia je tzv. *endpoint*, ktorý je identifikovaný jednou, alebo viacerými IP adresami a SCTP portom [7].

- **MTP2 Peer to Peer Adaptation Layer** (M2PA) a **MTP3 User Adaptation Layer** (M3UA) sú protokoly poskytujúce služby 2., resp. 2. a 3. úrovne MTP a využívajúce služieb SCTP. Dôležitou úlohou je napríklad mapovanie SPC na IP adresy. Protokol M3UA taktiež odstraňuje limitácie protokolu MTP spočívajúce v maximálnej veľkosti prenášanej správy. Protokoly sú definované v RFC 4165 [5] a v RFC 4666 [9].

[6] nadviazanie spojenia pomocou viacerých IP ciest

2.5 Bezpečnosť

V súčasnej dobe je známych niekoľko rôznych útokov a podvodov, ktoré zneužívajú slabiny štandardu GSM. Tento fakt však nie je ničím prekvapivý, ak zohľadníme dobu a podmienky obdobia vzniku štandardu.

V 90-tych rokoch minulého storočia bola situácia na trhu mobilných operátov diametrálne odlišná od tej dnešnej. Jednotlivý operátori boli totižto vlastnení jednotlivými štátmi a tým pádom boli považovaní aj za dôveryhodných. S rozmachom budovania sietí a so vstupom súkromných subjektov, ktorí tak získali prístup k SS7 sieti, sa však tento stav zmenil.

Štandard GSM a rodina protokolov SS7 boli navrhované s predpokladom vzájomnej dôvery medzi prevádzkovateľmi jednotlivých, navzájom prepojených, národných sietí [8]. A práve na tomto fakte je založených mnoho útokov a podvodov, ktoré budú na nasledujúcich riadkoch predstavené.

Typické podvody

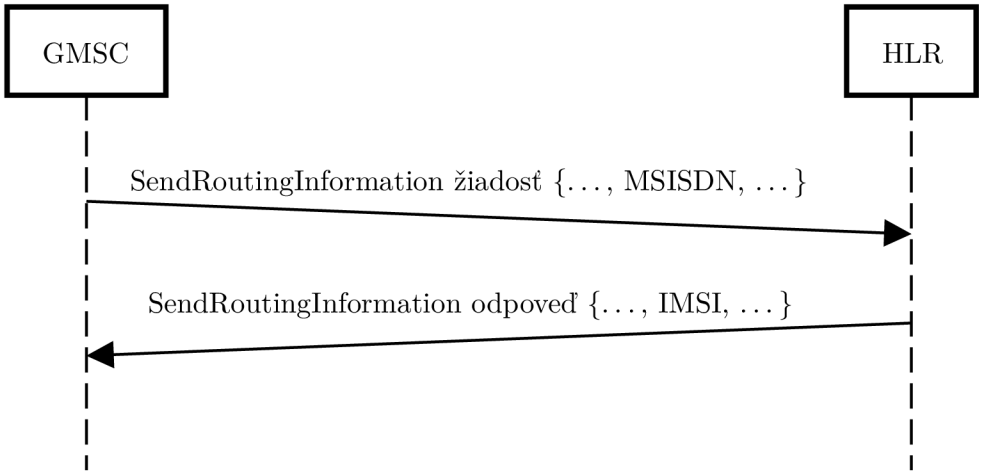
Na nasledujúcich riadkoch budú predstavené niektoré bežné scenáre podvodov v SS7 sieťach. Spoločnou vlastnosťou všetkých nižšie prezentovaných podvodov je, že útočník sa vydáva za legitímny sieťový prvok (MSC, VLR, ...) a s využitím jednotlivých MAP služieb komunikuje s ostatnými sieťovými prvkami. Použité MAP služby, falošná identita útočníka a dopytovaný prvok sú pritom častokrát plne v súlade so špecifikáciou a z jej pohľadu teda ide o legitímnu komunikáciu.

IMSI retrieval

Ako už bolo vysvetlené vyššie, IMSI je základný identifikátor každého predplatiteľa v sieti (pozri 2.3). Z tohoto dôvodu slúži aj ako primárny kľúč mnohých tabuliek v HLR, VLR, či v iných databázach a pokiaľ chce útočník získať osobné údaje o užívateľovi, je často nutné najskôr získať jeho IMSI.

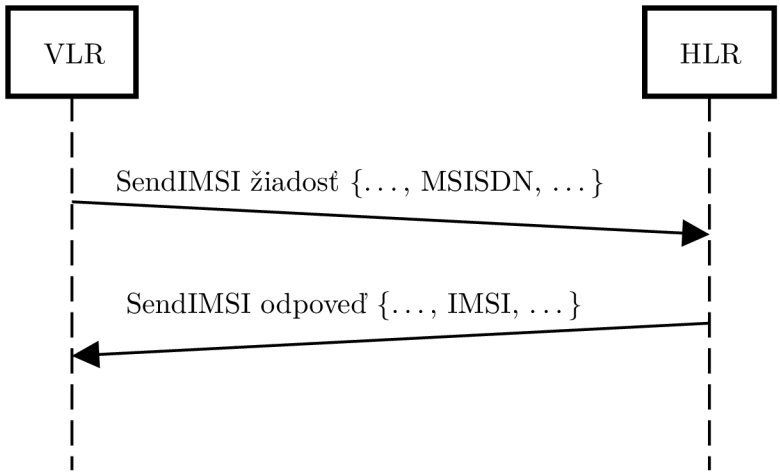
Získanie IMSI predstavuje pre útočníka prekážku, nakoľko IMSI užívateľa nie je verejne známe. K jeho získaniu môže útočník zneužiť jednu z nasledujúcich MAP služieb:

- **Send Routing Information** (SRI) je služba, pomocou ktorej GMSC zisťuje informácie potrebné pre smerovanie hovoru k volanej MS a jednou z týchto informácii je práve aj IMSI. Použitie tejto služby je znázornené na obrázku 2.6.



Obr. 2.6: Využitie služby SRI na získanie IMSI užívateľa.

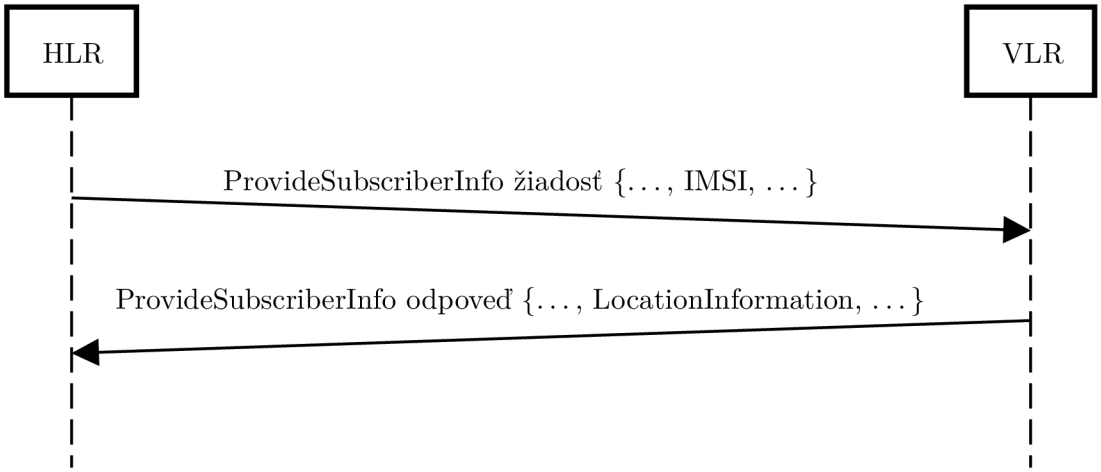
- **Send IMSI** (SIMSI) je služba používaná VLR na získanie IMSI užívateľa pre potreby hostujúcej siete v prípade, keď je jeho jediná známa identifikácia MSISDN. Obrázok 2.7 zobrazuje použitie tejto služby.



Obr. 2.7: Využitie služby SIMSI na získanie IMSI užívateľa.

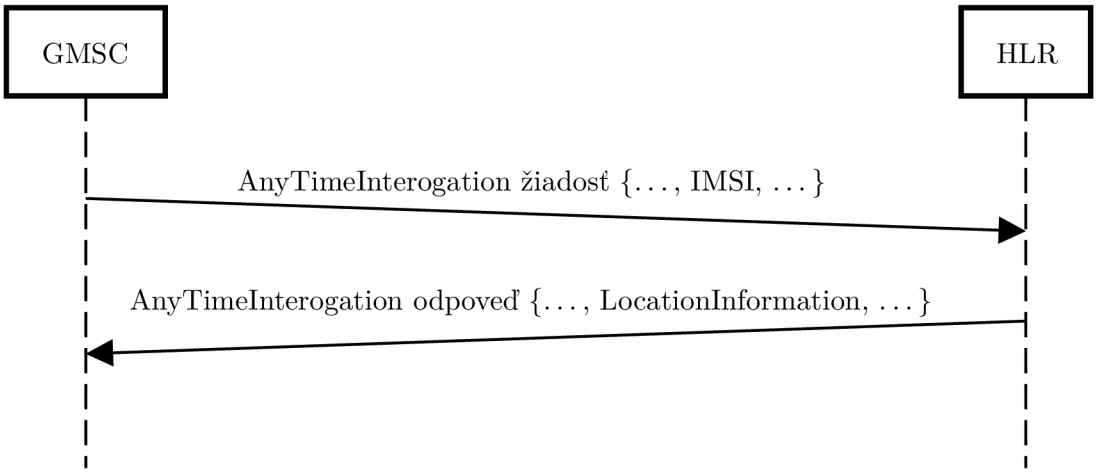
Location retrieval

Location retrieval predstavuje najčastejší prípad zneužitia IMSI. Pri tomto podvode sa útočník snaží zistiť geografickú polohu užívateľa v sieti. K dosiahnutiu tohoto cieľa môže útočník zneužiť napríklad službu **Provide Subscriber Info** (PSI), nakoľko slúži práve na tento účel. Podľa špecifikácie protokolu MAP [2] sa služba PSI používa na vyžiadanie si informácií (napr. stav užívateľa či poloha) z VLR. Toto využitie je znázornené na obrázku 2.8.



Obr. 2.8: Využitie služby PSI na získanie polohy užívateľa.

Okrem služby PSI možno fakticky totožným spôsobom zneužiť aj službu **Any Time Interrogation** (**ATI**). Tá má presne rovnaký účel aj zoznam parametrov ako PSI, jediným rozdielom je, že pomocou nej komunikujú iné uzly SS7 siete. Obrázok 2.9 znázorňuje tento prípad.

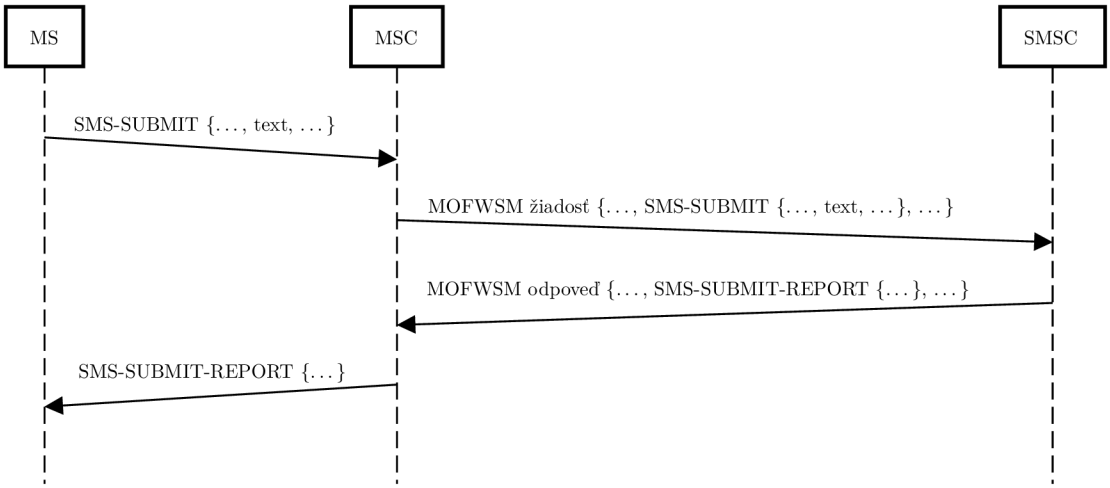


Obr. 2.9: Využitie služby ATI na získanie polohy užívateľa.

SMS fraud

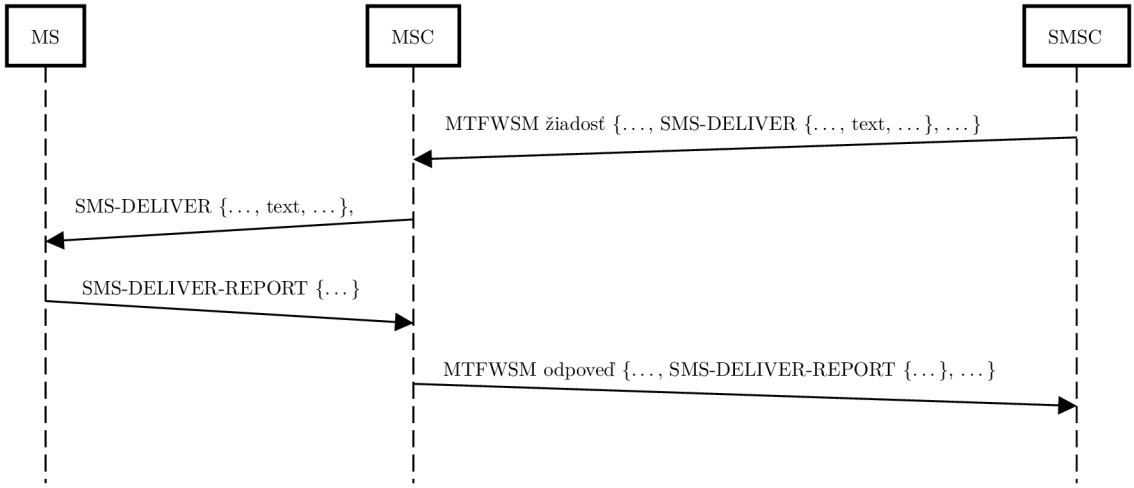
SMS fraud je označenie takého zneužitia služieb SS7 siete, ktoré slúži na distribúciu podvodných, nevyžiadaných alebo spamových textových spáv v mobilnej sieti. Príkladom služieb, ktoré možné k tomuto účelu zneužiť sú **Mobile Originated Forward Short Message** (**MOFWSM**) a **Mobile Terminated Forward Short Message** (**MTFWSM**).

Služba MOFWSM je využívaná pri doručovaní správ do SMS centra (SMSC). Táto procedúra je znázornená na diagrame na obrázku 2.10. V prípade, že sa útočníkovi podarí získať prístup do siete, môže sa vydávať za obsluhujúce MSC a pomocou správ MOFWSM odosielať textové správy (SM).



Obr. 2.10: Využitie služby MOFWSM na získanie polohy užívateľa.

Diagram na obrázku 2.11 zase znázorňuje procedúru doručovania SM z SMSC do adresátovej MS. V tejto variante zase útočníkovi postačí, aby sa vydával za SMSC a zneužil službu MTFWSM.



Obr. 2.11: Využitie služby MTFWSM na získanie polohy užívateľa.

Kapitola 3

Honeypoty

Nasledujúca kapitola detailne pojednáva o nástrojoch či počítačových systémoch, ktoré sa v odbornej terminológii prezývajú *honeypoty*. Jej náplňou je ozrejmenie samotného významu pojmu *honeypot*. Ďalej sú tu vysvetlené motivácie k nasadzovaniu týchto nástrojov do počítačových sietí a výhody aj nevýhody spojené s týmto nasadením. Čitateľ je zoznámený s kritériami, podľa ktorých je možné tieto nástroje deliť do rôznych kategórií, ktorých charakteristika je taktiež náplňou tejto kapitoly.

3.1 Definícia honeypotu

Slovo honeypot má v angličtine viacero významov. Podľa *Collins dictionary*[1] môže tento termín označovať nádobu s medom, alebo niečo, čo púta pozornosť veľkého množstva ľudí. A práve druhý menovaný význam sa najviac blíži významu, s ktorým je termín honeypot používaný v oblasti počítačových sietí a ich bezpečnosti.

L. Spitzner vo svojej knihe [23] opisuje honeypot ako jedinečný technologický systém, ktorý je špeciálne navrhnutý s účelom byť skúšaný, napádaný či kompromitovaný. Takýto systém potom poskytuje bezprecedentnú možnosť ofenzívy voči útočníkom a to napríklad tým, že umožňuje zbierať užitočné informácie o útočníkoch, ako sú ich motívy a taktika.

Alternatívna definícia z knihy *Virtual Honeypots* [10] znie, že honeypot je dôkladne monitorovaný zdroj informačného systému, ktorého hodnota spočíva v neautorizovanom či nedovolenom použití tohto zdroja.

3.2 Výhody a nevýhody honeypotov

Nasadenie honeypotov predstavuje menej tradičné bezpečnostné opatrenie v sieti. Na rozdiel od iných bezpečnostných prvkov, ktoré sú často zamerané na riešenie špecifického bezpečnostného problému, honeypoty cielia na širšie a obecnejšie spektrum problémov [23] a je preto nevyhnutné byť si vedomý ich silných a slabých stránok. Medzi základne výhody a motivácie pre nasadenie honeypotov radíme nasledovné:

- honeypoty poskytujú detailné informácie o činnosti útočníka, ktorých analýzou je možné zistiť jeho motiváciu a stratégiu útoku,
- honeypoty sú obecne nenáročné na výpočtové zdroje, či sieťovú infraštruktúru,

[1] www.collinsdictionary.com

- honeypoty sú vo svojej podstate veľmi jednoduché systémy, bez žiadnych zložitých algoritmov, či konfigurovania pravidiel a politík, kde sa dá ľahko urobiť chyba,
- honeypoty veľmi rýchlo a často demonštrujú svoju pridanú hodnotu a to pri každom jednom napadnutí. To je v kontraste napríklad s firewallom, kde čím lepšiu ochranu firewall poskytuje, tým menší je počet bezpečnostných incidentov, čo môže znižovať motiváciu manažmentu v budúcnosti investovať do takýchto bezpečnostných riešení.

Pozornému čitateľovi sú z hore uvedených výhod okamžite zrejmé aj niektoré nevýhody a riziká používania honeypotov. K týmto rizikám a nevýhodám radíme hlavne nasledovné:

- honeypoty majú veľmi obmedzenú pôsobnosť. Pokiaľ útočník napadne aj ostatné systémy v sieti, alebo sa cielene honeypotu vyhne, je hodnota takéhoto systému minimálna až žiadna,
- honeypoty (obzvlášť komerčné verzie) môžu vykazovať určité charakteristiky a správanie, ktoré umožňujú odhaliť pravú identitu honeypotu a útočníkovi potom naň stačí jednoducho neútočiť,
- v závislosti na komplexnosti a spôsobe fungovania môže honeypot predstavovať riziko, že po jeho úspešnom napadnutí ho útočník zneužije na ďalšie útoky.

3.3 Delenie honeypotov podľa miery interakcie

N. Provos [10] prezentuje viacero aspektov, podľa ktorých možno honeypoty deliť. Jedno zo základných delení, ktoré vo svojej knihe uvádza aj L. Spitzner [23], je delenie honeypotov podľa miery ich interakcie, a to na nasledujúce dve kategórie:

- Honeypot **s vysokou mierou interakcie** je regulárny počítačový systém, ktorý však neplní jemu typickú úlohu v sieti. Takýto systém nemá žiadnych užívateľov a nemal by generovať žiadnu, respektíve len minimálnu sieťovú prevádzku. Všetky tieto vlastnosti uľahčujú detegovanie útoku, nakoľko akúkoľvek interakciu s týmto typom honeypotu možno automaticky považovať za podozrivú.

  Pre úplný obraz je nutné zmieniť aj výhody a úskalia takéhoto riešenia. Pozitívne, resp. negatívne vlastnosti spojené s týmto typom honeypotu sú nasledovné:

  - ťažko odhaliteľný, nakoľko v princípe nie je rozdiel medzi honeypotom a regulárnym systémom,
  - dokáže zaznamenať veľké množstvo podrobných detailov o aktivite útočníka,
  - predstavuje vyššie riziko z pohľadu možného zneužitia,
  - komplikovanejšie nasadzovanie a správa.

- Honeypot **s nízkou mierou interakcie** implementuje iba podmnožinu služieb systému, za ktorý sa vydáva a miera interakcie je preto limitovaná. Pri tomto type honeypotov sa uplatňuje princíp čo najmenšej možnej miery interakcie postačujúcej na oklamanie útočníka.

  Tento typ honeypotu má nasledovné pozitívne, resp. negatívne vlastnosti:

  - rýchle a jednoduché nasadenie,

  - väčšinou dokáže zaznamenať iba štatistické a kvantitatívne informácie o aktivite útočníka,
  - relatívne nízke riziko z pohľadu možného zneužitia.

Okrem vyššie uvedených dvoch kategórií sa v niektorej literatúre [18] možno stretnúť s kategóriou **rýdzich** honeypotov (anglicky *pure honeypots*). Táto skupina predstavuje inštancie reálnych systémov, ktoré sú ako celok nasadené ako návnada pre útočníka. Monitorovanie aktivity prebieha iba monitorovaním sieťovej komunikácie. V praxi je takýto typ honeypotov užitočný najmä v prípade, kedy je neobyčajne dôležité, aby honeypot zostal neodhalený.

3.4 Delenie honeypotov podľa smeru interakcie

Smer interakcie je ďalší zo spôsobov delenia honeypotov. N. Provos [10] predstavuje vo svojej knihe alternatívny smer interakcie a tento spôsob delenia preto rozdeľuje honeypoty do dvoch kategórií:

- **Serverové** honeypoty (anglicky *server-side honeypots*), ktoré predstavujú tradičný koncept honeypotov. Podľa pôvodného konceptu totiž honeypot simuluje reálny systém (server) a pasívne čaká na napadnutie. Následne útočníkovi poskytne návnadu a zároveň zaznamená detaily tejto interakcie, ktoré sú potrebné pre budúcu analýzu útoku.
- **Klientské** honeypoty (anglicky *client-side honeypots*) interagujú v opačnom smere, honeypot simuluje chovanie klienta a sám iniciuje interakciu so serverom. Z analýzy prijatých odpovedí je potom možné zistiť, či bol klient-honeypot terčom nejakého útoku, prípadne ďalšie detaily takéhoto útoku.

Kapitola 4

Návrh SS7 honeypotu

Obsahom tejto kapitoly je popis návrhu implementovaného SS7 honeypotu. V tejto časti práce budú predstavené základné požiadavky kladené na výsledný produkt, ktoré boli zadané zadávateľom tejto práce – spoločnosťou *Mavenir*. Ďalej je v kapitole uvedený detailný popis spracovávania prijatých správ a obsah honeypotom generovaných odpovedí. V neposlednom rade je tu prezentovaný zamýšľaný spôsob nasadzovania honeypotu do SS7 sietí a iné dôležité vlastnosti výsledného riešenia.

4.1 Špecifikácia požiadaviek

Samotnej implementácii predchádzalo viacero konzultácii s odborným konzultantom, z ktorých vyplynuli očakávania od budúcej implementácie honeypotu. Z diskusie vzišli nasledovné požiadavky:

- honeypot bude schopný reagovať na nasledovné MAP správy – SIMSI, SRI, PSI, ATI, MOFWSM, MTFWSM,
- honeypot bude implementovaný nad modulom *bafomet* (pozri 5.1),
- vybrané parametre prijatých správ a odpovedí generovaných honeypotom budú zaznamenávané do CSV súboru na účely analýzy sieťovej prevádzky,
- honeypot bude kompletne bezstavový, nebude ukladať žiadne perzistentné dáta, kontexty, ...,
- honeypot bude mať priradenú jednu či viacero GT adries, ale jeho prítomnosť v sieti nebude nijakým spôsobom propagovaná – bude čakať, kým bude objavený útočníkom (napríklad pomocou GT skeneru),
- okrem MAP odpovede bude honeypot schopný reagovať aj správami TCAP Abort a TCAP Error, či prípadne nereagovať vôbec. To, akým spôsobom bude honeypot reagovať, bude rozhodnuté podľa dopredu nakonfigurovaných rozsahov GT adries.

4.2 Zamýšlané spracovávanie prichádzajúcich správ

Na nasledujúcich riadkoch bude detailne predstavený spôsob vytvárania odpovedí na podporované MAP správy. Hneď úvodom však treba poznamenať, že samotný štandard špecifikuje

ohromné množstvo voliteľných parametrov, ktoré môžu odpovede na jednotlivé MAP správy obsahovať. Pre účely honeypotu je však väčšina z nich nepodstatná. Voliteľné parametre honeypotom generovaných odpovedí boli zvolené po analýze reálnej sieťovej prevádzky. Záznamy tejto prevádzky boli poskytnuté zákazníkmi spoločnosti *Mavenir*.

4.2.1 Spracovávanie Send IMSI (SIMSI) správ

Štruktúra správy SIMSI (pozri tabuľky 4.1 a 4.2) je najjednoduchšia zo všetkých podporovaných správ. Žiadosť obsahuje iba jeden povinný parameter a to parameter *msisdn*. Odpoveď na túto správu zase obsahuje jediný povinný parameter a to parameter *imsi*, ktoré prislúcha *msisdn* zo žiadosti.

Parameter	Povinnosť	Popis
msisdn	M	MSISDN užívateľa, ktorého IMSI chceme zistiť.

Tabuľka 4.1: Čiastočná štruktúra SIMSI žiadosti.

Parameter	Povinnosť	Popis
imsi	M	Žiadané IMSI užívateľa.

Tabuľka 4.2: Čiastočná štruktúra SIMSI odpovede.

Pri vytváraní odpovede na túto správu bude do parametru *imsi* priradená náhodne vybraná hodnota z predom definovaného rozsahu, prípadne vygenerovaná náhodná hodnota podľa predom definovaného regulárneho výrazu.

4.2.2 Spracovávanie Send Routing Information (SRI) správ

Štruktúra správy SRI (pozri tabuľky 4.3 a 4.4) je zložitejšia, nakoľko obsahuje viacero voliteľných parametrov. Žiadosť povinne obsahuje, podobne ako v prípade SIMSI, parameter *msisdn* a niekoľko, pre naše účely nepodstatných, voliteľných parametrov.

Parameter	Povinnosť	Popis
msisdn	M	MSISDN užívateľa, pre ktorého chceme zistiť informácie nevyhnutné pre smerovanie hovoru.

Tabuľka 4.3: Čiastočná štruktúra SRI žiadosti.

Parameter	Povinnosť	Popis
imsi	M	Žiadané IMSI užívateľa.
routingInfo.roamingNumber	M	Číslo pridelené MS, pokiaľ je roamovaná mimo domácu sieť. Je nevyhnutné pre správne smerovanie hovorov.
subscriberInfo.locationInformation.vlr-number	O	Identifikácia VLR poskytujúceho dopytované informácie.

Tabuľka 4.4: Čiastočná štruktúra SRI odpovede.

V odpovedi sa bude opäť povinne nachádzať prislúchajúce IMSI v parametri *imsi* a navyše aj jeden z parametrov – *roamingNumber*, resp. *forwardingData*. Z analýzy reálnej prevádzky vyplynulo, že typicky je v odpovediach používaný parameter *roamingNumber*. Odpoveď ešte typicky obsahuje voliteľnú štruktúru *subscriberInfo.locationInformation* s nepovinným parametrom *vlr-number*.

Hodnoty všetkých parametrov budú náhodne vybrané z predom definovaného rozsahu, alebo vygenerované z predom definovaného regulárneho výrazu.

4.2.3 Spracovávanie Provide Subscriber Info (PSI) správ

MAP žiadosť PSI (pozri tabuľka 4.5) obsahuje jediný povinný parameter a to IMSI užívateľa, o ktorom chceme od siete dostať informácie. Táto žiadosť ďalej obsahuje štruktúru *requestedInfo*, v ktorej je možné bližšie špecifikovať, ktoré údaje nás konkrétne zaujímajú. Bolo vypozorované, že táto štruktúra typicky obsahuje parametre *imei*, *locationInformation*, *subscriberState* a *currentLocation*. Z tohto dôvodu to budú práve tieto štyri parametre, ktoré dokáže honeypot spracovať a pripraviť na ne adekvátnu odpoveď.

Parameter	Povinnosť	Popis
imsi	M	IMSI užívateľa, o ktorom chceme získať informácie.
requestedInfo.imei	O	Vyžiadanie si IMEI MS užívateľa.
requestedInfo.locationInformation	O	Vyžiadanie si informácií o polohe užívateľa.
requestedInfo.subscriberState	O	Vyžiadanie si informácií o stave užívateľa (nedostupnosť, prebiehajúci hovor, ...).
requestedInfo.currentLocation	O	Vynútenie aktualizácie polohy užívateľa sieťou.

Tabuľka 4.5: Čiastočná štruktúra PSI žiadosti.

Parameter	Povinnosť	Popis
subscriberInfo.imei	O	IMEI MS užívateľa.
subscriberInfo.locationInformation. ageOfLocationInformation	O	Čas v minútach od poslednej aktualizácie polohy užívateľa.
subscriberInfo.locationInformation. vlr-number	O	Identifikácia VLR poskytujúceho dopytované informácie.
subscriberInfo.locationInformation. cellGlobalIdOrServiceAreaId	O	Jedinečný globálny zložený identifikátor, ktorý identifikuje LA a samotnú bunku, v ktorej je užívateľ registrovaný.
subscriberInfo.subscriberState	O	Aktuálny stav užívateľa (nedostupnosť, prebiehajúci hovor, ...).
subscriberInfo. currentLocationRetrieved	O	Indikácia, že aktualizácia polohy bola úspešne vynútená.

Tabuľka 4.6: Čiastočná štruktúra PSI odpovede.

PSI odpoveď (pozri tabuľka 4.6) bude obsahovať dopytované informácie o užívateľovi siete. Za pozornosť predovšetkým stojí štruktúra *locationInformation*. Tá opäť môže obsahovať veľké množstvo voliteľných parametrov, ale v typickej prevádzke táto štruktúra obsahuje nasledovné parametre:

- *ageOfLocationInformation*, ktorého hodnota bude náhodne vybraná z intervalu $\langle 0, 59 \rangle$,
- *vlr-number*, ktorého hodnota bude náhodne vybraná z predom nakonfigurovaného rozsahu, alebo vygenerovaná z regulárneho výrazu,
- *cellGlobalIdOrServiceAreaId*, ktorého hodnota bude vytvorená z nakonfigurovanej hodnoty MCC, MNC, LAC a CID.

Hodnota parametru *imei* bude náhodne vybraná z nakonfigurovaného rozsahu, alebo vygenerovaná z regulárneho výrazu. Parameter *subcriberState* v honeypotom generovanej odpovedi bude vždy nastavený na hodnotu *assumedIdle*, čo indikuje, že na MS užívateľa neprebieha žiadna aktivita. Príznak *currentLocationRetrieved* bude vždy prítomný, pokiaľ bolo v žiadosti vynútenie aktualizácie polohy.

4.2.4 Spracovávanie Any Time Interrogation (ATI) správ

Všetky parametre žiadosti a odpovede MAP ATI, podstatné z hľadiska fungovania honeypotu, sú zaznamenané v tabuľkách 4.7 a 4.8. Pozornému čitateľovi isto neunikne, že ich obsah je prakticky totožný s obsahom analogických tabuliek patriacich k PSI. Je to z toho dôvodu, že štruktúra aj funkcia oboch správ je veľmi príbuzná. Implementácia spracovania ATI správ bude preto totožná so spracovaním PSI správ.

Parameter	Povinnosť	Popis
imsi	M	IMSI užívateľa, o ktorom chceme získať informácie.
requestedInfo.imei	O	Vyžiadanie si IMEI MS užívateľa.
requestedInfo.locationInformation	O	Vyžiadanie si informácií o polohe užívateľa.
requestedInfo.subscriberState	O	Vyžiadanie si informácií o stave užívateľa (nedostupnosť, prebiehajúci hovor, ...).
requestedInfo.currentLocation	O	Vynútenie aktualizácie polohy užívateľa sieťou.

Tabuľka 4.7: Čiastočná štruktúra ATI žiadosti.

Parameter	Povinnosť	Popis
subscriberInfo.imei	O	IMEI MS užívateľa.
subscriberInfo.locationInformation.ageOfLocationInformation	O	Čas v minútach od poslednej aktualizácie polohy užívateľa.
subscriberInfo.locationInformation.vlr-number	O	Identifikácia VLR poskytujúceho dopytované informácie.
subscriberInfo.locationInformation.cellGlobalIdOrServiceAreaId	O	Jedinečný globálny zložený identifikátor, ktorý identifikuje LA a samotnú bunku, v ktorej je užívateľ registrovaný.
subscriberInfo.subscriberState	O	Aktuálny stav užívateľa (nedostupnosť, prebiehajúci hovor, ...).
subscriberInfo.currentLocationRetrieved	O	Indikácia, že aktualizácia polohy bola úspešne vynútená.

Tabuľka 4.8: Čiastočná štruktúra ATI odpovede.

4.2.5 Spracovávanie Mobile Originated Forward Short Message (MO-FWSM) správ

Žiadosť MOFWSM (pozri tabuľka 4.9) obsahuje tri povinné parametre – *sm-RP-DA*, *sm-RP-OA* a *sm-RP-UI*. Prvé dva zmienené parametre identifikujú odosielateľa, resp. adresáta SM.

Parameter *sm-RP-UI* obsahuje samotný obsah SM. Tento obsah je uložený vo formáte TPDU. Implementovaný honeypot bude podporovať dekódovanie TPDU typu **SMS-SUBMIT** a **SMS-DELIVER** z tela žiadosti MOFWSM.

Parameter	Povinnosť	Popis
sm-RP-DA	M	Cieľová adresa pre doručenie textovej správy (SM). Toto pole môže (mimo iné) obsahovať adresu MS (IMSI), alebo adresu SMSC.
sm-RP-OA	M	Zdrojová adresa textovej správy (SM). Toto pole môže napríklad obsahovať adresu MS (MSISDN), alebo adresu SMSC.
sm-RP-UI	M	Obsah MOFWSM žiadosti uložený vo formáte TPDU. Typicky **SMS-SUBMIT**.

Tabuľka 4.9: Čiastočná štruktúra MOFWSM žiadosti.

Parameter	Povinnosť	Popis
sm-RP-UI	O	Obsah MOFWSM odpovede uložený vo formáte TPDU. Typicky **SMS-SUBMIT-REPORT**.

Tabuľka 4.10: Čiastočná štruktúra MOFWSM odpovede.

Telo odpovede MOFWSM (pozri tabuľka 4.10) obsahuje voliteľný parameter *sm-RP-UI*. Z analýzy reálnej prevádzky vyplynulo, že tento parameter typicky býva v odpovediach prítomný, a že obsahuje potvrdenie o spracovaní SM-RP-UI zo žiadosti. Honeypot v ním ge-

nerovaných odpovediach preto bude podporovať TPDU typu **SMS-SUBMIT-REPORT** a **SMS-DELIVER-REPORT**.

4.2.6 Spracovávanie Mobile Terminated Forward Short Message (MTFWSM) správ

Žiadosť MTFWSM (pozri tabuľka 4.11) je štrukturálne veľmi podobná MOFWSM. Obsahuje rovnaké povinné parametre (*sm-RP-OA*, *sm-RP-DA* a *sm-RP-UI*), odlišuje sa iba inou sadou voliteľných parametrov. Tieto sú rovnako ako v prípade MOFWSM pre činnosť honeypotu nepodstatné.

Parameter	Povinnosť	Popis
sm-RP-DA	M	Cieľová adresa pre doručenie textovej správy (SM). Toto pole môže (mimo iné) obsahovať adresu MS (IMSI), alebo adresu SMSC.
sm-RP-OA	M	Zdrojová adresa textovej správy (SM). Toto pole môže napríklad obsahovať adresu MS (MSISDN), alebo adresu SMSC.
sm-RP-UI	M	Obsah MTFWSM žiadosti uložený vo formáte TPDU. Typicky **SMS-DELIVER**.

Tabuľka 4.11: Čiastočná štruktúra MTFWSM žiadosti.

Parameter	Povinnosť	Popis
sm-RP-UI	O	Obsah MTFWSM odpovede uložený vo formáte TPDU. Typicky **SMS-DELIVER-REPORT**.

Tabuľka 4.12: Čiastočná štruktúra MTFWSM odpovede.

Honeypotom generovaná odpoveď (pozri tabuľka 4.12) na túto správu bude zložením totožná s odpoveďou na MOFWSM. Bude obsahovať jeden voliteľný parameter *sm-RP-UI*, nesúci TPDU potvrdzujúce spracovanie SM-RP-UI zo *sm-RP-UI* parametru žiadosti.

4.2.7 Štruktúry SMS-SUBMIT-REPORT a SMS-DELIVER-REPORT

Ako už bolo spomenuté vyššie, štruktúry **SMS-SUBMIT-REPORT** a **SMS-DELIVER-REPORT** sú dva typy TPDU, ktoré sa používajú v odpovediach a slúžia na potvrdenie spracovania TPDU zo žiadostí. Rovnako ako vyššie uvedené MAP správy aj tieto štruktúry môžu voliteľne obsahovať rozličné parametre. Pre účely honeypotu, tzn. prosté potvrdzovanie aktivity útočníka, ale stačí prostá prítomnosť prázdnych štruktúr v MAP odpovedi.

SMS-SUBMIT-REPORT

Táto štruktúra sa bežne nachádza v MAP odpovedi, ktorá potvrdzuje prijatie SM SMS centrom (SMSC). Z tohto dôvodu táto štruktúra obsahuje jediný povinný parameter a to *serviceCentreTimeStamp*, ktorý udáva čas prijatia SMS centrom. V honeypotom generovaných odpovediach bude táto hodnota rovná aktuálnemu času.

SMS-DELIVER-REPORT

Podobne aj obsah tejto štruktúry tvorí viacero povinných indikátorov prítomnosti voliteľných parametrov. Štruktúra býva prítomná v potvrdení, že správa doručovaná SMSC bola úspešne doručená. Okrem už uvedeného, štruktúra neobsahuje žiadne povinné parametre, či parametre významné pre činnosť honeypotu.

4.2.8 Negatívne odpovede TCAP Abort a TCAP Error odpovedí

Jedna z požiadaviek na implementáciu bola, aby honeypot okrem imitovania úspešnej interakcie pomocou MAP správ dokázal na vybrané GT adresy reagovať aj negatívnymi odpoveďami. Ide o odpovede TCAP Abort a TCAP Error, ktorých obsah a tvorba bude predstavený v tejto častí.

TCAP Abort

Odpoveď TCAP Abort je zjednodušené označenie takej TCAP správy, ktorá obsahuje tzv. *abort* primitívum. Takáto správa býva zasielaná v prípade výskytu chyby a spôsobuje zatvorenie celého TCAP dialógu.

Zmienené *abort* primitívum bude obsahovať dva parametre, ID dialógu (rovnako ako ostatné primitíva) a dôvod prerušenia komunikácie (parameter *p-abortCause*). Hodnota tohto parametru bude predom definovaná v konfiguračnom súbore honeypotu.

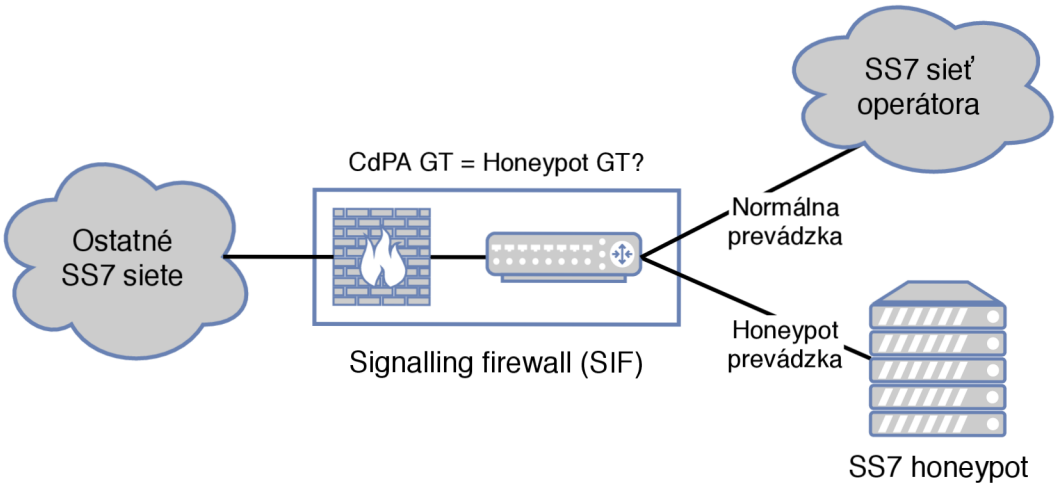
TCAP Error

Odpoveď TCAP Error je označenie MAP správy, ktorá je prenášaná v tzv. *returnError* komponente. Táto MAP správa obsahuje jediný parameter *errorCode* označujúci typ vyskytnutej chyby. Podobne ako v predchádzajúcom prípade aj hodnota tohto parametru bude predom definovaná v konfigurácii honeypotu.

4.3 Integrácia do siete operátora

Navrhovaný honeypot nebude zákazníkom spoločnosti *Mavenir* ponúkaný ako samostatný produkt, ale bude súčasťou komplexnejšieho nasadenia viacerých produktov spoločnosti. Jedným z týchto ďalších produktov bude aj tzv. Signalling firewall (SIF). Jedná sa o komplexný nástroj umožňujúci pokročilé spracovávanie SS7 prevádzky, jej filtrovanie, či smerovanie.

Ako už bolo spomenuté, honeypotu bude pridelená unikátna GT adresa. Toto priradenie bude realizované práve v SIF, ktorý bude mať s honeypotom nadviazané MTP-3/M3UA spojenie a všetka prevádzka smerujúca na GT honeypotu bude presmerovávaná do tohto spojenia. Toto zapojenie je znázornené na obrázku 4.1.



Obr. 4.1: Zapojenie honeypotu do SS7 siete.

Kapitola 5

Implementácia SS7 honeypotu

Na základe špecifikovaných požiadaviek a návrhu, ktoré boli prezentované v predchádzajúcej kapitole, bol zamýšľaný honeypot implementovaný. Implementácia bola realizovaná v jazyku Python s využitím princípov objektovo orientovaného programovania.

Náplňou tejto kapitoly je predstavenie technológií použitých pri implementácii a opis celkovej štruktúry implementácie. Tento opis je nasledovaný popisom úloh a zodpovedností jednotlivých tried implementácie a opisom konfigurácie a výstupu honeypotu. Záver kapitoly potom pojednáva o špecifikách prostredia, do ktorého bude honeypot typicky nasadzovaný.

5.1 Technológie použité pri implementácii

Na nasledujúcich riadkoch sa nachádza stručný prehľad použitých technológii. Všetky použité technológie boli zvolené po konzultácii s odborným konzultantom.

Python

Celá implementácia honeypotu je realizovaná v jazyku Python. Hlavnou motiváciou pre použitie tohto jazyka bola existencia programového vybavenia vo forme modulu, ktorý implementuje celý SIGTRAN sieťový zásobník. Tento modul má názov *bafomet* a jeho podrobnejší opis bude nasledovať neskôr.

Jazyk Python je interpretovaný jazyk, ktorý podporuje viacero programovacích paradigiem a to vrátane procedurálneho, objektovo orientovaného, či funkcionálneho programovania. Je to jazyk silno a dynamicky typovaný, s automatickou správou pamäte.

Bafomet

Ako už bolo spomenuté, modul *bafomet* implementuje (mimo iné) celý SIGTRAN sieťový zásobník. Pomocou volania jednej metódy umožňuje jednoduché nadviazanie spojenia medzi uzlami SS7 siete. Podporuje spojenie či už pomocou protokolu M2PA, ako aj protokolu M3UA.

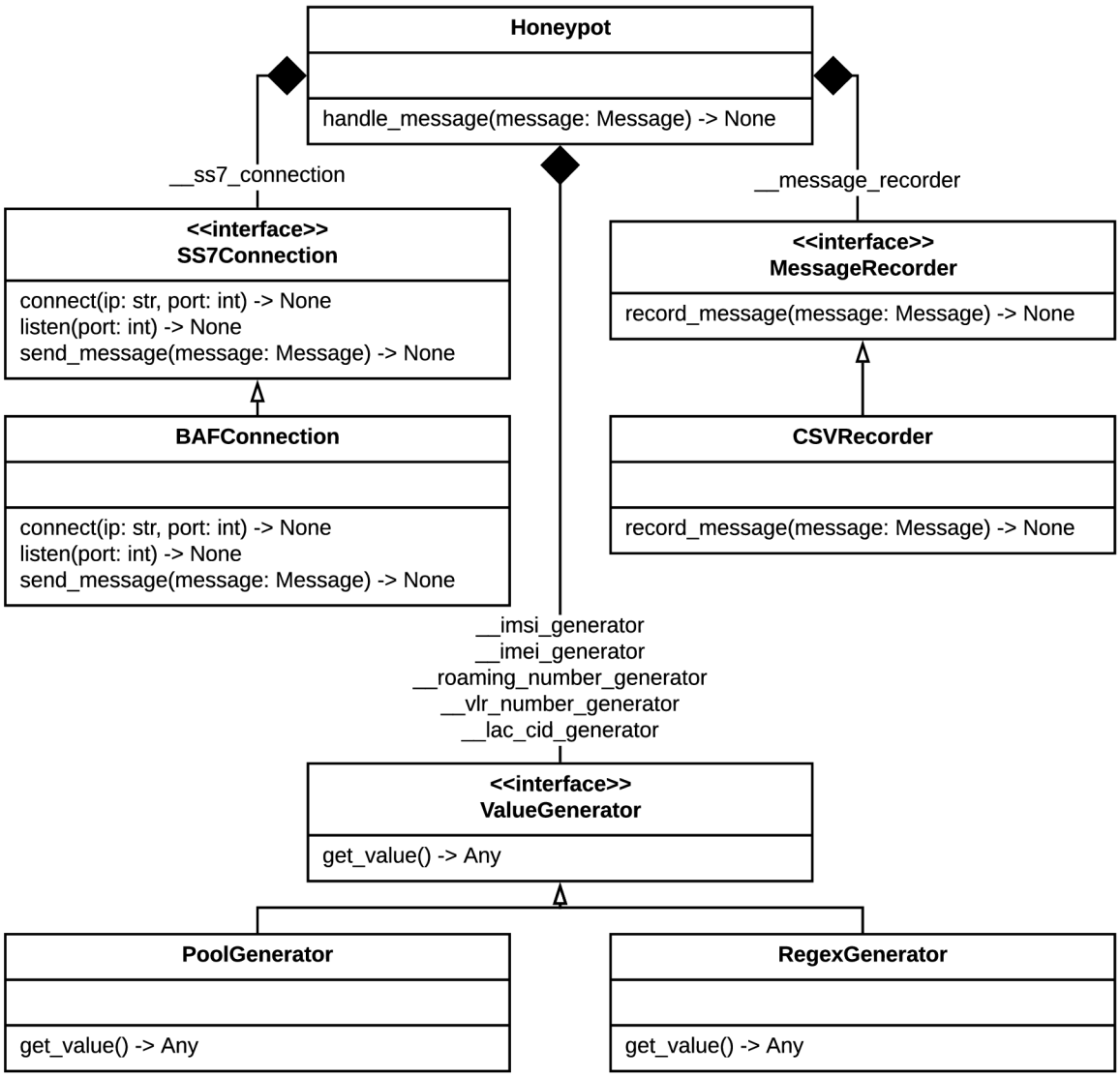
Formát CSV

Formát Comma Separated Values (CSV) je jednoduchý textový protokol, ktorý umožňuje ukladať tabuľkové dáta. Je definovaný v štandarde RFC 4180 [22]. Každý záznam sa nachádza na novom riadku, ktoré sú oddelené pomocou `CRLF`. Jednotlivé položky jedného

záznamu sú oddeľované znakom `,` a voliteľne môžu byť jednotlivé položky uzatvorené v dvojitých úvodzovkách `""`.

5.2 Popis implementácie

Implementácia honeypotu je rozdelená do niekoľkých tried. Tieto triedy, závislosti medzi nimi a ich verejné rozhrania sú znázornené na diagrame tried na obrázku 5.1. Nižšie nasleduje detailný opis jednotlivých tried, ich činnosti a zodpovednosti.



Obr. 5.1: Diagram tried výslednej implementácie.

5.2.1 Trieda `Honeypot`

V triede `Honeypot` je sústredená všetka logika spracovávania a reagovania na prichádzajúce správy, ako aj spracovanie konfiguračného súboru.

Pri inštanciovaní tejto triedy je ako jediný parameter konštruktora predaná cesta ku konfiguračnému súboru a v rámci inštancializácie je tento súbor spracovaný. V prípade,

že konfiguračný súbor nie je syntakticky správny, alebo v ňom nie sú definované všetky povinné parametre, je inštancializácia prerušená vyvolaním výnimky.

Následne sú na základe konfiguračného súboru inicializované všetky parametre potrebné na obsluhu prichádzajúcich správ a sú vytvorené inštancie všetkých generátorov hodnôt (pozri 5.2.3). Rovnako tak je vytvorená inštancia „zapisovača“ správ do CSV súboru (pozri 5.2.4) a „obsluhovača“ SS7 spojenia (pozri 5.2.2). V úplnom závere vytvárania inštancie triedy `Honeypot` je nadväzované spojenie podľa nakonfigurovaných parametrov.

Obsluha prichádzajúcich správ je potom vykonávaná asynchrónne a to volaním metódy `handle_message(message)` z triedy `BAFConnection`. Bližší opis spracovávania prichádzajúcich správ je opísaný v sekcii 5.3.

5.2.2 Rozhranie `SS7Connection` a jeho implementácia `BAFConnection`

Jednou z požiadaviek, ktoré stáli na začiatku implementácie bolo, že implementovaný honeypot bude využívať funkcionalitu modulu *bafomet*. Trieda `BAFConnection` zapuzdruje používanie tohoto modulu a tým ho oddeľuje od logiky samotného honeypotu implementovanej v triede `Honeypot`.

Toto oddelenie spĺňuje dva ciele. Prvým bola eliminácia závislosti obslužných rutín na module *bafomet*. V prípade, že by v budúcnosti bolo zmenené aplikačné rozhranie tohto modulu, alebo by ho bolo nutné úplne nahradiť, z implementačného hľadiska postačí vytvoriť novú triedu implementujúcu rozhranie `SS7Connection` a z pohľadu triedy `Honeypot` bude táto zmena úplne transparentná.

Druhým sledovaným cieľom bolo, že pri implementácii bola snaha o využívanie statického typovania. Vzhľadom na to, že v programovacom jazyku *Python* je uvádzanie typov premenných, inštančných premenných, typov parametrov funkcií a metód, či ich návratových hodnôt, dobrovoľné (a v starších verziách dokonca táto podpora nebola vôbec), tak ani modul *bafomet* neposkytuje staticky typované rozhranie. Zapuzdrenie tohto modulu do triedy so staticky typovaným rozhraním umožňuje sa s touto skutočnosťou vysporiadať a vo zvyšku implementácie statické typovanie bez obmedzení používať.

5.2.3 Rozhranie `ValueGenerator` a jeho implementácie `PoolGenerator` a `RegexGenerator`

Odpovede generované honeypotom obsahujú veľké množstvo parametrov, ktorých hodnoty boli vygenerované podľa dopredu generovaného kľúča. Každý takýto generátor v implementácii predstavuje inštanciu triedy, ktorá implementuje rozhranie `ValueGenerator` a teda obsahuje metódu `get_value()`. V implementácii sa nachádzajú dve triedy implementujúce toto rozhranie – `PoolGenerator` a `RegexGenerator`.

Trieda `PoolGenerator` generuje hodnoty z dopredu definovaného zoznamu platných hodnôt, alebo z dopredu definovaného intervalu. Hodnoty poskytované týmto typom generátoru nesledujú poradie, v akom boli definované v konfigurácii, ale ich voľba je náhodná.

Druhá zmienená trieda,`RegexGenerator`, zase implementuje mechanizmus, ktorý generuje reťazce podľa nakonfigurovaného regulárneho výrazu. Podobne ako v prípade `PoolGenerator`, aj v tomto prípade je generovanie náhodné, tzn. sled po sebe vygenerovaných reťazcov nepredstavuje či už lexikografické, alebo iné usporiadanie.

Oddelenie implementácie generátorov do samostatných tried, ktoré implementujú rozhranie `ValueGenerator`, dovoľuje jednoduché rozšírenie implementácie o honeypotu o nové mechanizmy generovania hodnôt v budúcnosti.

5.2.4 Rozhranie `MessageRecorder` a jeho implementácia `CSVRecorder`

Ďalšou požiadavkou na implementovaný honeypot bolo zaznamenávanie prichádzajúcich správ a honeypotom generovaných odpovedí do CSV súboru. Táto funkcionalita je implementovaná v triede `CSVRecorder`. Táto trieda implementuje rozhranie `MessageRecorder` a preto implementuje metódu `record_message(message)`. Po zavolaní tejto metódy je v cieľovom CSV súbore vytvorený nový záznam s vybranými parametrami správy predanej ako parameter pri jej volaní.

V rámci triedy `CSVRecorder` je taktiež implementovaný mechanizmus „otáčania“ výstupných CSV súborov. Namiesto ukladania všetkých záznamov do jediného súboru je do jedného súboru uložený (predom nakonfigurovaný) maximálny počet záznamov. Po dosiahnutí tohto limitu je súbor uzatvorený a ďalšie záznamy sú ukladané do novovytvoreného súboru.

Okrem limitu čo do počtu záznamov, je tu taktiež limit časový (taktiež konfigurovateľný), určujúci maximálnu dobu, po ktorú sú záznamy ukladané do jedného súboru. Zmena súboru nastane vždy pri dosiahnutí jedného z týchto limitov.

CSV súbory sú ukladané do nakonfigurovaného adresára, kde názov každého súboru tvorí prefix `honyepot_`, nasledovaný časovou pečiatkou vytvorenia súboru.

Oddelenie implementácie do samostatnej triedy opäť umožňuje jednoduché rozšírenie implementácie v budúcnosti. Tou môže byť zmena mechanizmu ukladania záznamov do CSV súboru, zmena zaznamenávaných parametrov, úplná zmena formátu, alebo cieľa ukladania záznamov (napríklad ukladanie do databáze, či odosielanie po sieti na ďalšie spracovanie). Pre dosiahnutie týchto cieľov postačí zmena implementácie v triede `CSVRecorder`, alebo vytvorenie úplne novej triedy implementujúcej rozhranie `MessageRecorder`. Obe tieto zmeny budú plne transparentné z pohľadu triedy `Honeypot`.

5.3 Popis činnosti honeypotu

Na sekvenčných diagramoch na obrázkoch 5.2 a 5.3 je znázornený princíp činnosti implementovaného honeypotu. Prichádzajúca správa je predaná honeypotu pomocou volania metódy `handle_message`, ktorá volá privátnu metódu `__handle_message`.

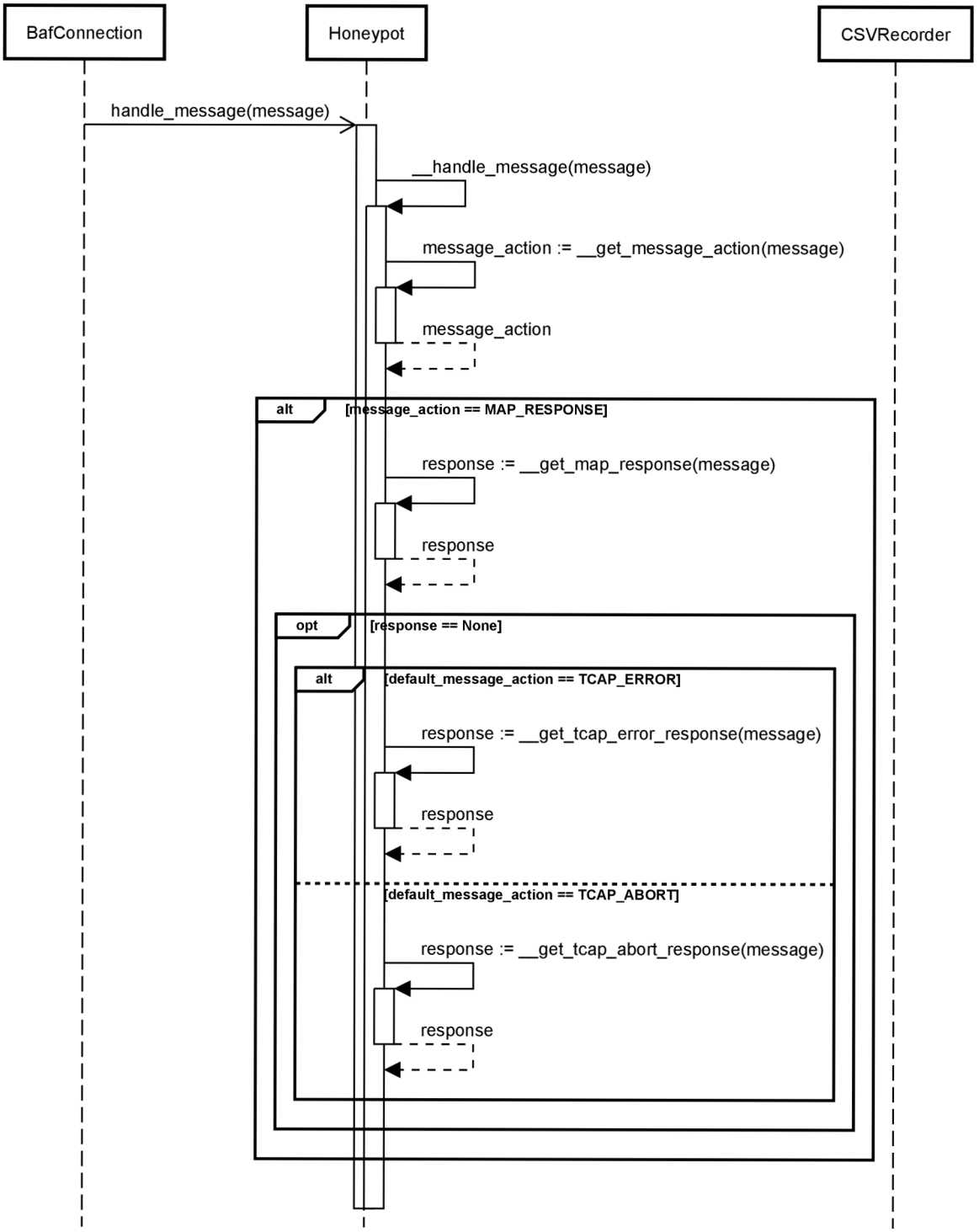
Ako prvé musí byť rozhodnuté, akým spôsobom má honeypot na prijatú správu reagovať. Spôsob reakcie získame pomocou volania pomocnej metódy `__get_message_action`. Návratová hodnota tejto metódy patrí do vymenovaného typu `MessageAction`.

Chod programu je ďalej vetvený podľa spôsobu reakcie honeypotu. Pomocou volania metód `__get_map_response`, `__get_tcap_error_response`, resp. `__get_tcap_abort_response` je pripravená adekvátna odpoveď na prijatú správu. V prípade, že prijatá správa má byť zahodená bez žiadnej odpovede, je spracovávanie tejto správy zastavené a hodnota premennej `response` zostáva rovná `None`. V prípade, že by mal honeypot reagovať MAP odpoveďou, ale prijatá správa nepatrí medzi podporované MAP správy, zareaguje honeypot dopredu nakonfigurovaným spôsobom.

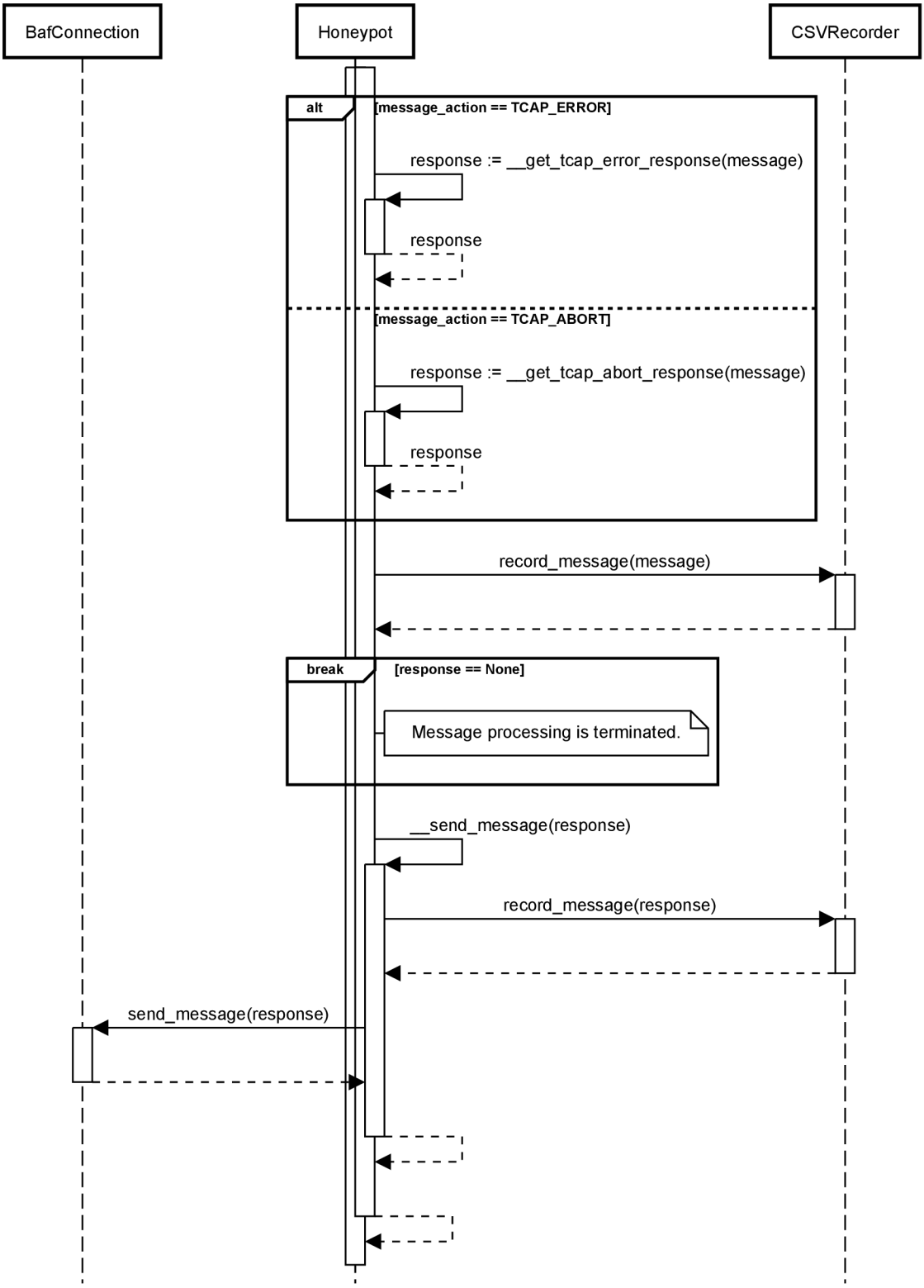
Následne sú vybrané parametre prijatej správy, spolu so spôsobom reakcie honeypotu, zaznamenané do CSV výstupu honeypotu. Zaznamenávanie správ do CSV súboru je implementované v triede `CSVRecorder` a je uskutočnené volaním jej metódy `record_message`.

V prípade, že hodnota premennej `response` je `None`, je ďalšie spracovávanie zastavené. Program opúšťa metódu `__handle_message` a tiež metódu `handle_message`.

Pripravená odpoveď je ďalej spracovaná metódou `__send_message`. V nej sú vybrané parametre pripravenej odpovede zaznamenané do CSV výstupu volaním metódy `record_message`. Následne je odpoveď odoslaná volaním metódy `send_message` triedy `BafConnection`.



Obr. 5.2: Sekvenčný diagram spracovania prichádzajúcej správy.



Obr. 5.3: Sekvenčný diagram spracovania prichádzajúcej správy (pokračovanie).

5.4 Konfigurácia

Konfigurácia honeypotu je realizovaná pomocou konfiguračného súboru, ktorého cesta je predaná ako jediný argument pri spúšťaní. Čo sa týka formátu tohoto súboru, nejedná sa o nejaký tradičný konfiguračný formát (YAML, XML, JSON, ...), ale ide o ďalší Python skript.

Tento spôsob konfigurácie bol zvolený z toho dôvodu, že takýto konfiguračný súbor má ďaleko bohatšie vyjadrovacie možnosti a umožňuje tzv. *reflexiu* – je možné znovu použiť už skôr definované parametre pri definovaní ďalších (príklad takéhoto využitia je ukázaný na výpise 1).

Parametre konfiguračného súboru

Na nasledovných riadkoch nasleduje zoznam parametrov konfiguračného súboru. Ku každému parametru je uvedený krátky popis a formát jeho zápisu v konfiguračnom súbore.

- `LISTEN` – hodnota tohto parametru predstavuje číslo SCTP portu, na ktorom honeypot počúva a čaká na nadviazanie spojenia. Formát zápisu je celé kladné číslo.
- `CONNECT` – dvojica IP adresa a SCTP port, na ktorý má honeypot vytvoriť SCTP spojenie. Formát zápisu je dvojica (IP adresa zapísaná ako reťazec a SCTP port zapísaný ako celé kladné číslo).
- `MCC` – MCC používaná pri generovaní identifikátora bunky (pozri 4.2.3 a 4.2.4). Formát zápisu ako celé číslo z intervalu $\langle 1, 999 \rangle$.
- `MNC` – MNC používaná pri generovaní identifikátora bunky (pozri 4.2.3 a 4.2.4). Formát zápisu ako celé číslo z intervalu $\langle 0, 999 \rangle$.
- `ABORT_CAUSE` – číselný kód chyby používaný pri generovaní TCAP Abort odpovedi (pozri 4.2.8). Formát zápisu ako celé kladné číslo.
- `ERROR_CODE` – číselný kód chyby používaný pri generovaní TCAP Error odpovedi (pozri 4.2.8). Formát zápisu ako celé kladné číslo.
- `CSV_DIR` – cesta k adresáru, do ktorého majú byť ukladané výstupné CSV súbory (pozri 5.5).
- `CSV_MAX_RECORDS` – maximálny počet záznamov v jednom CSV súbore (pozri 5.5). Formát kladné celé číslo.
- `CSV_TIMEOUT` – maximálna doba v sekundách, po ktorú je možné ukladať záznamy do jedného CSV súboru (pozri 5.5). Formát zápisu kladné desatinné číslo.
- `MESSAGE_ACTIONS` – spôsob reakcie honeypotu na prichádzajúcu správu. Zápis vo formáte slovníka {regulárny výraz pre CgPA GT: reakcia honeypotu}. Reakcia honeypotu môže byť jedna z nasledovných hodnôt: `MessageAction.MAP_RESPONSE`, `MessageAction.TCAP_ABORT`, `MessageAction.TCAP_ERROR` a `MessageAction.SILENT_DROP`.
- `DEFAULT_MESSAGE_ACTION` – východzí spôsob reakcie honeypotu. Prípustné hodnoty sú `MessageAction.TCAP_ABORT`, `MessageAction.TCAP_ERROR` a `MessageAction.SILENT_DROP`.

- `IMSI_GENERATOR` – inicializácia generátora hodnôt IMSI (pozri 4.2.1 a 4.2.2). Formát zápisu je buď regulárny výraz, alebo list hodnôt.
- `IMEI_GENERATOR` –inicializácia generátora hodnôt IMEI. Formát zápisu je buď regulárny výraz, alebo list hodnôt.
- `ROAMING_NUMBER_GENERATOR` – inicializácia generátora hodnôt *roaming number* (pozri 4.2.2). Formát zápisu je buď regulárny výraz, alebo list hodnôt.
- `VLR_NUMBER_GENERATOR` – inicializácia generátora hodnôt *VLR number* (pozri 4.2.3 a 4.2.4). Formát zápisu je buď regulárny výraz, alebo list hodnôt.
- `LAC_CID_GENERATOR` – dvojice (LAC a CID) používané identifikátora bunky (pozri 4.2.3 a 4.2.4). Formát zápisu ako list dvojíc kladných celých čísel.

Na výpise 1 je možné vidieť príklad obsahu konfiguračného súboru. Za zmienku stojí už zmienená reflexia na riadku 14. V regulárnom výraze sa časti `{MCC}` a `{MNC}` substituujú za hodnoty definované na riadkoch 2 a 3. Tento efekt by nebolo možné dosiahnuť v typických formátoch pre konfiguračné súbory.

```
1  CONNECT = ("127.0.0.1", 6666)
2  MCC = 420
3  MNC = 34
4  ERROR_CODE = 12
5  ABORT_CAUSE = 14
6  CSV_DIR = "/tmp/csv/"
7  CSV_MAX_RECORDS = 100
8  CSV_TIMEOUT = 3600
9  MESSAGE_ACTIONS = {
10     regex("420.+") : MessageAction.MAP_RESPONSE,
11     regex("421.+") : MessageAction.TCAP_ABORT,
12     regex(".+")    : MessageAction.TCAP_ERROR,
13     regex("420610.+"): MessageAction.SILENT_DROP,
14 }
15 DEFAULT_MESSAGE_ACTION = MessageAction.SILENT_DROP
16 IMSI_GENERATOR = regex(f"{MCC}{MNC}[0-9]{{10}}")
17 IMEI_GENERATOR = regex("[0-9]{6}")
18 ROAMING_NUMBER_GENERATOR = ['1.1.111111111']
19 VLR_NUMBER_GENERATOR = ['1.1.111111111']
20 LAC_CID_GENERATOR = [
21     (1835, '28088'),
22     ('45', 6)
23 ]
```

Výpis 1: Príklad obsahu konfiguračného súboru honeypotu.

5.5 CSV výstup

Zaznamenávanie aktivity honeypotu do CSV súboru bola ďalšia požiadavka kladená na implementáciu. Honeypot vytvára do výstupného súboru vždy jeden záznam pre prichá-

dzajúcu správu a v prípade vygenerovania odpovede aj správu odchádzajúcu. Tento výstup plní základnú úlohu honeypotu a to zber informácii o interakcii s honeypotom – teda zber informácií o potencionálnom útoku.

Každý jeden záznam prijatej/odoslanej správy obsahuje nasledovné parametre:

- čas prijatia/odoslania správy,
- typ vnútornej reprezentácie správy honeypotu (konkrétna trieda z dátového modelu),
- v prípade MAP správy, tzv. *operation code*, ktorý identifikuje konkrétny typ MAP správy (užitočný v prípade obdržania nepodporovaného typu MAP správy),
- v prípade prichádzajúcej správy, spôsob reakcie honeypotu (pozri 5.4),
- zdrojový a cieľový SPC,
- zdrojová a cieľová GT adresa (CgPA, CdPA),
- zdrojové a cieľové ID TCAP transakcie a
- ďalšie vybrané parametre z vrstvy TCAP (tzv. TCAP *component* a TCAP *application context*).

Okrem vyššie uvedených parametrov, v prípade MAP správ môžu byť prítomné nasledovné parametre:

- IMSI (prítomné pri SIMSI, SRI odpovediach a PSI, ATI žiadostiach),
- MSISDN (prítomné pri SIMSI, SRI žiadostiach) a
- SM-RP-UI (prítomné pri MOFWSM a MTFWSM žiadostiach aj odpovediach).

5.6 Nasadzovanie nástroja

Vzhľadom na charakter zariadení, na ktorý bude honeypot nasadzovaný, je nutné eliminovať sťahovanie externých závislostí na strane servera. Bezpečnostné opatrenia totižto výrazne eliminujú lokality, z ktorých môžu byť uvedené závislosti sťahované.

Produkty spoločnosti *Mavenir* sú nasadzované na serveroch, na ktorých je prevádzková upravená linuxová distribúcia RHEL[1] s balíčkovacím systémom RPM[2]. Z tohto dôvodu bola zvolená distribúcia vo forme RPM balíčka.

Do tohoto balíčka sú okrem samotných zdrojových súborov implementácie pribalené aj všetky externé závislosti. Vďaka tomu je jedinou prerekvizitou k úspešnej prevádzke honeypotu nainštalovaný interpret jazyka Python, a to konkrétne CPython vo verzii 3.6. Nutnosť takto špecifickej verzie je binárna povaha modulu *bafomet*, ktorý je vždy skompilovaný pre konkrétnu platformu.

[1] https://www.redhat.com/en/technologies/linux-platforms/enterprise-linux
[2] https://rpm.org/

Kapitola 6

Testovanie a nasadenie SS7 honeypotu

Overenie, či implementovaný systém spĺňuje podmienky a očakávania kladené jeho návrhom, je nevyhnutná súčasť vývoja každého programátorského diela. Náplňou tejto kapitoly je preto detailný opis procesu overovania správnosti implementácie.

Konkrétne sa kapitola venuje jednotlivým spôsobom testovania implementovaného honeypotu, ktorými sú manuálne testovanie a testovanie pomocou automatizovaných testovacích scenárov. Po predstavení jednotlivých spôsobov testovania nasleduje vyhodnotenie ich výsledkov. V závere kapitoly je načrtnutý ďalší zamýšľaný krok testovania, ktorým je experimentálne nasadenie nástroja.

6.1 Manuálne testovanie

Manuálne testovanie predstavovalo prvý spôsob overovania správnej funkcionality vznikajúceho honeypotu. Toto testovanie bolo uskutočnené s využitím populárneho open-source nástroja *Wireshark*[1].

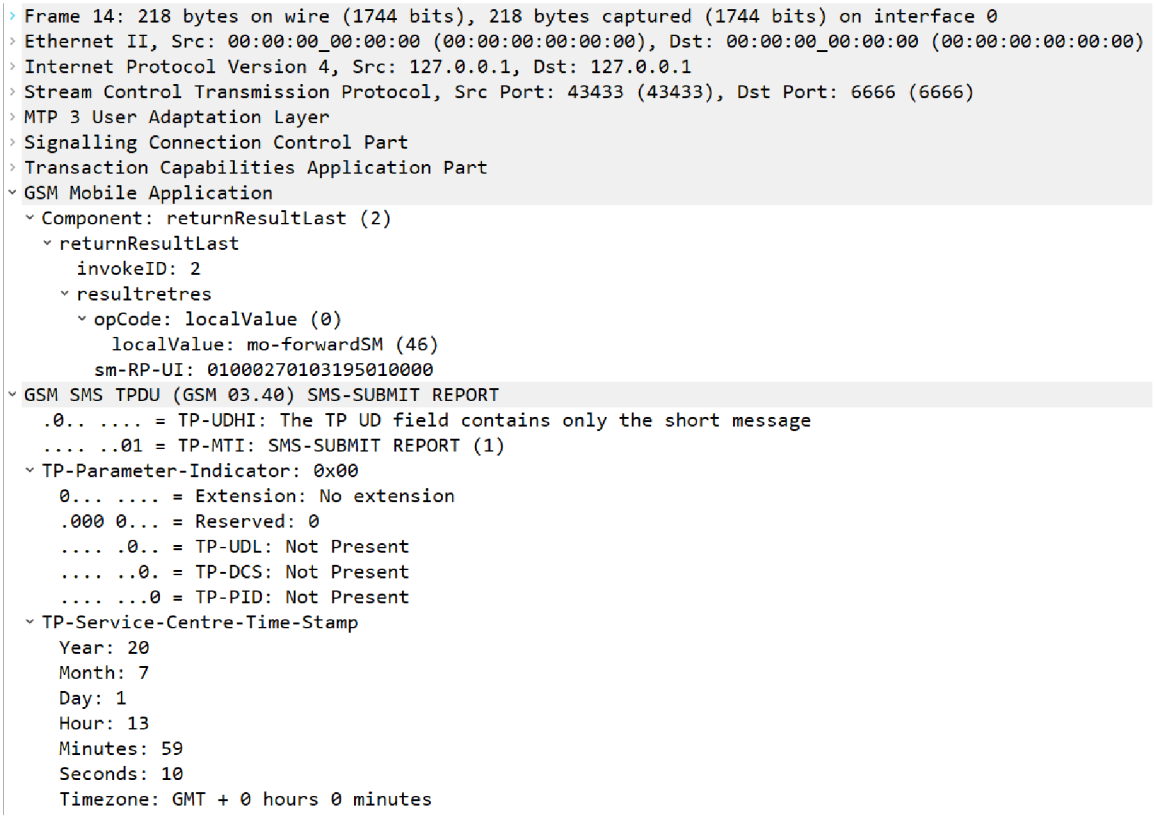
Tento nástroj pravdepodobne nie je nutné detailne predstavovať. Ide o nástroj, ktorý je schopný zachytávať sieťovú prevádzku a dekódovať parametre použitých sieťových protokolov. Nástroj podporuje dekódovanie veľkého množstva sieťových protokolov a to vrátanie protokolov rodiny SS7 a SIGTRAN.

Samotné manuálne testovanie bolo realizované nasledovným spôsobom. Po dokončení implementácie konkrétnej časti funkcionality bola do honeypotu zaslaná očakávaná správa. Táto správa, spoločne s honeypotom vygenerovanou a odoslanou odpoveďou, bola pomocou nástroja Wireshark zachytená, dekódovaná a následne bol manuálne detailne preskúmaný ich obsah.

Tento spôsob testovania dokázal odhaliť najzávažnejšie chyby v implementácii hneď na počiatku. Typicky to boli chýbajúce povinné parametre správy, alebo nesprávne kódovanie hodnôt parametrov. Všetky takéto chyby totižto viedli k zlyhaniu procesu dekódovania zachytenej správy v nástroji *Wireshark*, ktorý tento stav aj pomerne presne signalizuje užívateľovi nástroja.

Na obrázku 6.1 je zobrazená časť grafického užívateľského rozhrania nástroja Wireshark. Konkrétne sú tu zobrazené všetky parametre dekódovanej správy MAP MOFWSM.

[1] https://www.wireshark.org/



Obr. 6.1: Časť z grafického užívateľského rozhrania nástroja *Wireshark*.

6.2 Automatické testovanie

So zvyšujúcou sa komplexnosťou implementovaného riešenia sa manuálne testovanie stávalo komplikovanejšie a obťažnejšie a s pribúdajúcim počtom podporovaných typov správ bolo toto testovanie aj čoraz zdĺhavejšie. Takýto spôsob testovania navyše nedokázal pohodlne overovať zložitejšie mechanizmy implementácie, ako je napríklad podmienenosť prítomnosti parametru v odpovedi, či validita náhodne generovaných hodnôt prítomných v odpovediach.

Východiskom z tejto situácie bola automatizácia procesu testovania, ktorá spočívala vo vytvorení testovacích scenárov, ktoré vedú k využitiu testovaných častí implementácie. Vzhľadom na fakt, že samotný honeypot je implementovaný v jazyku Python, bolo tento programovací jazyk a jeho prostriedky zvolený aj na vytvorenie týchto testovacích scenárov.

6.2.1 Framework pytest

Konkrétne boli testovacie scenáre vytvorené s podporou testovacieho frameworku *pytest*[2]. Tento framework umožňuje jednoduché vytváranie testov, ich hierarchickú organizáciu, či správu ich závislostí.

Správa závislostí je realizovaná prostredníctvom jednotiek nazývaných *fixtures*. Táto jednotka má priradenú dobu života a sekvenciu príkazov (*statements*), ktoré sa majú vykonať pri jej inicializácii (na začiatku doby života) a sekvenciu príkazov, ktoré sa majú vykonať pri jej deštrukcii (na konci jej doby života). Táto jednotka je typicky realizovaná formou

[2] https://docs.pytest.org/en/stable/

funkcie generátora, kde inicializačná a deštrukčná časť je oddelená operátorom `yield`. Návratová hodnota takéhoto generátora je potom predávaná ako parameter všetkým na nej závislým testovacím funkciám (t.j. funkcie implementujúce jednotlivé testovacie scenáre).

Jednotlivé testovacie scenáre sú implementované ako funkcie (resp. metódy nejakej triedy), ktorých meno začína prefixom `test`. Syntakticky nasleduje za menom funkcie zoznam jej parametrov, v prostredí frameworku *pytest* je ale sémantický význam tohto zoznamu iný. Jednotlivé parametre predstavujú názvy *fixtures*, na ktorých je daný test závislý. Pri vykonávaní testu sú za tieto parametre dosadené hodnoty vrátené (alebo `yield`-nuté) z daných *fixtures* a v tele funkcie s nimi pracuje rovnako ako s parametrami funkcie v tradičnom chápaní. Príklad testovacej funkcie a *fixture* je znázornený na výpise 2.

```
1  import pytest
2
3  @pytest.fixture(scope="session")
4  def file_dependency():
5      file = open("/tmp/tmp_file.txt", "w")
6      yield file
7      file.close()
8
9  def test_scenario_1(file_dependency):
10     file_dependency.write("Hello world!\n")
```

Výpis 2: Príklad jednoduchej *fixture* a jednoduchého testu, ktorý je na nej závislý.

6.2.2 Použité fixtures

V implementovaných automatických testoch sa pracuje s troma *fixtures*. Prvá z nich sa stará o vytvorenie testovacej konfigurácie. V dočasnom adresári je vytvorený konfiguračný súbor s testovacou konfiguráciou a podadresár pre ukladanie výstupných CSV súborov. Cesta ku konfiguračnému súboru a cesta k CSV adresáru je potom vrátená ako dvojica pomocou operátora `yield`. Za týmto operátorom sa nachádza príkaz k odstráneniu celého dočasného adresára, ktorý sa vykoná až pri konci doby života tejto *fixture*, ktorá je nastavená na *session* – adresár bude odstránený až po dokončení všetkých testovacích scenárov.

Druhá *fixture* vytvára inštanciu honeypotu, ktorý je predmetom testovania. V procese inštancie je teda honeypot skonštruovaný a nakonfigurovaný podľa testovacej konfigurácie. Jeho inštancia je potom vrátená ako návratová hodnota. Doba života tejto *fixture* je taktiež *session*, takže všetky testovacie scenáre pracujú s jednou spoločnou inštanciou honeypotu.

Tretia *fixture* v rámci inicializácie skonštruuje inštanciu *bafomet*-u a nadväzuje spojenie s testovaným honeypotom. Okrem toho ďalej vytvára front, do ktorého sú ukladané odpovede poslané honeypotom a prijaté *bafomet*-om. Samotná inštancia a tento front je potom vrátený ako dvojica. Doba života je opäť *session*, čo mimo iné znamená, že nadviazané spojenie trvá naprieč jednotlivými testovacími scenármi.

6.2.3 Testovacie scenáre

Pre každý typ honeypotom podporovanej MAP správy bol implementovaný jednoduchý scenár, ktorý overuje jej správne spracovanie a reagovanie. Menovite sú to nasledovné scenáre, resp. funkcie, ktoré ich implementujú:

- `test_sendIMSI` predstavuje scenár, v rámci ktorého je do honeypotu zaslaná SIMSI správa s vymysleným MSISDN. Po obdržaní SIMSI odpovede je overené, či hodnota IMSI obdržaná v odpovedi patrí do rozsahu generátoru nakonfigurovaného v honeypote.

- `test_SRI` je scenár, kde je honeypotu zasielaná žiadosť SRI s vymyslenými hodnotami MSISDN a adresy GMSC. V obdržanej odpovedi je následne overené, či hodnoty IMSI či adresa VLR zodpovedajú nakonfigurovaným generátorom v honeypote.

- `test_PSI` a `test_ATI` sú dva parametrizované testovacie scenáre. Účelom tejto parametrizácie je otestovať rôzne kombinácie podporovaných príznakov v štruktúre *requestedInfo* (t.j. *imei*, *locationInformation*, *subscriberState* a *currentLocation*). Tieto dva testy sú preto každý spustený celkom 16-krát, vždy ale s inou sadou príznakov prítomných v PSI, resp. ATI žiadosti.

  V obdržaných odpovediach je kontrolovaná prítomnosť, resp. neprítomnosť jednotlivých voliteľných parametrov štruktúry *subscriberInfo* (t.j. *imei*, *locationInformation*, *subscriberState* a *currentLocationRetrieved*). Ďalej je overené, či hodnoty týchto parametrov odpovedajú nakonfigurovaným generátorom v honeypote a či zložený identifikátor bunky *cellGlobalIdOrServiceAreaId* je správne zakódovaný.

- `test_MOFWSM` implementuje scenár, pri ktorom je honeypotu zaslaná žiadosť MOFWSM. V tele žiadosti je ako parameter *sm-rp-ui* uložená TPDU štruktúra *SMS-SUBMIT*, čo je indikované aj tým, že v žiadosti je ako adresa príjemcu (parameter *sm-rp-da*) uložená adresa SMSC.

  V prijatej odpovedi je skontrolované, či sa v jej tele nachádza TPDU štruktúra *SMS-SUBMIT-REPORT*, čo je indikované prvým oktetom rovným 1.

- `test_MTFWSM` je analógiou k scenáru `test_MOFWSM`, s tým rozdielom, že v tele žiadosti je nesená štruktúra *SMS-DELIVER* indikovaná adresou SMSC ako adresou odosielateľa SM. V odpovedi je zase očakávaná štruktúra *SMS-DELIVER-REPORT*, čo implikuje, že prvý oktet je rovný 0.

Okrem už uvedených scenárov boli implementované ešte tri scenáre, ktoré overujú schopnosť honeypotu reagovať alternatívnymi spôsobmi (negatívnymi odpoveďami, alebo bez odpovede) na správy obsahujúce konkrétne CgPA, alebo na nepodporované správy. Sú to nasledovné scenáre:

- `test_cgpa_message_action_abort` je scenár, pri ktorom je honeypotu zaslaná správa s takou CgPA GT adresou, ktorá spôsobuje, že honeypot reaguje negatívne odpoveďou TCAP *abort*. Po obdržaní odpovede je overené, že tomu tak skutočne je a že hodnota *p-abortCause* je rovná hodnote nakonfigurovanej v honeypote.

- `test_cgpa_message_action_error` prestavuje scenár s obdržanou správou s takým CgPA, že to spôsobuje reakciu negatívnou odpoveďou TCAP *error*. Scenár je úspešný v prípade obdržania správy s hodnotou *errorCode* rovnou nakonfigurovanej hodnote.

- `test_unsupported_map_request` implementuje scenár, pri ktorom honeypot obdrží nepodporovanú MAP správu. Nakoľko honeypot nevie skonštruovať adekvátnu odpoveď, test predpokladá východzí spôsob reagovania – v testovacej konfigurácii nastavený na zahadzovanie takýchto správ a nereagovanie na ne.

Všetky testovacie scenáre overujú, okrem správnosti vytváraných odpovedí, aj správne zaznamenávanie aktivity honeypotu do CSV súboru. Pre účely tohto testovania je honeypot nakonfigurovaný tak, aby do každého CSV súboru uložil maximálne dva záznamy (parameter v konfiguračnom súbore `CSV_MAX_RECORDS` je nastavený na 2). To znamená, že pre každý testovací scenár je vytvorený nový CSV súbor s práve dvoma záznamami (žiadosť a odpoveď).

V závere každého testu je prislúchajúci súbor otvorený (je to vždy naposledy vytvorený súbor) a spracovaný. Jednotlivé parametre žiadosti či odpovede sú potom porovnané s hodnotami uloženými v CSV súbore.

6.3 Výsledky testovania

Manuálne testovanie, rovnako aj všetky testovacie scenáre uvedené v sekcii 6.2.3 dopadli úspešne. Honeypot reagoval na všetky prijaté správy očakávaným spôsobom, tzn. vytvorením validnej MAP odpovede, či vytvorením niektorej negatívnej odpovede, alebo nereagovaním žiadnou odpoveďou. Parametre jednotlivých odpovedí patrili do nakonfigurovaných rozsahov, alebo vyhovovali nakonfigurovanému regulárnemu výrazu.

Obdržanie validných odpovedí na prijaté správy mimo iné potvrdzuje správnosť implementácie mechanizmov zapuzdrujúcich použitie modulu *bafomet* (je asi zrejmé, že ich správna funkčnosť je nevyhnutná podmienka k správnej funkčnosti honeypotu ako celku).

Obsah vygenerovaných CSV súborov rovnako tak zodpovedal očakávanému stavu. Súbory obsahovali očakávané záznamy konkrétnych správ a zaznamenané parametre sa zhodovali s parametrami v správach samotných. Nakoľko bolo úspešné spárovanie záznamov s jednotlivými správami závislé na ukladaní práve dvoch záznamov do jedného súboru, bola nepriamo overená aj funkčnosť mechanizmu „otáčania“ CSV súboru po dosiahnutí nakonfigurovaného limitu čo do počtu záznamov v jednom CSV súbore.

Vytvorený mix testovacích scenárov zároveň nepriamo potvrdil správnu funkčnosť rozhodovacích mechanizmov (rozhodovanie spôsobu reakcie na prichádzajúcu správu, či voľba správnej rutiny pre obsluhu konkrétneho podporovaného typu prichádzajúcej správy).

Všetky horeuvedené konštatovania preto dovoľujú formulovať záver, že implementácia odpovedá návrhu.

6.4 Experimentálne nasadenie honeypotu

Experimentálne nasadenie je ďalším zamýšľaným spôsobom testovania a odladenia implementovaného nástroja. V dobe dokončovania tejto práce boli započaté prvé kroky k jeho prvému nasadeniu. Toto nasadenie bude možné realizovať vďaka partnerom spoločnosti *Mavenir*, ktorí jej pre jej interné účely poskytujú konektivitu do globálnej SS7 siete. Zo skúseností z minulosti už bolo zároveň potvrdené, že zariadenia v tejto sieti sú cieľom mnohých nevyžiadaných, či podozrivých správ, takže tu existuje veľký potenciál na overenie konceptu SS7 honeypotu ako takého.

Honeypot bude do siete nainštalovaný v súlade s návrhom (pozri 4.3), tzn. že honeypotu bude priradená jedna z GT adries, ktoré má firma k dispozícii a v nástroji SIF bude všetka prevádzka smerujúca ja tento GT presmerovávaná do M3UA spojenia s honeypotom. Ako už bolo taktiež spomenuté, softvérové riešenia spoločnosti *Mavenir* sú nasadzované spoločne s vlastnou Linux distribúciou. Distribúcia honeypotu na server preto bude realizovaná pomocou RPM balíčka (pozri 5.6).

Kapitola 7

Záver

V rámci tejto práce boli preštudované princípy činnosti mobilných telefónnych sietí. Nakoľko je táto téma dosť obšírna, boli všetky informácie, ktoré sú relevantné pre túto prácu, zhrnuté v prvej polovici tejto práce. Konkrétne tu bol popísaný historický vývin mobilných sietí, či ich základná topológia spolu s vysvetlením úloh a zodpovedností jednotlivých prvkov siete. V práci boli taktiež rozoberané sieťové protokoly, ktoré sa v týchto sieťach používajú a aké role v sieti plnia. V kontexte všetkých týchto informácií potom boli čitateľovi predstavené niektoré bezpečnostné zraniteľnosti SS7 sietí a spôsoby ich zneužitia.

Ďalej boli preštudované nástroje typu honeypot, princíp ich činnosti, či dôvod ich nasadenia v sieti. Všetky tieto informácie boli spracované a prezentované v ďalšej časti tejto práce. V nej bolo taktiež predstavené základné delenie týchto nástrojov, spolu s výhodami a nevýhodami jednotlivých kategórií.

V spolupráci s odborným konzultantom boli zvolené niektoré typické prípady zneužitia štandardu SS7, na ktoré by mal byť honeypot schopný reagovať. Bol spracovaný podrobný návrh, ako by mal tento honeypot fungovať a ako by mal reagovať na jednotlivé podporované správy. Do návrhu boli zapracované aj iné pripomienky konzultanta, ktoré nesúviseli so samotnou funkcionalitou (napríklad určité preferencie v spôsobe konfigurácie, či spôsob ukladania výstupných CSV súborov).

Podľa vytvoreného návrhu bol honeypot implementovaný. Ako implementačný jazyk bol zvolený jazyk *Python* a to z dôvodu existujúcej implementácie SIGTRAN zásobníka vo forme rozširujúceho modulu *bafomet*. Pri implementácii bol kladený dôraz na izolovanie tohto modulu od logiky samotného honeypotu, čo výrazne zjednodušuje prípadné budúce zmeny v jeho implementácii. Implementácia taktiež využíva možnosť voliteľného statického typovania, kontrola ktorého zachytila veľké množstvo chýb priamo pri vývoji.

Implementovaný honeypot bol otestovaný najprv manuálnym spôsobom a neskôr aj pomocou vytvorených automatických testov. Cieľom testov bolo overiť, či na všetky podporované, resp. aj nepodporované správy honeypot reaguje spôsobom špecifikovaným v návrhu a v súlade so samotnými SS7 štandardmi.

Úspešné vykonanie všetkých testovacích scenárov dokladá, že implementovaný honeypot dokáže reagovať na vybrané MAP správy a teda reagovať na útoky spočívajúce v zneužívaní týchto správ. Nakoľko však štandard špecifikuje ďaleko väčšie množstvo MAP správ, nie sú možnosti SS7 honeypotu zďaleka vyčerpané. Jedným zo smerov jeho ďalšieho vylepšovania preto pravdepodobne bude doimplementovanie podpory pre ďalšie MAP správy.

Uvažované sú napríklad MAP správy SRISM, SRILCS, či SRIGPRS. Tieto správy, podobne ako správa SRI, pri zneužití umožňujú získanie IMSI účastníka siete.

Ďalším potencionálnym vylepšením by mohlo byť rozšírenie CSV výstupu o ďalšie parametre. Potreba tohto rozšírenia, a hlavne o ktoré konkrétne parametre bude nutné výstup rozšíriť, pravdepodobne vyplynie až z analýzy zaznamenaných dát z reálneho nasadenia do ostrej prevádzky.

Implementovaný honeypot je bezstavový, čo výrazne zjednodušuje jeho nasadenie, avšak zároveň to aj znamená, že v prípade prijatia rovnakých MAP žiadostí budú odpovede na ne typicky rôzne (náhodný výber parametrov v odpovedi). Jedno z možných vylepšení môže preto spočívať v udržiavaní kontextu, čím by bolo možné dosiahnuť, že honeypot bude reagovať rovnakými odpoveďami na rovnaké žiadosti (v prípade zisťovania IMSI užívateľa). V prípade získavania polohy účastníka by zase bolo možné na opakované dopyty odpovedať susednými CGI, čím by bolo možné simulovať pohyb užívateľa naprieč bunkami siete.

Literatúra

[1] 3GPP. *Mobile Application Part (MAP) Specification*. Technical Specification (TS) 09.02. 3rd Generation Partnership Project (3GPP), marec 2004.

[2] 3GPP. *Mobile Application Part (MAP) specification*. Technical Specification (TS) 29.002. 3rd Generation Partnership Project (3GPP), jún 2019.

[3] ITU-T. *The international public telecommunication numbering plan*. Recommendation. ITU Telecommunication Standardization Sector (ITU-T), október 2010.

[4] ITU-T. *The international identification plan for public networks and subscriptions*. Recommendation. ITU Telecommunication Standardization Sector (ITU-T), september 2016.

[5] George, T., Bidulock, B., Dantu, R., Schwarzbauer, H. a Morneault, K. *Signaling System 7 (SS7) Message Transfer Part 2 (MTP2) - User Peer-to-Peer Adaptation Layer (M2PA)*. RFC 4165. RFC Editor, september 2005.

[6] Immonen, M. *Signalling over IP – a step closer to an all-IP network*. Stockholm, Sweden, 2005. Diplomová práca. Royal Institute of Technology (KTH).

[7] Jeff Hewett, L. D. a. *Signaling System No. 7 (SS7/C7): protocol, architecture, and services*. 1. vyd. Cisco Press, 2005. ISBN 1-58705-040-4.

[8] Jensen, K., Do, T. V., Nguyen, H. T. a Arnes, A. Better Protection of SS7 Networks with Machine Learning. In: IEEE. *2016 6th International Conference on IT Convergence and Security (ICITCS)*. 2016, s. 1–7. ISBN 978-1-5090-3765-0.

[9] Morneault, K. a Pastor Balbas, J. *Signaling System 7 (SS7) Message Transfer Part 3 (MTP3) - User Adaptation Layer (M3UA)*. RFC 4666. RFC Editor, september 2006.

[10] Provos, N. *Virtual honeypots – From Botnet Tracking to Intrusion Detection*. 1. vyd. Addison-Wesley, 2008. ISBN 978-0-321-33632-3.

[11] Pužmanová, R. *Moderní komunikační sítě od A do Z*. 2. vyd. Computer Press, 2006. ISBN 80-251-1278-0.

[12] ITU-T. *Functional description of the message transfer part (MTP) of Signalling System No. 7*. Recommendation. ITU Telecommunication Standardization Sector (ITU-T), marec 1993.

[13] ITU-T. *Functional description of the signalling connection control part.* Recommendation. ITU Telecommunication Standardization Sector (ITU-T), marec 2001.

[14] ITU-T. *Signalling connection control part formats and codes.* Recommendation. ITU Telecommunication Standardization Sector (ITU-T), marec 2001.

[15] ITU-T. *Functional description of transaction capabilities.* Recommendation. ITU Telecommunication Standardization Sector (ITU-T), jún 1997.

[16] R. Stewart, E. *Stream Control Transmission Protocol.* RFC 4960. RFC Editor, september 2007.

[17] Rao, S. P., Oliver, I., Holtmanns, S. a Aura, T. We know where you are! In: IEEE. *2016 8th International Conference on Cyber Conflict (CyCon).* 2016, s. 277–293. ISSN 2325-5374.

[18] Remenyi, D. *ECIW2006-Proceedings of the 5th European Conference on i-Warfare and Security: ECIW 2006.* 1. vyd. Academic Conferences, 2006. ISBN 978-1-905305-20-6.

[19] Riegel, M. a Tüxen, M. Mobile SCTP transport layer mobility management for the Internet. In: FESB, Split. *SoftCOM 2002 – International conference on software, telecommunications and computer networks.* 2002, s. 305–309. ISBN 953-6114-52-6.

[20] Russell, T. *Signaling System #7.* 6. vyd. McGraw-Hill, 2014. ISBN 978-0-07-182214-5.

[21] Schiller, J. *Mobile communications.* 2. vyd. Addison-Wesley, 2003. ISBN 0-321-12381-6.

[22] Shafranovich, Y. *Common Format and MIME Type for Comma-Separated Values (CSV) Files.* RFC 4180. RFC Editor, október 2005.

[23] Spitzner, L. *Honeypots tracking hackers.* 1. vyd. Addison-Wesley, 2003. ISBN 0-321-10895-7.

[24] Yeboah, P. N. *Proposal and implementation of an IDS for potential SMS spam signaling messages on SS7.* 2016. Diplomová práca. Norwegian University of Science and Technology.

Príloha A

Obsah CD

- `/`
  - `doc/` — Zdrojové LaTeX súbory tejto práce.
  - `ss7-honeypot/` — Súbory projektu SS7 honeypot.
    - `ss7_honeypot/` — Zdrojové súbory implementácie.
    - `tests/` — Zdrojové súbory testov.
    - `baf2.cpython-36m-x86_64-linux-gnu.so` — Skompilovaný modul *bafomet*.
    - `config.py` — Ukážka konfiguračného súboru.
    - `Dockerfile` — Dockerfile prostredia so všetkými závislosťami.
    - `README` — Návod na spustenie nástroja alebo automatických testov.
  - `thesis.pdf` — Text tejto práce vo formáte PDF.