VŠB TECHNICKÁ UNIVERZITA OSTRAVA
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

# Bezpečnostní analýza podvodných volání v prostředí IP telefonie

Security Analysis of Fraud Calls in IP Telephony Infrastructures

Bc. Jiří Jakuba

Diplomová práce

Vedoucí práce: Ing. Filip Řezáč, Ph.D.

Ostrava, 2023

VŠB TECHNICKÁ UNIVERZITA OSTRAVA | FAKULTA ELEKTROTECHNIKY A INFORMATIKY

Č.j.VSB/23/030235
vsboee2300C4EC

# Zadání diplomové práce

Student: **Bc. Jiří Jakuba**

Studijní program: N2647 Informační a komunikační technologie

Studijní obor: 2601T013 Telekomunikační technika

Téma: Bezpečnostní analýza podvodných volání v prostředí IP telefonie
Security Analysis of Fraud Calls in the IP Telephony Infrastructures

Jazyk vypracování: čeština

Zásady pro vypracování:

Cílem diplomové práce je provést důkladnou analýzu podvodných volání, která se v různých podobách vyskytují v současných telefonních infrastrukturách. Hlavním záměrem je navrhnout a implementovat komplexní systém pro realizaci tzv. Wangiri útoků s cílem získat informace o chování a možných slabinách tohoto typu podvodu s následnou definicí bezpečnostních zásad a protiopatření.

Zadání:
1. Detailně nastudujte problematiku podvodných útoků v současné IP telefonii.
2. Proveďte analýzu nástrojů pro možnou realizaci útoku Wangiri s využitím autonomního vyhledáváče telefonních záznamů.
3. Navrhněte testovací topologii a systém pro generování podvodných hovorů.
4. Proveďte instalaci, konfiguraci a bezpečnostní analýzu takového systému.
5. Důkladně zdokumentujte implementaci systému a průběh testování s výsledky.
6. Na základě výsledků bezpečnostní analýzy a testování definujte vhodná protiopatření a bezpečnostní zásady.

Seznam doporučené odborné literatury:

[1] M. Arafat, A. Qusef, G. Sammour - Detection of Wangiri Telecommunication Fraud Using Ensemble Learning, 2019, DOI: 10.1109/JEEIT.2019.8717528

[2] I. I. Androulikadis - VoIP and PBX Security and Forensics: A Practical Approach, 2016, Springer, ISBN-13: 978-3319297200

[3] M. Collier, D. Endler - Hacking Exposed Unified Communications & VoIP Security Secrets & Solutions, 2013, McGraw-Hill Education, ISBN-13: 978-0071798761

17. listopadu 2172/15
708 00 Ostrava-Poruba
Česká republika

spojovatelka: +420 597 321 111
epodatelna: epodatelna@vsb.cz
ID datové schránky: d3kj88v

IČ: 61989100
DIČ: CZ61989100

email: studijni.fei@vsb.cz
www.fei.vsb.cz

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **Ing. Filip Řezáč, Ph.D.**

Datum zadání: 01.09.2021

Datum odevzdání: 30.04.2023

Garant studijního oboru: doc. Ing. Petr Šiška, Ph.D.

V IS EDISON zadáno: 28.04.2023 08:42:29

## Abstrakt

Diplomová práce se zaměřuje na analýzu podvodných volání v IP telefonii, především na podvodná volání typu Wangiri. Cílem této diplomové práce je návrh a implementace testovacího systému pro realizaci Wangiri útoků a jejich následná analýza. Součástí práce je realizace navrženého systému, který obsahuje autonomní vyhledávač telefonních záznamů (web scraper), který zpracovává telefonní čísla z webových stránek, a ta jsou následně pomocí generátoru telefonního provozu SIPp využita pro provedení samotného útoku Wangiri. Systém také obsahuje Asterisk VoIP ústřednu, přes níž jsou všechna podvodná volání realizována. Průběh a výsledky celého experimentu jsou zdokumentovány, zanalyzovány a následně jsou navrhnuta vhodná protiopatření vůči Wangiri podvodu.

## Klíčová slova

Wangiri; podvod; volání; SIP; VoIP; Asterisk; Python; SIPp

## Abstract

The diploma thesis focuses on the analysis of fraudulent calls in IP telephony, especially on Wangiri type of fraud. The aim of this diploma thesis is to design and implement an experimental system for the implementation of Wangiri attacks and their subsequent analysis. The work contains implementation of the proposed system that consists of an autonomous search engine for telephone records (web scraper), which processes telephone numbers from websites and these are then used by the SIPp telephone traffic generator to carry out the Wangiri attack itself. The system also includes an Asterisk VoIP exchange, through which all fraudulent calls are made. The course and results of the whole experiment are documented, analyzed and then suitable countermeasures against the Wangiri fraud are proposed.

## Keywords

Wangiri; fraud; calls; SIP; VoIP; Asterisk; Python; SIPp

## Poděkování

Rád bych poděkoval panu Ing. Filipu Řezáčovi, Ph.D. za jeho vstřícnost, trpělivost, pomoc a věcné připomínky při vypracování této diplomové práce.

# Obsah

# Seznam použitých zkratek a symbolů

| | | |
|---|---|---|
| API | – | Application Programming Interface |
| ARP | – | Address Resolution Protocol |
| B2BUA | – | Back to Back User Agent |
| CDR | – | Call Detail Records |
| CFCA | – | Communications Fraud Control Association |
| CLI | – | Command Line Interface |
| CR | – | Carriage Return |
| CSV | – | Comma-separated Values |
| CVC | – | Card Verification Code |
| CVV | – | Card Verification Value |
| DAHDI | – | Digium Hardware Device Interface |
| DHCP | – | Dynamic Host Configuration Protocol |
| DNS | – | Domain Name System |
| DoS | – | Denial of Service |
| HTML | – | Hypertext Markup Language |
| HTTP | – | Hypertext Transfer Protocol |
| IAX | – | Inter-Asterisk Exchange |
| IMS | – | IP Multimedia System |
| IP | – | Internet Protocol |
| IRPN | – | International Premium Rate Number |
| IT | – | Information Technology |
| IVR | – | Interactive Voice Response |
| JSON | – | JavaScript Object Notation |
| LAN | – | Local Area Network |
| LDAP | – | Lightweight Directory Access Protocol |
| LF | – | Line Feed |
| MAC | – | Media Access Control |
| PBX | – | Private Branch Exchange |

| | | |
|---|---|---|
| PSTN | – | Public Switched Telephone Network |
| RFC | – | Request For Comments |
| RTCP | – | RTP Control Protocol |
| RTP | – | Real-time Transport Protocol |
| SDP | – | Session Description Protocol |
| SIP | – | Session Initiation Protocol |
| SMTP | – | Simple Mail Transfer Protocol |
| SSL | – | Secure Sockets Layer |
| TCP | – | Transmission Control Protocol |
| TFTP | – | Trivial File Transfer Protocol |
| TLS | – | Tranport Layer Security |
| UA | – | User Agent |
| UAC | – | User Agent Client |
| UAS | – | User Agent Server |
| UDP | – | User Datagram Protocol |
| URI | – | Uniform Resource Identifier |
| URL | – | Uniform Resource Locator |
| UTF | – | Unicode Transformation Format |
| VoIP | – | Voice over Internet Protocol |
| VoLTE | – | Voice over Long Term Evolution |
| XML | – | Extensible Markup Language |

# Seznam obrázků

# Kapitola 1

# Úvod

Komunikace je neodmyslitelnou součástí života každého člověka na světě. Dnes si spousta lidí nedokáže bez telefonu, především toho mobilního, svoje bytí ani představit. S vývojem technologií a neustálým posunem výrobků směrem kupředu se navíc objevily i nové způsoby, jakými můžeme jeden druhého kdykoli kontaktovat. Ať už se chceme s někým spojit prostřednictvím textové konverzace, nebo raději volíme verbální a v dnešní době čím dál více populární videohovor, či videokonferenci, díky vzniku internetu a chytrých telefonů to nikdy nebylo jednodušší.

Jedním z nejrozšířenějších prostředků komunikace je IP telefonie. IP telefon nebo aplikace jako Skype, Messenger, Whatsapp, Microsoft Teams zná nebo již používá většina uživatelů internetu. Stejně jako u jiných informačních technologií však i v prostředí IP telefonie existují bezpečnostní rizika a s nimi spojené podvody a útoky, prostřednictvím kterých se snaží útočník získat cizí informace, nebo se dokonce obohatit na nevědomosti nebo nepozornosti běžných uživatelů.

Cílem této diplomové práce je návrh a implementace systému pro realizaci Wangiri útoků a jejich následná analýza. V první části diplomové práce jsou popsány různé druhy podvodů a útoků v prostředí IP telefonie, se kterými se mohou uživatelé této technologie v dnešní době setkat. Následuje analýza a srovnání vybraných nástrojů, pomocí kterých je možné zprostředkovat systém pro provedení podvodu Wangiri. Jedná se o generátory provozu, telefonní ústředny pro IP telefonii a autonomní vyhledávače části textu (angl. web scraper). Autonomní vyhledávač umožňuje získávat telefonní čísla různých subjektů přímo z webových stránek na internetu.

Praktická část začíná kapitolou o návrhu a popisu celého řešení, představení systému a odůvodnění volby jednotlivých jeho částí. Další kapitoly se věnují samotné instalaci, konfiguraci, dokumentaci, implementaci útoku Wangiri a testování s finálními výsledky v laboratorním prostředí. Poslední kapitola se zabývá doporučeními a vhodnými protiopatřeními, jak předcházet tomu, aby se uživatel stal obětí Wangiri podvodu. Všechny zásady vycházejí z výsledků rešerše, testování systému a získaných poznatků během vypracování diplomové práce.

# Kapitola 2

# IP telefonie - podvody a útoky

Jelikož je práce zaměřena především na prostředí IP telefonie, obsahuje druhá kapitola stručně popis této technologie. V další části je uvedena problematika útoků a podvodů v současné IP telefonii, jejich druhy a především způsob, jakým jsou prováděny.

## 2.1 IP Telefonie

IP telefonie (angl. Internet Protocol Telephony) je termín používaný k popisu technologií, které využívají protokolů UDP, TCP/IP pro výměnu informací tradičně předávaných přes veřejnou telefonní síť (PSTN - Private Switched Telephone Network). Jedná se především o hovory, fax, videohovory. Hovor je přenášen prostřednictvím paketů přes LAN (Local Area Network) sítě nebo internet. IP telefonie a VoIP (Voice over Internet Protocol) ovšem stále umožňuje přímý přístup k PSTN. Lze tedy zprostředkovávat hovory pouze přes VoIP, tak i z VoIP prostředí na PSTN.[1][2]

Již od počátku 90. let 20. století internet a TCP/IP začaly měnit možnosti vývoje telefonní a jiné komunikace. IP protokol se od té doby stal hlavním prostředkem pro přenos informací. V dnešní době je tato technologie upřednostňovaná nejenom ve firmách jako mnohem levnější, a často také lépe konfigurovatelná alternativa ke klasickým veřejným telefonním sítím.[1][2] Mnoho podniků a institucí i mimo IT průmysl používá ke komunikaci VoIP (Voice over Internet Protocol) aplikace jako jsou Microsoft Teams, Zoom, Google Meet atd.

Velkou výhodou IP telefonie je její dostupnost. Není třeba žádná speciální infrastruktura (např. pevná linka), ale stačí jakýkoli běžný počítač nebo mobilní telefon a přístup k síti, což je dnes již součást většiny moderních domácností a firem. Tradiční mobilní systémy vyžadují náročnou implementaci složité architektury, jejichž návrh a provoz může stát řádově mnohem více peněz. Také jsou často mnohem složitější na administraci, konfiguraci a provoz.[2]

Provoz VoIP také značně snižuje náklady. VoIP umožňuje firmám se jednoduše spojit se svými pobočkami nebo jinými firmami kdykoli pouze za cenu VoIP platformy, která je ve výsledku mnohem nižší, než pokud by byl započítáván každý hovor podle jeho délky, jako to bývá u klasických veřejných

telefonních sítí. Cena za jeden hovor je tak násobně snížena a náklady např. za roamingové nebo mezistátní tarify prakticky odstraněny.[2]

IP telefonie má i své nevýhody. Za jednu z největších se dá považovat nutnost připojení k internetu nebo jiné síti. Především bezdrátová připojení a připojení s nižšími rychlostmi nemusí poskytovat požadovanou kvalitu služby. Může také docházet k výpadkům spojení. Pokud chce uživatel využívat IP telefonii pro volání na služby PSTN, musí často za takovou službu platit poskytovateli IP řešení příplatek navíc. Není neobvyklé, kdy tato služba je dražší, než pokud by klient využil pro komunikace jen PSTN. To lze považovat za další nevýhodu IP telefonie.

Následuje popis některých protokolů, které VoIP využívá. Zaměřil jsem se především na protokol SIP, který je stěžejní pro tuto diplomovou práci.

## 2.2 SIP

SIP (Session Initiation Protocol) je protokolem pro sestavení, modifikaci a ukončení obecné relace přes internet. SIP je textově orientovaný protokol, nejvíce se podobná HTTP protokolu. Využívá modelu klient – server. Klient zasílá požadavky na server, server posílá odpovědi na tyto požadavky. Jedná se o end-to-end orientovaný signalizační protokol. To znamená, že všechny informace o logice systému jsou uloženy do koncových zařízení.[1]

Nejčastěji je používán pro audio komunikaci. SIP pracuje na aplikační vrstvě, byl navržen tak, aby byl snadno implementovatelný, rozšiřitelný a dostatečně flexibilní. Protokol je užíván pro sestavení, modifikaci a ukončení spojení s jedním nebo více účastníky, ale není jediným protokolem, který je potřebný pro audiovizuální komunikaci. Ve spojení se SIP protokolem se nejčastěji používají protokoly RTP a SDP.[1][3]

SIP podporuje různé aspekty při navazování a ukončování multimediální komunikace. Jedná se o zjišťování umístění cílového uživatele, stav dostupnosti volaného, určení médií a jejich parametry, vytvoření parametrů pro relace mezi volaným a volajícím a správu relací, jejich ukončování, modifikaci a přidávání služeb.[3]

### 2.2.1 Identifikace v SIP

Každý objekt je identifikován prostřednictvím SIP URI (SIP Uniform Resource Identifier), což je jmenný identifikátor zařízení. Ty jsou velmi podobné hlavičkám v SMTP komunikaci. Každý jmenný identifikátor obsahuje uživatelské jméno a doménové jméno (host), heslo (použití této části není z bezpečnostních důvodu doporučováno), port (zpravidla UDP 5060) a parametry upřesňující požadavek URI.

Obecný formát SIP URI:

```
sip:uzivatel:heslo@host:port;uri-parametry?hlavicky
```

Příklad SIP URI:

```
sip:jakuba@vsb.cz
sip:123567894@google.com
```

### 2.2.2 Architektura SIP

Architektura protokolu SIP se skládá z těchto hlavních komponent: User Agents (UA), Proxy server, Redirect server, Registrar server. Každá síť zpravidla sestává z více prvků architektury.[1][4]

#### 2.2.2.1 User Agent

User Agent je koncový bod, ve kterém vznikají a jsou ukončovány SIP relace. Většinou se jedná o hardwarové IP telefony, aplikace, systémy s interaktivní hlasovou odezvou, PSTN brány apod. Téměř každý koncový bod obsahuje dvě části: UAC (User Agent Client) a UAS (User Agent Server). Pokud koncový bod obsahuje obě části, je zpravidla označován jen jako UA. UAC je část zodpovědná za odesílání požadavků a přijímání odpovědí. UAS na druhou stranu přijímá požadavky a odesílá odpovědí.

Existuje speciální typ User Agenta, a to B2BUA (Back to Back User Agent). Jedná se o UA vložené do cesty uprostřed spojení. Na B2BUA je první relace ukončena a vytvořena další směrem na cíl. Rozdílem mezi B2BUA a SIP Proxy je ten, že SIP proxy nesestavuje nové zprávy k oběma stranám, ale pouze přeposílá zprávy vytvořené oběma stranami, jež spolu komunikují. To má za následek, že počet obsloužených požadavků na spojení je u B2BUA znatelně nižší, neboť dochází k vytváření spojení navíc při každé komunikaci. Příkladem B2B User Agenta je pobočková ústředna Asterisk.[1] [3]

#### 2.2.2.2 SIP proxy server

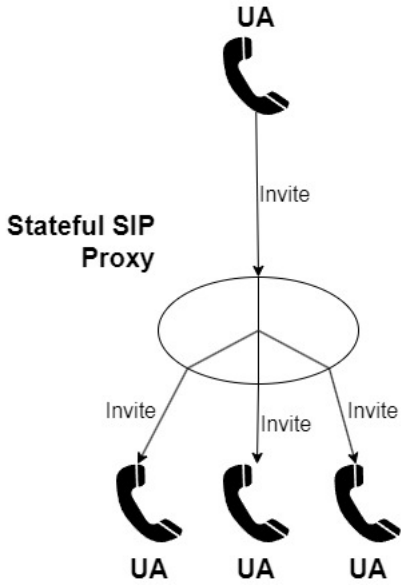
SIP proxy servery jsou důležitou složkou infrastruktury sítě využívající protokol SIP. Zajišťují směrování SIP žádostí podle umístění koncové bodu, autentizaci koncových bodů a lze je rozšířit o další služby (např. přesměrování). Příkladem SIP proxy serveru je Kamailio. [1] [3]

Při směrování žádostí SIP proxy nejdříve rozhoduje, zda musí na žádost odpovědět. Například když proxy potřebuje zjistit údaje o klientu. V takovém případě vystupuje prvek jako UAS. Poté musí určit, na který další prvek bude žádost směrována (next hop) a zda bude modifikována. Odpověď na žádost prochází vždy stejnou cestou v opačném směru.[3]

Každá proxy může při každé nové žádosti operovat v jednom ze dvou módů - stateless nebo stateful.

- **Stateless proxy** - Slouží jako jednoduchý směrovací prvek. Přeposílá žádosti na základě hledání dalších SIP proxy dle informací z jednotlivých žádostí. Informace o předání zpráv po přeposlání zprávy zahazuje.[3]

- **Stateful proxy** - Na rozdíl o stateless proxy ukládá informace o příchozích žádostech, dokud nedojde k ukončení transakce. Využívá tyto informace pro zpracování dalších budoucích zpráv týkajících se stejné žádosti. Může také rozeslat žádost na více cílů najednou. Každý takový požadavek musí být zpracován SIP proxy ve stavu stateful.[3]

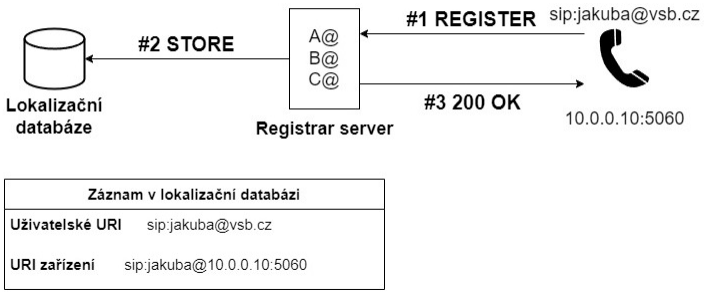



Obrázek 2.1: Stateful SIP proxy - rozeslání žádosti na více cílů

#### 2.2.2.3 SIP registrar server

SIP registrar je speciální část SIP serveru. Přijímá požadavky uživatelů na registraci. Tím o nich získá jejich informace o poloze tzn. IP adresu, port a jméno. Tyto informace ukládá do lokalizační databáze. Ta je využívána Proxy serverem.[1]

Na obrázku 2.2 je zobrazena SIP registrace. Zpráva REGISTER obsahující užívatelské URI *sip:jakuba@vsb.cz* a URI zažízení *sip:jakuba@10.0.0.10:5060* je zaslána na Registrar server. Ten zaznamená její obsah do lokalizační databáze. Pokud byla registrace úspěšná, zašle registrar server zpět zprávu 200 OK a registrace je dokončena. Má však omezenou dobu platnosti. Po její expiraci je nutné proces registrace zopakovat. V případě zobrazeném na obrázku 2.2 došlo k namapování adresy URI uživatele *sip:jakuba@vsb.cz* na doménová jména *@a.cz, @b.cz, @c.cz.*

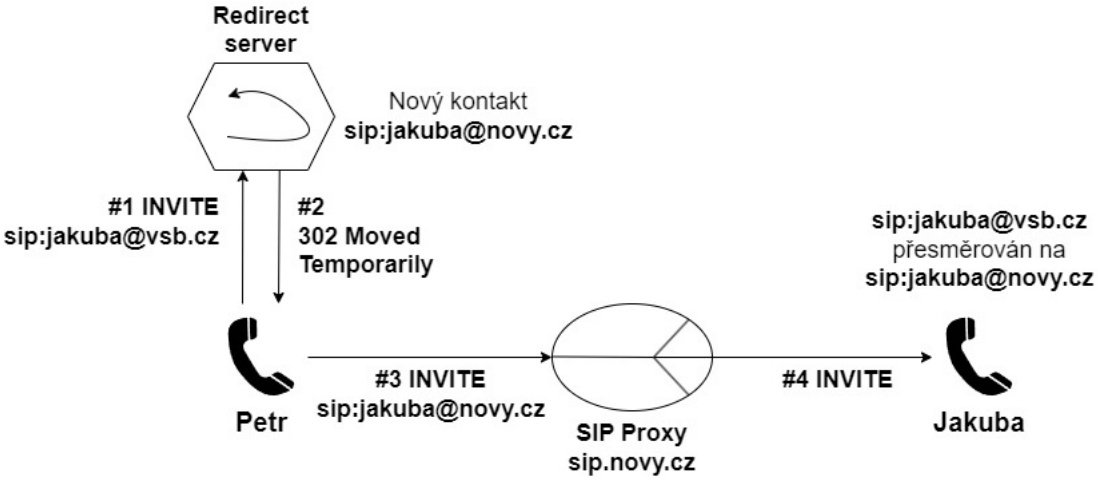



Obrázek 2.2: Registrace na registrar serveru

#### 2.2.2.4 SIP Redirect server

SIP redirect server slouží k lokalizaci uživatele. Přijímá požadavek, vyhledá příjemce požadavku v lokalizační databázi registrar serveru. Poté vytvoří seznam momentální lokace uživatele a pošle o ní odpověď odesílateli požadavku ve formě zprávy třídy 3xx. Odesílatel získá seznam lokací. Další požadavky zasílá již podle lokací z obdrženého seznamu.[1]

Na obrázku 2.3 je znázorněn průběh přesměrování přes redirect server. Petr volá Jakubovi. Zprávu INVITE přijme redirect server. Ten zjistí, že Jakuba má nový kontakt v doméně *novy.cz* a zasílá zpět odpověď 302 Moved Temporarily, ve které přidá do pole contact nové SIP URI volaného. Koncový bod Petr znovu zašle požadavek INVITE, tentokrát na nové URI změněné redirect serverem. Požadavek se dostane na SIP proxy server *sip.novy.cz* a ten přepošle INVITE na koncový bod Jakuba.




Obrázek 2.3: Přesměrování přes redirect server

### 2.2.3 SIP zpráva

SIP je textově orientovaný protokol a používá znakovou sadu UTF-8. SIP zprávy se dělí na žádosti z klienta na server a odpovědi serveru na klienta. Oba typy zprávy se skládají z počátečního řádku start-line, jednoho nebo více polí hlavičky (header), jednoho prázdného řádku označujícího konec polí záhlaví (CRLF - carriage-return line-feed) a nepovinného těla zprávy.[3]

```
generic-message  =  start-line
*message-header
CRLF
[ message-body ]
```

#### 2.2.3.1 SIP žádosti

SIP žádosti, nebo také metody, slouží k sestavení, ukončení nebo aktualizace spojení. V SIP specifikaci RFC 3261[3] je definováno šest základních metod:

- **REGISTER** - metoda pro registraci informací o kontaktu (IP adresa, port, jmenná adresa)
- **INVITE** - metoda pro zahájení procesu navázání relace
- **ACK** - metoda pro potvrzení přijetí žádosti na navázání spojení
- **CANCEL** - metoda pro zrušení spojení
- **BYE** - žádost pro ukončení probíhající relace
- **OPTIONS** - metoda sloužící k výměně informací o vlastnostech koncového bodu (podporované metody, používaný kodek, klapky)

V rámci dalších specifikací byly definovaný nové druhy SIP metod. Některé z nich jsou:

- **INFO** - přenos informací během spojení
- **NOTIFY** - doručení zprávy o události
- **UPDATE** - změna stavu spojení
- **SUBSCRIBE** - přihlášení k upozornění na vzniklou událost

Přebráno z referencí [1] a [3].

#### 2.2.3.2 SIP odpovědi

SIP odpovědi se liší od SIP žádostí v prvním řádku zprávy, kde se nachází tzv. *Status-Line*, která obsahuje verzi protokolu SIP (v dnešní době nejčastěji 2.0), kód odpovědi a popis odpovědi. Kód odpovědi se skládá z celého třímístného čísla v rozsahu od 100 až po 699. Je definováno 6 tříd odpovědí, jež se rozlišují svým druhem podle počátečního čísla. Následuje seznam tříd odpovědí s příklady kódů a popisu odpovědí. Seznam byl přebrán ze zdrojů [3], [1].

- **1xx:** dočasné informativní odpovědi
  - 180 Ringing - vyzvánění
  - 182 Queued - přidán do fronty
- **2xx:** úspěšné pozitivní odpovědi, akce byla přijata a potvrzena
  - 200 OK - žádost byla úspěšná
  - 202 Accepted - žádost byla přijata, ale ještě nedošlo k jejímu splnění
- **3xx:** přesměrování - informace o uživatelově nové lokaci nebo alternativním poskytovaným službám
  - 301 Moved Permanently - uživatel nebyl nalezen pod zažádaným URI
  - 305 Use Proxy - zažádaný prostředek je dostupný pouze přes proxy (uvedeném v kontaktním poli)
- **4xx:** problém se žádostí - chyba na straně klienta
  - 400 Bad Request - žádosti nebylo porozuměno kvůli špatné syntaxi
  - 404 Not Found - uživatel nebyl nalezen
  - 480 Temporarily Unavailable - volaného se podařilo kontaktovat, ale není momentálně dostupný (např. není přihlášen, je zapnuta možnost "Nerušit")
- **5xx:** problém na straně serveru - server selhal při zpracování žádosti, i když je žádost pravděpodobně v pořádku
  - 501 Not Implemented - server nepodporuje funkcionalitu pro splnění žádosti
  - 503 Service Unavailable - server momentálně nemůže splnit žádost kvůli probíhající údržbě nebo dočasnému přetížení
- **6xx:** globální chyba - žádost nemůže být splněna na žádném serveru
  - 600 Busy Everywhere - volaného se podařilo kontaktovat, ale je momentálně nedostupný a hovor nepřijme
  - 604 Does Not Exist Anywhere - server má autoritativní informaci, že volaný s číslem URI z žádosti neexistuje

### 2.2.4 RTP

RTP (Real-time Transport Protocol) je protokol transportní vrstvy, který využívá transportní protokol UDP. Definuje, jaký formát mají pakety pro přenos zvuku nebo videa. Zajišťuje seřazení zaslaných paketů a jejich časové značkovaní. Samotný přenos audia a videa v RTP bývá doplněn o kontrolní protokol RTCP (RTP Control Protocol), který nese statistické informace o průběhu přenosu.[1]

### 2.2.5 SDP

SDP (Session Description Protocol) je protokol, který slouží k popisu relace mezi dvěma (nebo více) koncovými zařízeními. Je často implementován společně se SIP. SDP se skládá z řádků obsahujících text ve formě položek typ a hodnota, ty jsou odděleny rovnítkem. Položky se dělí na popis relace (Session description), popis časování (Time description) a popis médií (Media description). Všechny povinné atributy jsou dále doplněny řadou nepovinných atributů.[1]

## 2.3 Podvody a útoky v prostředí IP telefonie

Stejně jako i v jiných odvětvích informačních a komunikačních technologií i v IP telefonii dochází k nejrůznějším zákeřným aktivitám. Jedná se o útoky na VoIP infrastrukturu, její špatnou konfiguraci, nebo využití bezpečnostních chyb, ale i podvody, jež využívají neznalosti, nepozornosti, či jen dobré vůle koncových uživatelů. Tato podkapitola se právě zaměřuje na popis takových útoků a podvodů, ke kterým v současné IP telefonii dochází. Největší část se věnuje podvodu Wangiri, protože na tento typ podvodu se zaměřuje praktická část této diplomové práce.

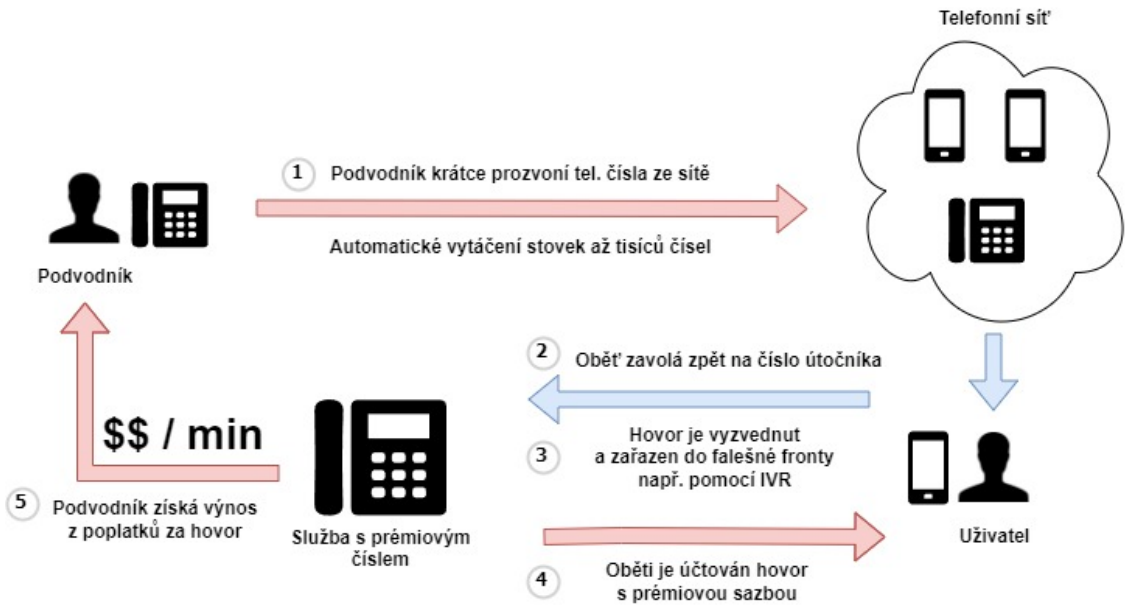
### 2.3.1 Wangiri podvod

Wangiri (z japonštiny doslova "Zazvonit a položit") je druh podvodného volání. První pokus o takový podvod se stal v Japonsku v roce 2002.[5]. Útočník krátce prozvoní volaného. Příjemce hovoru mylně předpokládá, že zmeškal legitimní hovor a zavolá zpět na číslo útočníka. Čísla jsou zpravidla z různých cizích až exotický zemí (Nigérie, Myanmar, Papua Nová Guinea, ale i např. Španělsko, Švýcarsko apod.) a přesměrují oběť útoku na mezinárodní číslo s prémiovou sazbou (IPRN - International Premium Rate Number) nebo jiné číslo účtované za vysoké poplatky, kde je oběť podvodu vyzvána, aby vyčkala, než dojde ke spojení. Podvodné hovory lze uskutečnit odkudkoli na světě, ale zpravidla pocházejí ze zemí, které jsou vzdálené od domovské sítě volaného účastníka z nepřidělených čísel nebo z čísel vyhrazených pro satelitní telefonní operátory. Ke spojení s podvodníkem samozřejmě nikdy nedojde, a to má za následek dlouhý čas strávený podvedeným ve frontě na zpoplatněné mezinárodní lince. Útočník pak získá výnos z těchto prémiových poplatků.[5][6]

Tyto útoky často provádějí koordinované týmy útočníků, jež si mezi sebou rozdělují výdělek. Pro útoky využívají technologie automatického vytáčení (angl. autodialing). Ta dokáže zahájit až

tisíce hovorů za minutu. Jedná část týmu provádí volání na číslo potenciální oběti. Druhá část týmu provozuje již dříve zmíněnou prémiovou linku, která přijímá hovory od původně volaného. Autodialing provádí velmi krátké hovory, které často negenerují záznam o provedení hovoru (Call Detail Records - CDR), což zvyšuje utajení útoků. Útočníkům stačí malý počet obětí, které zavolají zpět, aby vygenerovali zisk tisíců dolarů za jediný den. Následující den vytvoří nové číslo i seznam potenciálních obětí. Tyto mezinárodní týmy útočníků si pak vygenerované příjmy rozdělí.[5][6] Průběh Wangiri útoku je znázorněn na obrázku 2.4.

Získávat informace o provádění útoku Wangiri je pro poskytovatele telefonních služeb velmi složité. Největším problémem je detekce takového útoku co nejdříve, a to ještě v době, kdy k útoku dochází. Operátor může monitorovat příchozí hovory z podezřelých destinací nebo telefonních čísel, ale to není příliš efektivní, vzhledem k množství států a provedených hovorů zároveň značně nepraktické.[7]




Obrázek 2.4: Schéma útoku Wangiri

### 2.3.2 Vishing

Jedná se o spojení anglických slov Voice phishing (hlasový phishing). Je to telefonní podvod, kdy podvodník volá klientům banky a vydává se za jejího zaměstnance. Oznamuje klientovi (oběti podvodu), že byl jeho bankovní účet napaden a může mu v této situaci pomoci. Podvodník následně žádá klientovy informace o platební kartě (CVV/CVC, číslo kreditní karty, její platnost a jméno) nebo přímo přihlašovací údaje k internetovému bankovnictví. Pokud se mu podaří oběť přesvědčit, aby mu zmíněné informace předala, zpravidla převede všechny peníze z účtu na svůj, nebo používá ukradenou platební kartu k placení.[8]

### 2.3.3 Subscription fraud - podvod s předplatným

Při podvodu s předplatným využívá podvodník zpoplatněné telekomunikační služby bez úmyslu zaplatit poplatky telekomunikačním operátorům. Existuje několik variant tohoto podvodu. Jednou z nich je používání služby podvodníkem pod ukradenou identitou, kdy podvodník prostřednictvím sociálního inženýrství získá údaje oběti (jméno, číslo, přihlašovací údaje do aplikace) a používá zpoplatněné služby s identitou oběti. Další z variant podvodu s předplatným je prodej služeb pod takto ukradenou identitou. Podvodník tak trží zisk za takto prodané služby.[5]

### 2.3.4 Počítačové viry

Některé prvky VoIP sítě (počítače, servery) jsou připojeny k internetu. Pokud je takový počítač napaden počítačovým virem, může dojít k infiltraci VoIP sítě. Jedná se o trojské koně, kteří získají soukromá data (přihlašovací údaje, záznamy z hovorům, konfigurace sítě) uložená na zařízení. Tato data jsou pak použita pro získání přístupu do VoIP sítě a dále využívána k odposlouchávání, přesměrovávání a jiným zákeřným praktikám (Man in the Middle attack).

### 2.3.5 Flooding

Jedná se o útok typu DoS (Denial of Service). Prvek ve VoIP síti je (často SIP proxy server nebo samotná ústředna) zahlcen velkým množstvím SIP zpráv nebo požadavků. Většinou se jedná o zprávy typu REGISTER (SIP Register Flooding), INVITE (Call Flooding Attack), BYE nebo velké UDP datagramy (posílané přímo na port 5060). Ty jsou najednou ve velkém množství útočníkem zasílány na VoIP prvek. Při zahlcení dochází prvku prostředky, kvalita služby se snižuje, a nakonec může dojít až k jejímu úplnému výpadku. [9][10]

### 2.3.6 Man in the middle attack

Tento útok je specifický tím, že útočník infiltruje komunikaci mezi dvěma nebo více účastníky. Odposlouchává a mění směr toku dat mezi nimi, a to bez jejich vědomí. Záleží také na pozici útočníka v síti, kdy se v jednom případě může prezentovat jako falešná SIP proxy, v druhém případě může být umístěn mezi SIP Proxy a uživatelem, v dalším mezi uživatelem a důležitým serverem, který VoIP infrastruktuře poskytuje doplňující služby (DHCP, DNS, TFTP). V současnosti se pro realizaci Man in the middle útoku nejvíce používá metoda zvaná ARP Poisoning. Protokol ARP mapuje MAC adresy na IP adresy. Některé operační systémy nahradí, či přijmou záznam do své ARP cache bez ohledu na to, zda dříve poslaly či neposlaly ARP požadavek. To znamená, že případný útočník může oklamat jednu či obě účastnické strany tak, aby si myslely, že útočníkova MAC adresa je adresa druhého počítače či SIP serveru. Tím se stává prostředníkem mezi účastníky a může odposlouchávat či modifikovat veškerou komunikaci mezi nimi.[10]

### 2.3.7 Over-The-Top Bypass

Over-The-Top Bypass (dále jen OTT). Telefonní hovor je přesměrován přes IP na voice-chat aplikaci na chytrém telefonu, místo aby byl veden přes běžnou telekomunikační infrastrukturu. Tato přesměrování jsou prováděna společně s poskytovatelem služeb OTT, ale bez výslovného povolení od volajícího, volaného či jejich operátorů. Tím shromažďují část podílů za provedený hovor a způsobují značnou ztrátu obcházeným operátorů. Navíc tato praxe degraduje kvalitu služeb. [11]

### 2.3.8 Voice spam - hlasový spam

Hlasový spam označuje hromadná, automaticky generovaná, nevyžádaná volání. Hlasový spam je nejvíce využíván v telemarketingu. Vyznačuje se velkou frekvencí, s níž jsou hovory vytvářeny. Vetšinou je využito již dříve zmíněného autodialingu. Autodialer vytáčí náhodná čísla a snaží se najít člověka, který hovor přijme. Když člověk odpoví a je identifikován, hovor je předán jinému člověku. Ten následně zahájí prodejní nabídku.[4]

### 2.3.9 PBX hacking

Druh útoku, při kterém se snaží útočník získat nepovolený přístup na pobočkovou telefonní ústřednu PBX. Využívá k tomu různých slabin v implementaci systému, slabá hesla uživatelů nebo metody sociálního inženýrství. Po získání přístupu na ústřednu má prakticky plnou nadvládu nad celým systémem. Často dochází k přesměrovávání provozu přes prémiovou linku útočníka. Ten využívá poplatků za svou mezinárodní prémiovou linku ke svému obohacení.[12]

### 2.3.10 Spoofing telefonního čísla

Protože jsou řídicí zprávy SIP ve většině případů zasílány jako prostý text, jsou náchylné ke spoofingu (vydávání se za někoho jiného), úpravám nebo zachycení. SIP v podstatě umožňuje jakýkoli požadavek zpracovávat bez ověření a nevynucuje mechanismy ověřování zdrojových zpráv SIP. Služba SIP autentizace je volitelná a systém ji umožňuje, aby byl požadavek SIP zpracován bez ověření. Toho může útočník využít a vydávat se za jiného uživatele v síti. Uživateli nebude umožněno přijímat hovory, protože veškerý provoz bude zasílán na prvek útočníka.[13]

Spoofing telefonního čísla nebo použití anonymního telefonního čísla je dnes velmi snadné a lze jej provést různými způsoby. Například pomocí IP PBX, služeb založených na VoIP, nebo dokonce aplikací na chytrých telefonech.[4]

## 2.4 Shrnutí

Podkapitola 2.3 byla věnovaná nejčastějším podvodům a útokům v prostředí IP i standardní telefonie. V roce 2021 dle údajů ze studie CFCA (Asociace pro kontrolu podvodů v komunikacích) čítaly

globální ztráty zapříčiněné telekomunikačními podvody 39,89 miliard amerických dolarů.[14] Jedná se tedy o velmi rozšířený druh zákeřné a často i protiprávní činnosti. [14]

Vzhledem k stále rostoucímu využití a poptávce po telekomunikačních službách, roste i počet případů, kdy dochází k podvodnému jednání. Mezi lety 2019 a 2021 se ztráty tvořené telekomunikačními podvody zvýšily o 28 %. Tato skutečnost je zapříčiněná i pandemií Covid-19, které podvodníci zneužili zejména přes metody využívající sociální inženýrství cílené na důvěřivé uživatele telekomunikací.[14]

Jeden z takových podvodů je i Wangiri. Dle dříve zmíněné studie CFCA se Wangiri umístil na druhém místě mezi podvodnými metodami ve srovnání způsobených ztrát s ročními celosvětovými ztrátami až 2,23 miliardy USD, což v roce 2021 tvořilo zhruba 6 % všech ztrát v telekomunikačním průmyslu. [14]

Wangiri podvod je v dnešní době velmi relevantní. Z teoretického hlediska lze vytvořit prostředí k provedení podvodů Wangiri relativně snadno a s malými nároky na znalosti podvodníka. Zároveň je výdělečný, a jak už bylo uvedeno, stává se čím dál více rozšířenějším druhem zákeřného jednání v prostředí telekomunikací. Protože se jedná o aktuální téma, avšak často opomíjené, bude na něj zaměřen zbytek této diplomové práce.

# Kapitola 3

# Nástroje pro realizaci Wangiri podvodu

Pro realizaci útoku Wangiri je potřeba nástrojů, které umožní provádět útok co nejjednodušeji, pokud možno automatizovaně, a zároveň na co nejvíce subjektů. Toho lze docílit vybráním vhodné VoIP telefonní ústředny, generátoru hovorů a také způsobu, kterým budou čísla, na něž bude útok prováděn, vybírána. Tato kapitola obsahuje některé dostupné prostředky, jež byly zvažovány, anebo nakonec vybrány pro implementaci praktické části diplomové práce. Vybrané nástroje a konečná topologie jsou odůvodněny v další kapitole.

## 3.1 Autonomní vyhledávače záznamů

Autonomní vyhledávač záznamů slouží k automatizovanému získávání určité části dat. Pokud vyhledávač vyhledává data z webových stránek, jedná se o tzv. web scraper. Pro potřeby diplomové práce je tato část zaměřena na vyhledávače telefonních čísel, především na ty, které jsou schopny vyhledávat čísla z více webových stránek najednou.

### 3.1.1 Contact Details Scraper

Contact Details Scraper je autonomní vyhledávač kontaktních informací z internetových stránek. Dokáže získat e-maily, telefonní čísla i adresy profilů na sociálních sítích. Umožňuje vyhledávat informace z více URL najednou, výsledky ukládá ve formátu JSON, HTML, CSV nebo XML. Je zdarma.[15]

### 3.1.2 Phone Extractor For Web Pages and Text

Tento nástroj umožňuje vyhledávat telefonní čísla z webových stránek bez nutnosti stahování softwaru. Po vložení souboru, nebo zadání URL adresy však vypíše všechna čísla, která se na stránce nachází, i když se nejedná o telefonní číslo. Navíc umožňuje vyhledávání pouze z jednoho zdroje najednou.[16]

### 3.1.3 Scrape Box - Phone Number Scraper

Tento nástroj je součástí balíku jiných autonomních vyhledávačů Scrape Box. Umožňuje vyhledávat z více URL najednou i díky podpoře multithreadingu. Obsahuje možnost nastavení přesného formátu telefonního čísla, podle kterého je chce uživatel vyhledat. Baliček Scrape Box je však placený. [17]

### 3.1.4 Phone Number Extractor files

Phone Number Extractor Files (dříve veden jako Files Phone Number Grabber) je nástroj pro extrakci telefonních čísel z dokumentů formátu DOC, PDF, TXT, PPTX. Umožňuje vyhledávání čísel z více dokumentů najednou. Vyhledaná čísla ukládá do souboru ve formátech CSV nebo TXT. Nejedná se však o freeware, ale placený software. Nevýhodou pro účely práce je skutečnost, že tento vyhledávač záznamů neumožňuje vyhledávat záznamy přímo z webových stránek.[18]

### 3.1.5 Phone Number Web Extractor

Produkt společnosti Technocom Solutions. Nástroj pro vyhledávání telefonních nebo faxových čísel přes populární internetové vyhledávače (Google, Bing, Yahoo). Je placený, ale existuje i bezplatná zkušební verze. Lze vyhledávat z více URL najednou. Stejně jako většina předchozích web scraperů ukládá výsledky do souboru typu CSV a TXT. [19]

### 3.1.6 Phone Number Grabber

Phone Number Grabber je rovněž placený. Umožňuje ukládat výsledky pouze do TXT a CSV souborů. Svou funkcionalitou a různými prvky je velmi podobný dvěma předcházejícím web scraperům.[20]

## 3.2 Generátory provozu

### 3.2.1 SIPp

SIPp je bezplatný open source testovací nástroj a generátor SIP relací. Obsahuje UAC a UAS, poskytuje statistiky o zprávách v SIP signalizaci, umožňuje rovněž emulovat i média. SIPp pracuje s audiem či videem ve formě pcap. Statistiky, které jsou aplikací SIPp vytvářeny v reálném čase, poskytují informace o počtu sestavených volání, round trip delay a statistiky jednotlivých SIP zpráv. Lze vytvářet vlastní scénáře, což dává SIPp rozsáhlé možnosti, lze také generovat více SIP relací na vzdálený systém najednou. SIPp nemá vlastní grafické uživatelské rozhraní, veškerá konfigurace i spouštění je prováděno přes příkazový řádek.[21][1]

SIPp může být využit k testování reálných SIP zařízení jako SIP proxy (např. Kamailio), B2BUA (např. Asterisk), SIP media servery, SIP brány and SIP PBX. Nástroj může být využit k emulování

tisíců UA generujících relace do cílového zařízení. Kromě těchto výkonnostních testů lze využít SIPp i k testování interoperability, čili k ověření chování neznámého zařízení.[21][1]

### 3.2.2 SIP Loader

Jedná se o generátor SIP provozu. Umožňuje testovat telefony, koncové body, servery, PBX, VoIP brány a testovat zatížení a interakci funkcí pro zvuk a video, průběžné testování trasy po síti. Automatizuje deterministické ověřování tras a telefonních čísel napříč sítěmi VoIP, šetří čas a zvyšuje efektivitu a pokrytí testování. V nástroji jsou nastavena pravidla pro procházení více síťových cílů a telefonních čísel podle plánu testování. Poskytuje zprávy o přijatých médiích, době spojení hovoru, délce hovoru, jitteru, ztrátě paketů. Ověřuje, zda zařízení funguje a je nakonfigurováno správně. Pomáhá testovat VoLTE v síti IMS, kde je priorita paketů RTP a tísňových volání kritická.[22]

Obsahuje grafické uživatelské rozhraní. Uživatelské příkazy lze automatizovat přes rozhraní REST API nebo Python. Existuje pouze placená verze aplikace.[22]

### 3.2.3 StarTrinity SIP Tester

StarTrinity SIP Tester je nástroj pro generování provozu a testování VoIP sítě nebo SIP hardware a software. Je schopen simulovat a pasivně monitorovat tisíce současných příchozích a odchozích SIP hovorů s RTP médii, analyzovat kvalitu hovorů a vytvářet reporty v reálném čase. Obsahuje grafické uživatelské rozhraní. SIP Tester simuluje aplikační server, mediální server, SIP telefon nebo registrační server. Existuje ve více verzích. Freeware licence povoluje až 50 hovorů najednou. Komerční placená licence toto číslo ještě zvyšuje. [23]

Tok hovorů je specifikován CallXML skriptem, kde je možné navrhnout různé situace, které mohou způsobit selhání testovaného SIP stacku. Software je vhodný pro IP PBX emulaci, SIP DoS simulaci, testování výkonu ústředny, IVR testování. Lze jeho prostřednictvím i hovory nahrávat, provádět penetrační testování, sledovat kvalitu hovoru či diagnostikovat VoIP síť.[23]

## 3.3 VoIP ústředny

### 3.3.1 Asterisk

Asterisk je softwarový volně dostupný framework pro implementaci komunikačních aplikací. Je především softwarovou pobočkovou ústřednou určenou pro instalaci na standardních PC a spolu se správným rozhraním může být použit jako pobočková ústředna pro domácí uživatele, podniky, poskytovatele VoIP služeb a telefonní společnosti. Asterisk je rovněž open-source komunita a komerční produkt od firmy Digium.[1][24][25]

Asterisk systém je navržen tak, aby vytvořil rozhraní mezi telefonním hardwarem či softwarem a libovolnou telefonní aplikací. S jednotlivými protokoly SIP, IAX, H.323 pracuje Asterisk jako

s kanály navázanými na jádro, DAHDI (Digium Hardware Device Interface) představuje rozhraní směrem k PSTN. Velmi důležitým nástrojem je příkazový řádek CLI (Command Line Interface), který zprostředkovává konfiguraci. Srdcem Asterisku je Dialplan, kde je definováno chování v případě obsluhy požadavku, ať už odchozího či příchozího volání anebo požadavku na vyvolání služby.[1][24]

Kromě funkce pobočkové ústředny může fungovat např. jako packet voice server, IVR server, voicemail služba s adresářem nebo VoIP gateway. Dále podporuje různé služby: fronty volajících, integrace rozpoznávání hlasu, propojení s PSTN skrze digitální i analogové linky atd.[24]

### 3.3.2 3CX

3CX je PBX telefonní systém od společnosti 3CX. Umožňuje provádět videohovory, webové konference nebo fronty hovorů. Lze do něj integrovat textovou komunikaci, komunikaci přes sociální sítě. Je možné využívat mobilní aplikaci i grafické uživatelské rozhraní ve webovém prohlížeči. Je to také otevřená platforma, plně kompatibilní s IP telefony a SIP trunky. Lze jej provozovat na serveru, virtuálním počítači nebo v privátním cloudu. Existuje jak komerční, tak bezplatná verze, která je však omezena pouze na 10 uživatelů s omezenými poskytovanými službami.[2]

### 3.3.3 Kamailio

Kamailio je volně dostupný open source SIP proxy server. Dokáže obsluhovat tisíce spojení za sekundu. Kamailio lze použít k vytvoření rozsáhlých platforem pro VoIP komunikaci, WebRTC, Instant messaging a jiné aplikace. Kromě toho jej lze snadno použít pro škálování SIP-to-PSTN bran, systémů využívajících pobočkové ústředny nebo mediálních serverů, jako jsou Asterisk nebo FreeSWITCH.[26]

Mezi funkce Kamailia patří např: asynchronní TCP, UDP a SCTP, zabezpečená komunikace přes TLS pro VoIP hlas, video i text. Podpora IPv4 a IPv6, asynchronních operací, rozšíření IMS pro VoLTE, vyvažování zátěže (load balancing), účetnictví, autentizace a autorizace. Podporuje také mnoho backendových systémů, jako jsou MySQL, Postgres, Oracle, Radius, LDAP, Redis, Cassandra, MongoDB, Memcached; řídicí rozhraní Json a XMLRPC, monitorování přes protokol SNMP.[26]

# Kapitola 4

# Návrh systému pro generování podvodných hovorů

Tato kapitola se věnuje teoretickému návrhu systému pro generování podvodných hovorů Wangiri. Vzhledem k zákeřnému charakteru podvodu Wangiri je celý systém implementován v laboratorním prostředí a pro testovací prostředí následně upraven.

## 4.1 Návrh topologie systému

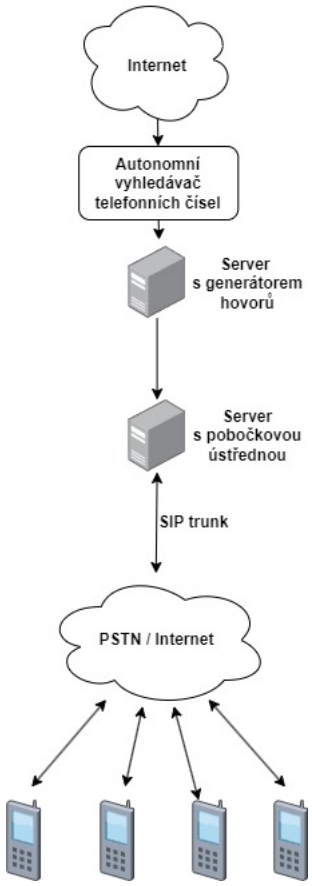
Systém pro generování podvodných volání Wangiri by se měl skládat z několika částí.

Pro takový systém je nutné zajistit způsob, jak získat telefonní čísla, která bude prozvánět, tedy stanou se potenciálními oběťmi podvodu. Dále způsob, kterým budou čísla prozvánět co nejefektivněji, což znamená, co nejvíce čísel v co nejkratším čase. Také je třeba vytvořit linku, na níž bude oběť útoku volat zpět. Tato linka by měla generovat příjem účtováním za hovor s prémiovou sazbou, nebo bude na linku s touto službou přepojovat. To také znamená, že by měl být součástí řešení spojení do veřejné telefonní sítě (PSTN). Na lince s prémiovou službou je žádoucí způsob, jak oběť podvodu zdržet na hovoru co nejdéle. Pro tento účel nejlépe slouží interaktivní hlasová odezva (IVR) a v ní vytvořená umělá fronta. Pobočková ústředna je propojená s internetem přes SIP trunk nebo přímo do veřejné telefonní sítě. Jeho prostřednictvím dochází k podvodným prozvoněním.

Kvůli lepšímu rozložení prostředků navrhuji systém, ve kterém se nachází dva oddělené servery. Na prvním z nich je provozován generátor podvodných hovorů a autonomní vyhledávač telefonních čísel. Na druhém serveru je provozována pobočková ústředna, přes níž jsou provozovány odchozí hovory a přijímány příchozí hovory.

Celé řešení by mělo být vytvořeno tak, aby náklady byly co nejnižší, to znamená použití open-source řešení, nebo volně dostupný software, pokud to bude možné. Samozřejmé je nutné počítat s cenou provozu linky s prémiovou sazbou, a také zpoplatnění SIP trunku do internetu nebo veřejné

telefonní sítě. Tyto náklady by měly být vyváženy ziskem z poplatků za hovory přijaté přes linku s prémiovou sazbou.




Obrázek 4.1: Schéma návrhu topologie systému

Následuje výběr technologií pro jednotlivé části topologie. Vzhledem k charakteru navrhované topologie jsem jako operační systém pro oba servery, na níž jsou provozovány služby pro provádění podvodu Wangiri, vybral Linux. Jedná se o volně dostupný, open-source operační systém. Je velmi dobře zdokumentován, a pro řadu aplikací je dokonce vhodnější než jiné operační systémy.

## 4.2 Autonomní vyhledávač telefonních záznamů

Pro efektivní získávání telefonních čísel je vhodné použití metody web scraping, což je metoda extrakce dat z webových stránek, především z jejich HTML nebo XHTML kódu.

Při porovnávání již existujících řešení z podkapitoly 3.1 jsem všechny zmíněné řešení označil jako nepříliš vhodné pro účely této práce.

Contact Details Scraper nevyhledává pouze telefonní čísla, ale zároveň i jiné informace ze stránek jako jsou odkazy na LinkedIn profil, e-maily ap. Všechna tato data kromě telefonních čísel jsou pro systém irelevantní a pouze by bylo nutné opět vyfiltrovat z výsledného souboru jen telefonní čísla.

Phone Extractor For Web Pages and Text funguje pouze ve webovém prohlížeči. Veškerá data telefonních čísel by musela být z každé stránky zpracovávána manuálně. Navíc, jelikož servery navrhovaného systému neobsahují grafické uživatelské rozhraní, není použití tohoto řešení možné.

Scrape box, stejně jako předchozí řešení, je ovládán pouze přes grafické uživatelské rozhraní. Navíc je zpoplatněn. Z navrhovaných autonomních vyhledávačů se zdá jako nejvhodnější, ovšem skutečnost, že je zpoplatněn, toto řešení diskvalifikuje pro použití v této diplomové práci.

Phone Number Extractor Files, Phone Number Web Extractor i Phone Number Grabber jsou zpoplatněny. Dalším faktorem je nedůvěryhodnost stránek, na kterých jsou tato řešení prezentována. Nejsou zde přítomny žádné reference na firmu, jejich sídlo či dokonce telefonní čísla firem. Pouze u Phone Number Grabber je uvedeno telefonní číslo s předvolbou státu Pákistán. Jejich popisy jsou si velmi podobné. Ze všech těchto důvodů jsem všechny tyto web scrapery označil za nevhodné.

Po analýze uvedených nástrojů jsem se rozhodl, že vytvořím svou vlastní implementaci autonomního vyhledávače telefonních čísel. Ten by měl rozpoznat telefonní čísla ze stránky a následně je vhodně uložit do požadovaného formátu tak, aby mohl auto-dialer tato čísla z vytvořeného souboru použít pro generovaní hovorů. Vytvoření vlastního nástroje umožňuje upravit web scraper podle potřeb jak generátoru hovorů, tak celého navrženého systému.

## 4.3 Generátor hovorů

Další částí navrhovaného systému je generování hovorů pro prozvonění potenciálních obětí Wangiri podvodu. Vhodný generátor hovorů by měl být snadno konfigurovatelný, rychlý a schopný spolupracovat s vybranou pobočkovou ústřednou. Porovnával jsem generátory SIP relací z podkapitoly 3.2.

SIP Loader je součástí balíku několika aplikací. To je značně nevýhodné, protože pro účely navrhovaného systému potřebujeme pouze generátor hovorů (SIP relací). Dalším negativním aspektem je ovládání aplikace přes grafické uživatelské rozhraní. Posledním velkým negativem je skutečnost, že se jedná o placenou aplikaci.

Stejně jako SIP Loader je i StarTrinity SIP Tester placená aplikace ovládaná prostřednictvím grafického uživatelského rozhraní. Oproti SIP Loaderu se ovšem jedná pouze o VoIP monitorovací nástroj, takže z tohoto ohledu je pro implementaci řešení vhodnější.

SIPp je jediným porovnávaným nástrojem, který je bezplatný. Navíc je i jeho kód volně dostupný a lze jej stáhnout např. z GitHubu. Zároveň je i v době tvorby této diplomové práce stále aktualizován.

Po druhotné analýze všech uvedených generátorů provozu jsem pro navržený systém zvolil volně dostupný testovací nástroj a generátor SIPp. Jako jeho hlavní výhody bych vyzdvihl volnou dostup-

nost a dobrou dokumentaci na webových stránkách projektu a GitHubu. Další vhodnou vlastností je absence grafického uživatelského rozhraní, které není nutné pro potřeby automatického vytáčení telefonních čísel.

## 4.4 Telefonní ústředna

Důležitým prvkem v topologii navrhovaného systému je pobočková ústředna sloužící k provádění prozvánění a přijímání příchozích hovorů. Porovnával jsem nástroje uvedené v části 3.3.

3CX je komplexní aplikace pro tvorbu komerčních telefonních systémů. Jejím největším negativem je bezplatná verze, kde je možné vytvořit pouze 10 uživatelů a navíc s omezenými dostupnými službami.

Asterisk je v dnešní době velmi populární. Má velkou komunitu a je volně dostupný. Jeho přednostmi jsou kvalitní dokumentace, podpora, množství poskytovaných služeb. Oproti Kamailia se jedná o Back to Back User Agenta.

Kamailio je také volně dostupné s otevřeným kódem. Jedná se o SIP proxy server, ale může také sloužit jako registrar nebo pobočková ústředna. Oproti Asterisku je vhodnější pro velké infrastruktury v řádech deseti až sta tisíců prvků. Na rozdíl od Asterisku je konfigurace řešena skriptem, které jsou implementovány v jazyce C. Od novějších verzí však lze využít i Python, Lua nebo JavaScript. I přes to bych to hodnotil jako nevýhodu, protože je pro obsluhu Kamailia nutná znalost programovacích či skriptovacích jazyků.

Na provádění hovorů a jejich příjem jsem se rozhodl použít pobočkovou ústřednu (PBX) Asterisk. Zvolil jsem ji kvůli její dobré dokumentaci, rozšířené komunitě a dostupnosti. Také s ní mám zkušenosti z předmětů absolvovaných během studia (VoIP, Komunikační systémy v podnikových sítích). Oproti Kamailiu je z mého subjektivního hlediska uživatelsky více přívětivý a jeho konfigurace méně náročná, zvláště kvůli nutnosti konfigurace Kamailia ve výše uvedených skriptovacích jazycích.

## 4.5 SIP trunk

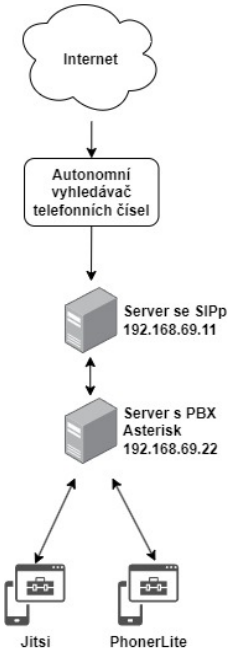
Poslední částí navržené topologie je SIP trunk do internetu nebo veřejné telefonní sítě. K jeho vytvoření je nutné si vyjednat u poskytovatele SIP trunk. Zároveň by bylo vhodné si u stejného poskytovatele zařídit linku s prémiovou sazbou pro generování zisku. Zpravidla takové služby ovšem nejsou bezplatné. Část zisku generovaného z linky s prémiovou sazbou si účtuje poskytovatel. Zároveň služby SIP trunku jsou zpoplatněny měsíčním paušálem.

Problém by mohl nastat při provádění Wangiri podvodu, kdy by poskytovatel SIP trunku a čísla s prémiovou sazbou byl schopen vypozorovat podezřelou aktivitu systému. A to např. velikým počtem odchozích hovorů za krátkou dobu. To je ovšem okolnost, se kterou podvodník musí počítat, jelikož se jedná o zákeřnou činnost spojenou s prvky sociálního inženýrství.

# Kapitola 5

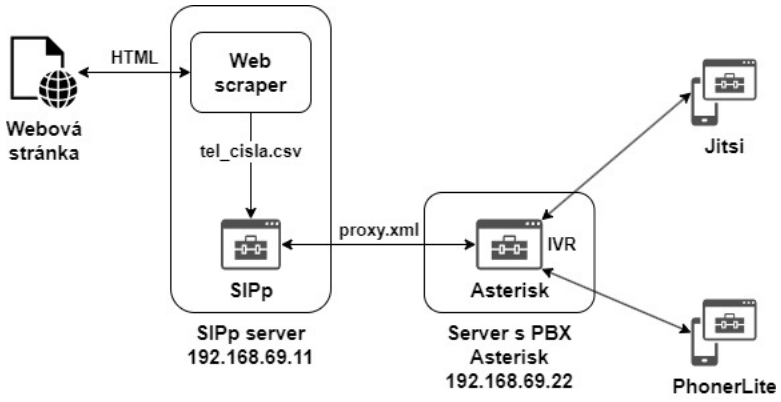
# Implementace řešení pro generování podvodných hovorů

Jak už bylo zmíněno v kapitole 4, celé řešení bylo implementováno v laboratorním prostředí. Z tohoto důvodu jsou v implementaci provedeny některé změny. Není zde přítomný žádný SIP trunk do veřejné telefonní sítě nebo internetu. Pro lepší přehlednost konfigurace Asterisk ústředny byla telefonní čísla zaměněna na čtyřciferné klapky. Na Asterisk ústředně je vytvořen účet pro klapku útočníka, účet simulující linku s prémiovou sazbou a účty s čtyřcifernými čísly jako klapky, která mají simulovat běžného uživatele telefonní nebo VoIP sítě. Schéma implementace upraveného návrhu topologie v laboratorním prostředí je zobrazeno na obrázku 5.1.




Obrázek 5.1: Schéma implementace systému

Implementovaný systém pro provádění útoků typu Wangiri je provozován na dvou virtuálních počítačích. Oba virtuální počítače jsou vytvořeny v hypervizoru Oracle VM Virtualbox, jako jejich operační systém je použit Debian 11. Na prvním virtuálním počítači je provozován autonomní vyhledávač telefonních záznamů (web scraper) implementovaný v jazyce Python, který zpracovává telefonní čísla z webových stránek a ukládá je do souboru formátu *.csv* pro generátor provozu SIPp. SIPp vytvoří žádost o provedení hovoru na ústřednu, která je zprovozněna na druhém virtuálním počítači. Asterisk ústředna slouží jako VoIP brána. Přes bránu dochází k samotnému Wangiri podvodu, kdy jsou realizovány hovory na všechna čísla ze seznamu telefonních čísel získaných autonomním vyhledávačem. Pokud cíl podvodu zavolá zpět na číslo podvodníka, je přepojen na číslo s prémiovou sazbou, kde je spuštěno IVR (Interaktivní hlasová odezva) menu simulující frontu. Celý proces průběhu Wangiri podvodu v implementovaném prostředí je zobrazen na obrázku 5.2.




Obrázek 5.2: Průběh Wangiri podvodu

## 5.1 Autonomní vyhledávač telefonních záznamů

Na prvním virtuálním počítači je provozován autonomní vyhledávač telefonních záznamů (web scraper) implementovaný jako skript, napsaný v jazyce Python, který zpracovává telefonní čísla z HTML kódu webových stránek a ukládá je do souboru formátu .csv pro generátor provozu SIPp.

Pro implementaci web scraperu jsem si vybral programovací jazyk Python, protože se v dnešní době jedná o velmi oblíbenou a rozšířenou technologii. Je open source, velmi dobře zdokumentován a funguje na většině dnes používaných platformách. Pro vývoj jsem použil editor kódu Microsoft Visual Studio Code. V další části podkapitoly je uvedena celková implementace autonomního vyhledávače telefonních čísel.

Použité knihovny:

- **csv** - modul pro čtení a psaní dat v csv formátu
- **os** - modul pro operace spojené s operačním systémem

- **re** - modul pro práci s regulárními výrazy
- **requests** - modul pro tvorbu HTTP požadavků
- **BeautifulSoup** - modul pro manipulaci s daty z HTML a XML souborů
- **sys** - modul systémových parametrů a funkcí

Nejdříve je uloženo URL webové stránky do proměnné *url*. URL stránky je do skriptu načtena z konzole. Pokud není při spuštění skriptu URL uvedena, zahlásí chybu.

```
# Kontrola zda byla uvedena URL adresa
if len(sys.argv) < 2:
    print("Chyba: Zadejte URL!")
    sys.exit(1)

# Uložení URL z konzole
url = sys.argv[1]
```

Následně je pomocí HTTP požadavku *GET* získán HTTP kód stránky a ten je pak přes metodu modulu BeautifulSoup upraven pro lepší čitelnost a uložen do proměnné *soup*. Dochází zde také k ošetření připojení k internetu. Pokud se nepodaří ze stránky získat odpověď, je vyvolána výjimka a do konzole se vypíše chyba.

```
# Zaslání požadavku GET na webovou stránku, obsahuje výjimku pro chyby se spojením
try:
    response = requests.get(url)
    response.raise_for_status()
except requests.exceptions.RequestException as e:
    print(f"Chyba: Nepodařilo se připojit na {url}: {e}")
    sys.exit(1)

soup = BeautifulSoup(response.content, 'html.parser')
```

Telefonní čísla z webových stránek nebo klapky v rámci laboratorní verze jsou vyhledávány z HTML kódu regulárními výrazy. Jsou vytvořeny i regulární výrazy pro různé verze telefonních čísel. Vyhledávání českých i amerických čísel obsahuje verze s mezerami mezi čísly, pomlčkami mezi čísly i čísla s předvolbou a bez ní. Uživatel skriptu si může zakomentovat a odkomentovat části s regulárními výrazy podle požadovaných čísel.

```
# Hledání telefonních čísel v HTML kódu pomocí regulárních výrazů
phone_numbers = re.findall(r"\b\d{3}[-.\s]?\d{3}[-.\s]?\d{3}\b", str(soup))
```

```
# Regulární výraz pro česká čísla
phone_numbers = re.findall(r"\+420\s?\d{3}\s?\d{3}\s?\d{3}", str(soup))

# Regulární výraz pro čísla z USA
phone_numbers = re.findall(r"\b\d{3}[-.\s]?\d{3}[-.\s]?\d{4}\b", str(soup))
```

Vysvětlení regulárního výrazu pro česká čísla

```
"\+420\s?\d{3}\s?\d{3}\s?\d{3}"
```

- **\+420** - číslo musí začínat řetězcem znaků +420
- **\s?** - značí nepovinnou mezeru mezi trojčíslím
- **\d{3}** - 3 čísla za sebou, d značí skupinu charakterů pro čísla 0-9 a {3} znamená, že by měly být porovnány právě 3 výskyty předchozího znaku
- **\s?** - značí nepovinnou mezeru mezi trojčíslím
- **\d{3}** - opět musí následovat 3 číslice
- **\s?** - značí další nepovinnou mezeru
- **\d{3}** - poslední 3 číslice telefonního čísla

Pro laboratorní verzi implementace je regulární výraz relativně méně složitý. Po nalezení všech telefonních čísel nebo klapek, jsou z nich odstraněny všechny mezery, pomlčky (především u čísel z USA) či tečky.

```
# Regulární výraz pro klapky v laboratorním prostředí
phone_numbers = re.findall(r'\b\d{4}\b', str(soup))

phone_numbers = [re.sub(r'[-.\s]', '', number) for number in phone_numbers]
```

Zbývá už jen vyhledaná čísla uložit do formátu vhodného pro SIPp. Jsou implementovány dvě části. První část vytvoří soubor *tel_cisla.csv*, pokud ještě neexistuje, a napíše na jeho začátek řetězec *SEQUENTIAL*. Ten je nutný kvůli SIPp a udává, že data mají být v souboru přečtena v sekvenci.

```
# Kontrola, zda soubor s telefonními čisly již existuje
if not os.path.isfile('tel_cisla.csv'):
    # Pokud soubor ještě neexistuje, vytvoří ho a přidá "SEQUENTIAL" na začátek
    with open('tel_cisla.csv', 'w', encoding='utf-8') as csvfile:
        writer = csv.writer(csvfile, lineterminator='\n')
        writer.writerow(['SEQUENTIAL'])
```

Druhá část ukládání jen přidá nalezená čísla na konec souboru. A nakonec skript vypíše do konzole, kolik čísel bylo ze stránky nalezeno.

```
# Přidaní vyhledaných čísel na konec souboru
with open('tel_cisla.csv', 'a', encoding='utf-8') as csvfile:
    writer = csv.writer(csvfile, lineterminator='\n')
    for phone_number in phone_numbers:
       writer.writerow([phone_number])

print(f'Nalezeno {len(phone_numbers)} čísel na {url} a uloženo do souboru
    tel_cisla.csv')
```

### 5.1.1 Rozšíření webscraperu

Pro větší automatizaci získávání čísel by mohlo být řešení autonomního vyhledávače ještě rozšířeno o automatické vyhledávání webových stránek. Jelikož ovšem nebyla implementace vlastního řešení v původním obsahu zadání, není tato část součástí vypracování.

Pokud by se jednalo o implementaci reálného prostředí pro provádění podvodů Wangiri, lze značné množství telefonních čísel získat z internetových rejstříků firem, lidí nebo online telefonních seznamů (např. online Zlaté stránky).

## 5.2 SIPp

Další částí implementace je generátor hovorů a testovací nástroj SIPp.

Nejdříve je nutné stáhnout instalační balíček SIPp ze služby GitHub pomocí příkazu *wget*. Následovalo stažení dalších balíčků ke správné funkcionalitě instalace. Jedná se o balíky pro zprovoznění SIPp s podporou TLS, PCAP a balíčků pro spouštění makefile. Při instalaci se nepodařilo zprovoznit nejnovější verzi SIPp 3.7, pravděpodobně kvůli absenci kompilátoru C++ 11, který je právě od verze 3.7 používán pro kompilaci zdrojového kódu. Proto jsem použil o jednu verzi starší, tedy 3.6.1, s jejíž instalací již žádné problémy nenastaly.

```
wget https://github.com/SIPp/sipp/releases/download/v3.6.1/sipp-3.6.1.tar.gz
tar -xvzf sipp-3.6.1.tar.gz
apt update
apt install g++ libncurses-dev libssl-dev openssl libpcap-dev libnet-dev make
cmake libscpt-dev lksctp-tools-dev pkg-config gsl-devel
```

Po stažení všech balíčků zbývala kompilace a instalace SIPp přes dodaný makefile.

```
cd sipp-3.6.1/
make
make install
```

Jako jeden ze vstupních parametrů při spouštění SIPp je nutné uvést scénář ve formě XML souboru, podle kterého dojde ke generování SIP relace, případně požadovanému testu. Jako XML scénář jsem použil soubor *proxy.xml*, který mi byl poskytnut vedoucím práce panem doktorem Řezáčem. Součástí scénáře je několik částí pro vytvoření SIP požadavku nebo odpovědi. Jedná se o metody REGISTER, INVITE, ACK a BYE. Soubor *proxy.xml* lze prohlédnout v přílohách této práce.

Jedná se o SIP požadavek na vytvoření hovoru. V uvedeném scénáři se objevují neurčité části například (*local_ip, field0* apod.) Jedná se o SIPp klíčová slova, na jejíž místo SIPp doplní správné údaje. Některá mají svoji výchozí hodnotu, jiná doplňuje dle vstupních parametrů testu. V případě tohoto řešení, například ze souboru *tel_cisla.csv*, doplní čísla volaného do hlavičkové části *To: [field0]*, výchozí hodnota parametru *[transport]* je UDP atd.

Část souboru proxy.xml se SIP metodou INVITE:

```
 <send retrans="500">
<![CDATA[

    INVITE sip:[field0]@[remote_ip] SIP/2.0
    Via: SIP/2.0/[transport] [local_ip]:[local_port];branch=[branch]
    From: [service]  <sip:[service]@[remote_ip]>;tag=[call_number]
    To: [field0] <sip:[field0]@[remote_ip]>
    Call-ID: d///[call_id]
    CSeq: 3 INVITE
    User-Agent: Grandstream GXP2000 1.1.6.37
    Contact: <sip:[service]@[local_ip]:[local_port];transport=[transport]>
    Max-Forwards: 70
    Allow: INVITE,ACK,CANCEL,BYE,NOTIFY,REFER,OPTIONS,INFO,SUBSCRIBE,UPDATE,
    PRACK,MESSAGE
    Supported: replaces, timer, path
```

```
      Subject: Performance Test
      Content-Type: application/sdp
      Content-Length: [len]

      v=0
      o=[service] 8000 8001 IN IP[local_ip_type] [local_ip]
      s=SIP Call
      c=IN IP[media_ip_type] [media_ip]
      t=0 0
      m=audio [auto_media_port] RTP/AVP 0 8 3
      a=rtpmap:0 PCMU/8000
      a=rtpmap:8 PCMA/8000
      a=rtpmap:3 GSM/8000
      a=ptime:20
      a=sendrecv


]]>
</send>
```

Vysvětlení jednotlivých částí SIP žádosti.
Hlavička žádosti:

- **INVITE** - jedná se o metodu INVITE pro sestavení spojení, součástí je i URI a verze SIP
- **Via** - slouží k záznamu cesty žádosti
- **From** - identifikace volajícího
- **To** - identifikace volaného
- **Call-ID** - unikátní identifikátor volání
- **Cseq** - pořadové číslo žádosti
- **User-agent** - druh koncového zařízení
- **Contact** - IP adresa port, na níž volaný očekává odpovědi a žádosti volaného
- **Max Forwards** - maximální počet skoků (mezi proxy, routery apod.), než bude žádost zahozena
- **Allow** - druhy metod povolené v relaci

- **Supported** - služby podporované UA odesílatele
- **Subject** - popis relace
- **Content-type** - typ těla zprávy
- **Content Length** - délka těla zprávy v bytech

Tělo žádosti obsahuje položky protokolu SDP.

- **v** - verze protokolu
- **o** - označuje původ relace tzn. název uživatele, identifikační číslo relace a jeho adresu
- **s** - název relace
- **c** - informace o připojení koncového bodu
- **t** - časové údaje pro začátek a konec relace, v tomto případě je připojení bez přerušení
- **m** - druh média
- **a** - atributy médií (např. preferované kodeky)

## 5.3 Asterisk ústředna

Další částí řešení je instalace a konfigurace Asterisk ústředny. Jak už bylo zmíněno v části 5, v rámci laboratorního prostředí jsou na ní vytvořeny i účty pro zařízení simulující koncové uživatele (oběti útoku).

### 5.3.1 Instalace PBX Asterisk

Stejně jako u instalace SIPp je pro úspěšnou instalaci Asterisku žádoucí nejdříve stáhnout balíčky závislostí.

```
apt update
apt install git curl wget libnewt-dev libssl-dev libncurses5-dev subversion
libsqlite3-dev build-essential libjansson-dev libxml2-dev  uuid-dev
```

Následuje stažení archivu se soubory Asterisku.

```
wget http://downloads.asterisk.org/pub/telephony/asterisk/asterisk-20-current
.tar.gz
tar -xvzf asterisk-20-current.tar.gz
```

Instalace pokračuje kontrolou prerekvizit a výběrem Asterisk modulů, které uživatel potřebuje z nabídky *menuselect*. V případě této implementace byly nainstalovány moduly *Dialplan Functions*, *PBX Modules* a všechny balíčky *Core-Sounds* a *Extra Sound Packages*.

```
contrib/scripts/install_prereq install
./configure
make menuselect
```

Zbývá kompilace a instalace Asterisku podle vybraných kriterií.

```
make
make install
make config
```

### 5.3.2 Vytvoření uživatelů

Pro vytvoření klapek uživatelů na simulaci veřejné telefonní sítě byl použit skript vytvořený v jazyce Python. Na začátku je rozmezí čísel pro uživatelské účty. V dalším kroku je otevřen konfigurační soubor pjsip.conf, a do něj jsou pak cyklicky vkládány jednotlivé vlastnosti každého koncového zařízení, jeho logická adresa a autentizační informace.

```
#!/usr/bin/env python
start = 3000
end   = 3999

soubor = open('pjsip.conf', 'at')

for i in range(start, end):
   account=str("""
[%s]
type = endpoint
direct_media = no
transport = simpletrans
context = test
disallow = all
allow = ulaw
aors = %s
auth = auth%s

[%s]
```

```
type = aor
max_contacts = 1

[auth%s]
type=auth
auth_type=userpass
password=uas
username=%s
""") % (i,i,i,i,i,i)
   soubor.write(account)
soubor.close()
```

Dále jsou vytvořeny další dva účty. 2000 pro generování a přijímaní hovorů a 4000, který simuluje linku s prémiovou sazbou pro přesměrování hovorů. Část konfigurace type transport udává typ používaného protokolu. V tomto případě se jedná o protokol UDP povolený na všech rozhraních. Zbytek konfigurace obsahuje vlastnosti koncového zařízení např. kodek, dialplan režim pro příchozí relace (*context*), nastavení autentizace a počet adres, na kterých může být koncový bod kontaktován (*max_contacts*).

Část konfiguračního soubor pjsip.conf:

```
[simpletrans]
type=transport
protocol=udp
bind=0.0.0.0



[2000]
type = endpoint
direct_media = no
transport = simpletrans
context = test
disallow = all
allow = ulaw
aors = 2000
auth = auth2000

[2000]
type = aor
max_contacts = 2
```

```
[auth2000]
type=auth
auth_type=userpass
password=2000ab
username=2000

[4000]
type = endpoint
direct_media = no
transport = simpletrans
context = test
disallow = all
allow = ulaw
aors = 4000
auth = auth4000

[4000]
type = aor
max_contacts = 1

[auth4000]
type=auth
auth_type=userpass
password=4000ab
username=4000
```

### 5.3.3 IVR menu

Posledním krokem bylo vytvořit IVR menu, které uživatel uslyší, pokud zavolá na klapku podvodníka 2000. Pro vytvoření IVR a dialplanu bylo třeba změnit konfigurační soubor *extensions.conf*, který se nachází v systémové složce Asterisku */etc/asterisk*. Další důležitou částí v souboru je automatické vyzvednutí každého hovoru na linku 2000. Linka 4000 je testovací klapka pro přepojení na potenciální linku s prémiovou sazbou. Hovory mezi čísly v rozmezí 3000 - 3999 probíhají standardně.

IVR obsahuje dvě nahrané zprávy *vitejte.gsm* a *wanfiri.gsm*. Pokud uživatelské číslo zavolá na číslo podvodníka 2000, je přepojen na klapku 4000, spustí se IVR menu a zazní zpráva *vitejte*, ve které zní, že se volající dovolal na linku s přepojením a je nutné počkat na spojení. Pro přepojení má uživatel stisknout příslušné číslo. Po stisknutí čísla je toto číslo přečteno a uživatel musí počkat

5 sekund a je navrácen zpět do menu simulujícího frontu, ve které se opakuje opět celý proces se stisknutím příslušného čísla na přepojení.

Pokud uživatel stiskne tlačítko 0, přehraje se mu zpráva Wangiri, kde je mu oznámeno, že se stal obětí Wangiri podvodu a po 5 sekundách je opět přepojen zpět do IVR menu.

Nahrané audio soubory bylo třeba konvertovat na formátu GSM a přidat do umístění */usr/share/asterisk/sounds/en*, pokud se nacházel jinde, nepodařilo se hovory uskutečnit.

Problém pouze nastal při nahrávání vlastních zvukových nahrávek. Podle dokumentace Asterisku by se měly tyto soubory nacházet ve */var/lib/asterisk/sounds*. Po přidání souboru *wangiri.gsm* a *vitejte.gsm* však docházelo k chybám a hovory vůbec neproběhly. Po delším zkoumání byly přesunuty soubory do umístění */usr/share/asterisk/sounds/en* a test hovoru i IVR byl úspěšný. Část výpisu z CLI Asterisku při vstupu do fronty a testování její funkcionality:

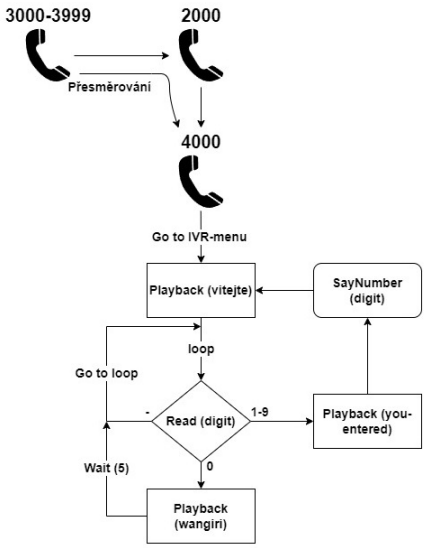
Konfigurační soubor *extensions.conf* s nastavením IVR:

```
[test]
exten => 4000,1,Goto(IVR-menu,s,1)
exten => 2000,1,Goto(4000,1)
exten => _3XXX,1,GotoIf($[${CALLERID(num)} != 2000]?not2000)
    same => n,Dial(PJSIP/${EXTEN},2)
    same => n,Goto(end)
exten => _3XXX,n(not2000),Dial(PJSIP/${EXTEN})
exten => _3XXX,n(end),Hangup()

[IVR-menu]
exten => s,1,Answer(500)
    same => n,Playback(vitejte)
    same => n(loop),Read(digit,,1,,,5)
    same => n,GotoIf($["${digit}" = ""]?loop)
    same => n,GotoIf($[${digit} = 0]?wangiri)
    same => n,Playback(you-entered)
    same => n,SayNumber(${digit})
    same => n,Playback(vitejte)
    same => n,Goto(loop)

    same => n(wangiri),Playback(wangiri)
    same => n,Wait(5)
    same => n,Goto(s,loop)

exten => t,1,Goto(loop)
```




Obrázek 5.3: Schéma IVR menu

## 5.4 Telefonní aplikace

Pro kontrolu hovoru byly využity VoIP aplikace Jitsi a PhonerLite. Jedná se o volně dostupné softwarové telefony. Obě aplikaci využívají pro signalizaci protokol SIP, takže byly vhodné pro účely této diplomové práce. Sloužily především jako simulace uživatelů telefonní sítě a k zahajování hovorů a testování IVR menu vytvořeném na ústředně Asterisk. Jitsi umožňuje registraci více klientů najednou, čehož bylo využito při testování generování podvodných volání přes SIPp a při volání zpět na číslo podvodníka.

## 5.5 Bezpečnostní analýza řešení

Implementované řešení na rozdíl od toho navrhovaného, je provozováno v uzavřeném prostředí. Je tedy pouze teoreticky možné odhadnout, jaká by byla úspěšnost podvodu, pokud by systém byl propojen do veřejné telefonní sítě nebo přes SIP trunk do internetu. Také je třeba zohlednit počet provedených hovorů z podvodného čísla. Čím více hovorů z podvodného čísla by bylo provedeno, lze předpokládat, že tím více volaných by zavolalo zpět a stalo se tak obětí Wangiri podvodu.

Dle informací norského poskytovatele telekomunikačních služeb Telenor z roku 2021 zhruba 7 až 8 % zákazníků zavolá zpět po prozvonění z podvodného čísla. Během jednoho cíleného útoku na služby firmy Telenor 19. ledna 2021, kdy podvodníci využívali čísla ze Severní Koreje, byla ovšem četnost zavolání zpět na číslo útočníka pouze 4 %. Výkyvy celkové úspěšnosti Wangiri podvodů jsou

pravděpodobně zapříčiněny faktem, že se jedná z části o druh sociálního inženýrství, a tudíž každý člověk na něj reaguje jinak. [27]

Úspěšnost reálného řešení bych zhodnotil po analýze výsledků poměru počtu provedených prozvonění k počtu zavolání zpět za určitou dobu. Jeden z dalších aspektů úspěšnosti řešení by byla jeho výdělečnost, tedy poměr zisku z čísla s prémiovou sazbou k nákladům na provoz systému.

Terčem útoku se však může stát i samotný navržený systém. Například typem útoku PBX hacking (viz 2.3.9) nebo Flooding DoS (viz 2.3.5). Proti nim se lze bránit např. použitím firewallu, omezením provozu atd. Lze také změnit nastavení pobočkové ústředny použitím protokolu TCP místo UDP pro přenos dat a využitím protokolů TLS nebo SSL pro bezpečnější komunikaci mezi prvky systému.

# Kapitola 6

# Testování implementovaného řešení

Tato část je věnovaná testování implementovaného řešení z kapitoly 5. Cílem bylo otestovat všechny části sytému, jejich funkčnost a případně popsat problémy vzniklé při tvorbě implementace. Na závěr návrh testovaní systému v reálném prostředí.

## 6.1 Testování web scraperu

Je nutné poznamenat, že web scraper vybírá celý HTML kód stránky a porovnává jeho části s regulárním výrazem, je možné, že se v kódu takové řetězce mohou vyskytovat i skrytě nebo se nebude jednat o telefonní čísla. V takovém případě by web scraper uložil i tato čísla do výsledného CSV souboru. SIPp by však jen vyzkoušel dané číslo prozvonit, nepodařilo by se, ale na výslednou funkcionalitu systému by to nemělo vliv.

### 6.1.1 Testování vyhledávání čísel z testovací stránky

Pro testování vyhledávání čísel pro laboratorní prostředí byla vytvořena testovací webová stránka přidaná na domovskou stránku autora na serveru *HomeL.vsb.cz - https://homel.vsb.cz/ jak0087/*. Kód stránky je přidán do příloh diplomové práce. Nachází se na ní tři čísla klapek - 3000, 3001 a 3789 různě rozmístěna v textu stránky. Po spuštění skriptu ze serveru, na němž je zprovozněn SIPp *webscraper.py https://homel.vsb.cz/ jak0087/* vznikl soubor *tel_cisla.csv*. Hláška z konzole:

```
python3 webscraper.py https://homel.vsb.cz/~jak0087/
Nalezeno 3 čísel na https://homel.vsb.cz/~jak0087/ a uloženo do souboru
tel_cisla.csv
```

Výsledný obsah souboru:

```
cat tel_cisla.csv
SEQUENTIAL
3789
3001
3000
```

### 6.1.2 Testování regulárního výrazu pro česká telefonní čísla

Dalším testem prošel regulární výraz pro česká telefonní čísla. Jako testovací stránky byly použity stránky s kontakty vedení katedry Telekomunikační techniky na *https://www.fei.vsb.cz/440/cs/o-katedre/vedeni-katedry/* a stránka Kontakty, mapy, podání VŠB-TUO (*https://www.vsb.cz/cs/o-univerzite/kontakty-mapy-parkovani/*. Na první zmíněné stránce je celkem pět telefonních čísel, na druhé jsou telefonní čísla dvě. Pro účel testu byl skript zkopírován a odkomentován regulární výraz pro vyhledávání čísel v českém formátu.

```
python3 webscrapercz.py https://www.fei.vsb.cz/440/cs/o-katedre/vedeni-katedry
Nalezeno 5 čísel na https://www.fei.vsb.cz/440/cs/o-katedre/vedeni-katedry
a uloženo do souboru tel_cisla.csv
python3 webscrapercz.py https://www.vsb.cz/cs/o-univerzite/kontakty-mapy-
parkovani
Nalezeno 2 čísel na https://www.vsb.cz/cs/o-univerzite/kontakty-mapy-parkovani
a uloženo do souboru tel_cisla.csv

cat tel_cisla.csv
SEQUENTIAL
+420596995940
+420596995942
+420596991290
+420596995944
+420596996070
+420596991111
+420596995500
```

## 6.2 Generování hovorů přes SIPp

Při testování generování hovorů nejdříve docházelo k problémům se čtením .csv souboru obsahující telefonní čísla, a to kvůli EOL (end of line) znakům na konci řádku každého čísla. Původně se

jednalo o Windows CR+LF znaky, ty však SIPp nerozpoznal a generování relací nefungovalo. Pro toto zjištění byl použit program PSPad a zkratka *:set list* v Linuxovém editoru *vi*.

Tento problém byl vyřešen změněním tohoto znaku ve skriptu web scraperu pomocí vlastnosti lineterminator na typ znaku Unix systémů - LF.

```
writer = csv.writer(csvfile, lineterminator='\n')
```

Hovory byly generovány přes tento příkaz:

```
./sipp -sf proxy.xml -aa -r 3 192.168.69.22 -s 2000 -i 192.168.69.11 -p 5060
-m 3 -au 2000 -ap 2000 -inf tel_cisla.csv
```

Vysvětlení parametrů příkazu:

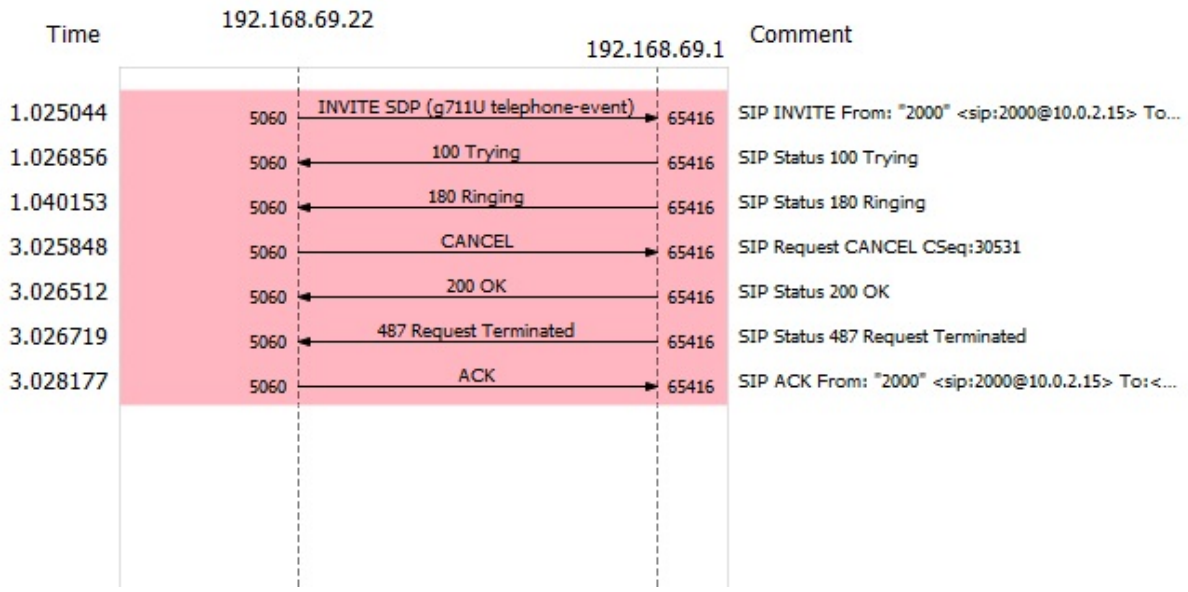
- **-sf** - cesta k XML souboru proxy.xml,
- **-aa** - autoanswer na žádosti Asterisku
- **-r** - počet hovorů za daný interval
- **192.168.69.22** - IP adresa cílového serveru s Asteriskem
- **-s** - klapka volajícího
- **-i** - lokální adresa (SIPp server) 192.168.69.11
- **-p** - port
- **-m** - počet hovorů
- **-au** - autentizační jméno volajícího
- **-ap** - heslo volajícího
- **-inf**- cesta k .csv souboru

Po spuštění generování hovorů, se provedly tři krátké prozvonění na čísla 3000, 3001 a 3789. Z diagramu dialogu na obrázku 6.1 je zřejmý průběh prozvonění, kdy po dvou sekundách je vyžádáno zrušení žádosti a dojde k ukončení hovoru.

Výpis z Asterisk CLI:

```
Executing [3789@test:1] GotoIf("PJSIP/2000-00000018", "0?not2000") in new
stack
-- Executing [3789@test:2] Dial("PJSIP/2000-00000018", "PJSIP/3789,2")
in new stack
-- Called PJSIP/3789
```

```
-- PJSIP/3789-00000019 is ringing
-- Executing [3001@test:1] GotoIf("PJSIP/2000-0000001a", "0?not2000")
in new stack
-- Executing [3001@test:2] Dial("PJSIP/2000-0000001a", "PJSIP/3001,2") in
new stack
-- Called PJSIP/3001
-- PJSIP/3001-0000001b is ringing
-- PJSIP/3789-00000019 is ringing
-- Executing [3000@test:1] GotoIf("PJSIP/2000-0000001c", "0?not2000") in new
stack
-- Executing [3000@test:2] Dial("PJSIP/2000-0000001c", "PJSIP/3000,2") in new
stack
-- Called PJSIP/3000
-- PJSIP/3000-0000001d is ringing
-- PJSIP/3001-0000001b is ringing
-- PJSIP/3789-00000019 is ringing
-- PJSIP/3001-0000001b is ringing
-- Nobody picked up in 2000 ms
-- Executing [3789@test:3] Goto("PJSIP/2000-00000018", "end") in new stack
-- Goto (test,3789,5)
-- Executing [3789@test:5] Hangup("PJSIP/2000-00000018", "") in new stack
== Spawn extension (test, 3789, 5) exited non-zero on 'PJSIP/2000-00000018'
-- Nobody picked up in 2000 ms
-- Executing [3001@test:3] Goto("PJSIP/2000-0000001a", "end") in new stack
-- Goto (test,3001,5)
-- Executing [3001@test:5] Hangup("PJSIP/2000-0000001a", "") in new stack
== Spawn extension (test, 3001, 5) exited non-zero on 'PJSIP/2000-0000001a'
-- Nobody picked up in 2000 ms
-- Executing [3000@test:3] Goto("PJSIP/2000-0000001c", "end") in new stack
-- Goto (test,3000,5)
-- Executing [3000@test:5] Hangup("PJSIP/2000-0000001c", "") in new stack
== Spawn extension (test, 3000, 5) exited non-zero on 'PJSIP/2000-0000001c'
```




Obrázek 6.1: Diagram dialogu

## 6.3 Asterisk ústředna a hovory

Posledním krokem bylo samotné testování hovorů na klapky uživatelů a klapku podvodníka. Pro registraci čísla 3000 jsem využil softwarový telefon PhonerLite, pro registraci čísel 3001, 3789 softwarový telefon Jitsi. Dle testu v podkapitole 6.2 lze konstatovat, že podvodníkovo číslo 2000 prozvonilo všechna tři čísla na 2 sekundy, a poté bylo vyzvánění automaticky ukončeno.

Pokud však došlo k zavolání na číslo útočníka, byl hovor automaticky vyzvednut a přesměrován na číslo 4000. Hovor pokračuje v IVR menu. Postup konfigurace IVR menu je popsán v kapitole 5.3.3 společně se schématem, jak probíhá.

Z výpisu CLI Asterisku lze vyčíst postup testování. V první části je po vytočení čísla 2000 hovor přesměrován na číslo 4000. Následuje vyzvednutí hovoru a spuštění IVR menu, které začne přehráním hlasového záznamu *vitejte.gsm*. V něm hlas vyzve volajícího, ať stiskne číslo pro přepojení.

```
-- Executing [2000@test:1] Goto("PJSIP/3000-00000020", "4000,1") in new stack
-- Goto (test,4000,1)
-- Executing [4000@test:1] Goto("PJSIP/3000-00000020", "IVR-menu,s,1")
in new stack
-- Goto (IVR-menu,s,1)
-- Executing [s@IVR-menu:1] Answer("PJSIP/3000-00000020", "500") in new stack
> 0x55b55a3b41a0 -- Strict RTP learning after remote address
set to: 192.168.69.1:65432
> 0x55b55a3b41a0 -- Strict RTP switching to RTP target address
```

```
192.168.69.1:65432 as
source -- Executing [s@IVR-menu:2] Playback("PJSIP/3000-00000020", "vitejte")
in new stack
-- <PJSIP/3000-00000020> Playing 'vitejte.gsm' (language 'en')
> 0x55b55a3b41a0 -- Strict RTP learning complete - Locking on source address
192.168.69.1:65432
```

Hovor pokračuje v IVR menu. Volající stiskl tlačítko 3. Jelikož se nejedná o číslo 0, jsou přes aplikace *Playback()* a *SayNumber()* přehrány výchozí zprávy Asterisku *you-entered.gsm* a *digits/3.gsm*, které uživateli oznámí, že zmáčkl číslo 3. Poté je uživatel přesunut zpět do fronty. Po přehrání hlasového souboru *vitejte.gsm* je opět vyzván ke stisku tlačítka.

```
-- Executing [s@IVR-menu:3] Read("PJSIP/3000-00000020", "digit,,1,,,5")
in new stack
-- Accepting a maximum of 1 digits.
-- User entered '3'
-- Executing [s@IVR-menu:4] GotoIf("PJSIP/3000-00000020", "0?loop") in new
stack
-- Executing [s@IVR-menu:5] GotoIf("PJSIP/3000-00000020", "0?wangiri")
in new stack
-- Executing [s@IVR-menu:6] Playback("PJSIP/3000-00000020", "you-entered")
in new stack
-- <PJSIP/3000-00000020> Playing 'you-entered.gsm' (language 'en')
-- Executing [s@IVR-menu:7] SayNumber("PJSIP/3000-00000020", "3") in new
stack
-- <PJSIP/3000-00000020> Playing 'digits/3.gsm' (language 'en')
-- Executing [s@IVR-menu:8] Playback("PJSIP/3000-00000020", "vitejte")
in new stack
-- <PJSIP/3000-00000020> Playing 'vitejte.gsm' (language 'en')
-- Executing [s@IVR-menu:9] Goto("PJSIP/3000-00000020", "loop") in new stack
-- Goto (IVR-menu,s,3)
-- Executing [s@IVR-menu:3] Read("PJSIP/3000-00000020", "digit,,1,,,5")
in new stack
-- Accepting a maximum of 1 digits.
```

Tentokrát volající zmáčkl číslo 0. Je splněna podmínka pro přehrání hlasové zprávy *wangiri.gsm*, v níž se dozví, že se stal obětí podvodu, ale pro přepojení může po 5 sekundách stisknout tlačítko s číslem. Po jejím přehrání je pauza 5 sekund a volající je opět přesunut zpátky do fronty.

```
-- User entered '0'
-- Executing [s@IVR-menu:4] GotoIf("PJSIP/3000-00000020", "0?loop") in new
stack
-- Executing [s@IVR-menu:5] GotoIf("PJSIP/3000-00000020", "1?wangiri") in
new stack
-- Goto (IVR-menu,s,10)
-- Executing [s@IVR-menu:10] Playback("PJSIP/3000-00000020", "wangiri")
in new stack
-- <PJSIP/3000-00000020> Playing 'wangiri.gsm' (language 'en')
-- Executing [s@IVR-menu:11] Wait("PJSIP/3000-00000020", "5") in new stack
-- Executing [s@IVR-menu:12] Goto("PJSIP/3000-00000020", "s,loop")  in
new stack
-- Goto (IVR-menu,s,3)
-- Executing [s@IVR-menu:3] Read("PJSIP/3000-00000020", "digit,,1,,,5") in new
stack
-- Accepting a maximum of 1 digits.
```

Volající stiskl 9 a opakuje se proces popsaný výše, když volající stiskl klávesu 3. Po opětovném návratu do fronty volající ukončuje hovor.

```
-- User entered '9'
-- Executing [s@IVR-menu:4] GotoIf("PJSIP/3000-00000020", "0?loop") in new
stack
-- Executing [s@IVR-menu:5] GotoIf("PJSIP/3000-00000020", "0?wangiri") in
new stack
-- Executing [s@IVR-menu:6] Playback("PJSIP/3000-00000020", "you-entered") in
new stack
-- <PJSIP/3000-00000020> Playing 'you-entered.gsm' (language 'en')
-- Executing [s@IVR-menu:7] SayNumber("PJSIP/3000-00000020", "9") in new stack
-- <PJSIP/3000-00000020> Playing 'digits/9.gsm' (language 'en')
-- Executing [s@IVR-menu:8] Playback("PJSIP/3000-00000020", "vitejte") in new
stack
-- <PJSIP/3000-00000020> Playing 'vitejte.gsm' (language 'en')
-- Executing [s@IVR-menu:9] Goto("PJSIP/3000-00000020", "loop") in new stack
-- Goto (IVR-menu,s,3)
-- Executing [s@IVR-menu:3] Read("PJSIP/3000-00000020", "digit,,1,,,5") in new
stack
-- Accepting a maximum of 1 digits.
-- User entered nothing.
```

```
-- Executing [s@IVR-menu:4] GotoIf("PJSIP/3000-00000020", "1?loop") in new
stack
-- Goto (IVR-menu,s,3)
-- Executing [s@IVR-menu:3] Read("PJSIP/3000-00000020", "digit,,1,,,5") in
new stack
-- Accepting a maximum of 1 digits.
-- User disconnected
```


Obrázek 6.2: Diagram dialogu hovoru na číslo podvodníka

### 6.3.1 Shrnutí testování

Testování autonomního vyhledávače telefonních čísel bylo úspěšné. Vyhledal správně jak čísla klapek pro laboratorní prostředí z testovací stránky, tak i česká telefonní čísla ze stránek na internetu. Jedinou nevýhodou je scraping pouze jedné stránky najednou při každém spuštění skriptu.

Testování generování podvodných hovorů přes SIPp bylo také úspěšné. Implementovaný systém fungoval podle návrhu. Došlo ke krátkému prozvonění čísel získaných z webových stránek a uložených do souboru přes autonomní vyhledávač. Zároveň tak byla otestována i část konfigurace pobočkové ústředny Asterisk.

Testování pobočkové ústředny Asterisk bylo také úspěšné. Hovory příchozí i odchozí byly prováděny podle návrhu. Přesměrování z čísla podvodníka 2000 na číslo 4000 simulující linku s prémiovou sazbou bylo také úspěšné. IVR menu fungovalo podle návrhu. Jednalo se o testovací prostředí interaktivního menu.

Při implementaci systému pro generování podvodných hovorů v reálném prostředí bych v testování postupoval podobně. Nejdřív bych otestoval správné získávání telefonních čísel z webových stránek. Pak bych otestoval systém pro generování hovorů. Po provedení několika testovacích odchozích hovorů a následné analýze poměru odchozích a příchozích hovorů, tedy úspěšných podvodů, by došlo k rozhodnutí, zda by dané řešení bylo úspěšné či nikoli. A to na rozdíl od testovacího laboratorního prostředí, kde bylo hlavním cílem dokázat, zda je takové řešení proveditelné.

# Kapitola 7

# Bezpečnostní zásady a vhodná protiopatření vůči podvodu Wangiri

Kapitola se věnuje návrhu vhodných protiopatření a bezpečnostním zásadám proti podvodu Wangiri. Je přihlédnuto k informacím získaných během vypracované rešerše v kapitole 2, a také při vypracování praktické části této diplomové práce. Doporučení jsou oddělena do dvou kategorií. První kategorií jsou personální zásady, druhou kategorií jsou technická protiopatření.

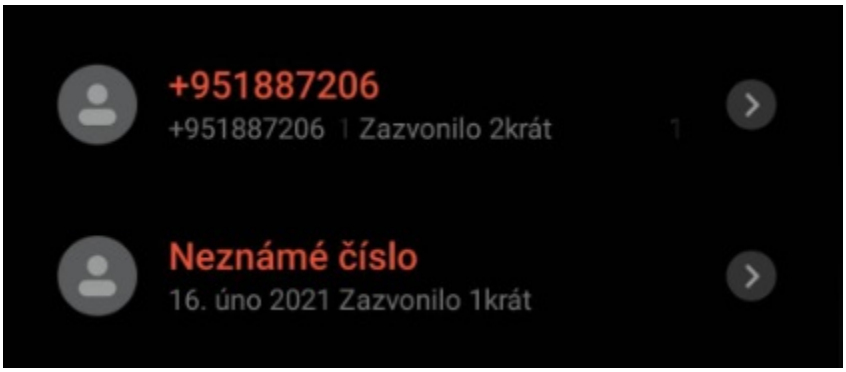
## 7.1 Personální bezpečnostní zásady

Personální zásady jsou opatření pro koncové uživatele, jak se nestát obětí podvodného volání.

### 7.1.1 Volání na neznámá čísla

Asi nejdůležitější prevencí je nevolat na neznámá, hlavně mezinárodní čísla. Během vypracování této práce jsem se pravděpodobné sám stal potenciálním cílem Wangiri podvodu, kdy mi nejdříve bylo voláno z neznámého čísla a následně jsem byl prozvoněn z čísla s předvolbou +95, což je mezinárodní předvolba státu Myanmar.

Pokud je uživatel cílem podvodných volání častěji, je vhodné upozornit svého mobilního operátora nebo poskytovatele telekomunikačních služeb. Zajímavým faktem je, že oba hovory byly již označeny jako podvodné. Proto jsou i na obrázku 7.1 a v historii hovorů napsány červeným písmem. Je to pravděpodobně právě díky mobilnímu operátorovi, který je takto označil.




Obrázek 7.1: Potenciální podvodný hovor

### 7.1.2 Vzdělávání uživatelů

Jelikož Wangiri podvod pro svůj úspěch využívá z části sociální inženýrství, nejlepším protiopatřením, jak se nestát jeho obětí, je vědět, jak probíhá, funguje a na co cílí. Důležitým aspektem je vzdělávání svých zákazníků, zaměstnanců a uživatelů před dopady podvodu. V dnešní době už o Wangiri útoku existuje relativně dost informací. Hlavními propagátory způsobu jak nebýt podveden, jsou mobilní operátoři, kteří jsou poznamenáni Wangiri podvody nejvíce. Hlavně díky ztrátovosti, která, jak bylo v práci uvedeno, se ročně vyšplhá až k několika miliardám dolarů.

Existují už i kampaně pro šíření osvěty proti Wangiri podvodu. Na obrázku 7.2 je informativní leták z kampaně proti podvodným voláním Agentury Evropské unie pro spolupráci v oblasti prosazování práva Europol.[28]


Obrázek 7.2: Leták organizace Europol

### 7.1.3 Aplikace pro rozpoznání nevyžádaných hovorů

Další bezpečnostní zásadou je používání aplikací pro rozpoznávání nevyžádaných hovorů. Jedná se o různé antiviry, blokátory příchozích hovorů z vybraných čísel, ochrana proti spam hovorům. Jedná se o aplikace jako Truecaller, ESET Security, Hiya atd. Chrání uživatele před nevyžádanými hovory prostřednictvím rozpoznání podezřelých čísel, blokováním mezinárodních čísel nebo filtry SMS zpráv.

Existuje také systém nahlašování čísel, kdy uživatel, který pozná potenciálně škodlivé číslo, ho nahlásí vývojáři aplikace nebo mobilnímu operátorovi. Číslo je pak zavedeno v databázi škodlivých čísel a zablokováno pro ostatní uživatele.

## 7.2 Technická protiopatření

Technická protiopatření mohou být implementovány na straně infrastruktury, tedy u poskytovatelů telekomunikačních služeb.

### 7.2.1 Sledování příchozích hovorů

Jedním z aspektů, jak mohou poskytovatelé služeb zabránit Wangiri útokům, je monitorování provozu a prostředků na síti. Existuji řada monitorovacích systémů. Serverové, aplikační, síťové apod. Jedná se dnes o standardní součást každé infrastruktury.

Pokud zpozoruje poskytovatel služeb podezřelou aktivitu, může ji včas zastavit například blokací příchozích a odchozích hovorů. Jedná se o sledování příchozích hovorů ze zemí s velkým rizikem podvodných volání. Sledování velkého množství krátkých hovorů, které nebyly vyzvednuty. Nebo naopak zvýšený počet odchozích hovorů na mezinárodní čísla, kdy zvýšená aktivita může znamenat, že už došlo k provedení Wangiri útoku a podvedení začali volat zpět na čísla s prémiovou sazbou.

### 7.2.2 Blokace příchozích a odchozích hovorů

Tato část úzce souvisí s některými ostatními protiopatřeními. Pokud poskytovatel služeb rozpozná škodlivá čísla, může je ze své sítě blokovat. Díky včasnému zásahu je možné předejít velkému útoku na infrastrukturu, kdy může během krátké doby dojít až ke stovkám tisícům podvodných volání.

### 7.2.3 Systémy detekce podvodů

Jedná se o systémy, které slouží k detekci již proběhlých nebo právě probíhajících podvodů. Mohou napomáhat ke zjištění slabých míst infrastruktury. Provozovatel pak může problémy zaznamenat, vyřešit a předejít tak dalším ztrátám v budoucnu.

Každý systém detekce podvodů by měl být schopen detekovat ještě probíhající podvod. Detekce podvodů je spojená s monitoringem, kdy je detekce vyhodnocována podle založených pravidel, profilování uživatelů nebo vizualizaci dat.[5]

U systému detekce podle definovaných pravidel dochází k detekci podvodu, pokud jsou kritéria pro vyvolání incidentu naplněna. Jeho nevýhodou je však zranitelnost proti neznámým druhům podvodu nebo proti podvodům, na které nejsou definovány pravidla. Také náročnost při implementaci nových pravidel pro detekci.[5]

Detekce na principu vizualizace je prakticky popsána v 7.2.1. Jeho nevýhodou je samozřejmě výskyt lidských chyb.

Profilování uživatelů je dnes stále populárnější díky vzestupu umělé inteligence a neuronových sítí. Profil uživatele je tvořen jeho chováním. Jeho profil chování v minulosti je porovnáván s momentálním profilem chování. Pokud je rozdíl velký, je možné, že se jedná o podvod.

# Kapitola 8

# Závěr

Práce byla zaměřena na analýzu podvodných volání v současné době a především na návrh a implementaci systému pro realizaci Wangiri podvodu.

První část práce popisuje prostředí IP telefonie a některé využívané protokoly. Zároveň je v této části popsáno několik nejrozšířenějších druhů útoků a podvodů. Hlavním tématem jsou podvodná volaní a útoky v prostředí IP telefonie. Především je kladen důraz na analýzu způsobu, jak podvody probíhají a analýzu podvodných volání Wangiri, která jsou v poslední době prováděna stále častěji.

Další kapitola obsahuje návrh řešení systému pro realizaci Wangiri podvodu v reálném prostředí. Po analýze a porovnání existujících řešení byla vytvořena vlastní implementace autonomního vyhledávače telefonních čísel.

Kapitola pět se zabývá implementací navrženého systému v testovacím prostředí. Jelikož se jedná o podvod spojený se sociálním inženýrstvím, tak především z etických důvodu byla implementace systému provedena pouze v laboratorním prostředí. To znamená, že telefonní čísla byla nahrazena čtyřcifernými čísly a celková topologie byla upravena. Byl kladen důraz na co nejnižší cenu takového řešení. Návrh obsahuje vlastní řešení vyhledávání telefonních čísel z webových stránek, konfiguraci generátoru hovoru a instalaci a konfiguraci pobočkové ústředny.

Předposlední část práce shrnula testování implementovaného řešení. Autonomní vyhledávač telefonních čísel úspěšně vyhledal všechna čísla z webového zdroje. Nicméně je možné jej zdokonalit, rozšířit a celou implementaci více zautomatizovat. Generování více hovorů najednou pomocí SIPp bylo také úspěšné. IVR menu a konfigurace Asterisk ústředny byly také otestovány. Všechny části řešení Asterisk ústředny fungovaly dle očekávání.

Poslední část práce se věnuje návrhu vhodných protiopatření, jak předcházet a nestát se obětí podvodu Wangiri. Zásady vychází z informací získaných během rešerše a samotného vypracování diplomové práce.

Závěr práce je takový, že teoreticky lze bez velkých nákladů vytvořit prostředí k provádění Wangiri podvodu. A to i v reálném prostředí, ovšem za předpokladu, že by implementované řešení bylo pozměněno podle návrhu a napojeno na veřejnou telefonní síť či SIP trunk.

Z provedené analýzy podvodných volání a z ní vyvozených protiopatřeních je zřejmé, že podvodům jako je Wangiri a jemu podobným se nejlépe předchází především tím, když potencionální oběť ví, jak je takový podvod nebo útok prováděn a na jaký typ sociálního chování cílí.

# Literatura


1. VOZŇÁK, Miroslav. *Technologie a protokoly multimediálních komunikací pro integrovanou výuku VUT a VŠB-TUO*. Ostrava: Vysoká škola báňská - Technická univerzita Ostrava, 2014. ISBN 978-80-248-3326-2.
2. 3CX. *3CX communications system* [online] [cit. 2022-01-07]. Dostupné z: `https://www.3cx.com/pbx/`.
3. ROSENBERG, J.; SCHULZRINNE, H.; CAMARILLO, G.; JOHNSTON, A.; PETERSON, J.; SPARKS, R.; HANDLEY, M.; SCHOOLER, E. *SIP: Session Initiation Protocol* [online]. 2002 [cit. 2023-04-22]. Dostupné z: `https://datatracker.ietf.org/doc/html/rfc3261`.
4. COLLIER, Mark; ENDLER, David. *Hacking Exposed Unified Communications & VoIP Security Secrets & Solutions, Second edition*. 2. vyd. New York: McGraw-Hill Education, 2014.
5. ALI, Mohammed Aamir; AZAD, Muhammad Ajmal; CENTENO, Mario Parreno; HAO, Feng; MOORSEL, Aad van. Consumer-facing technology fraud: Economics, attack methods and potential solutions. *Future Generation Computer Systems*. 2019, roč. 100, s. 408–427.
6. NETWORKS, Enghouse. *Detect & Protect Against Wangiri Callback Fraud* [online] [cit. 2022-01-07]. Dostupné z: `https://www.enghousenetworks.com/enghouse-resources/blog/detect-protect-against-wangiri-callback-fraud/`.
7. ARAFAT, Mais; QUSEF, Abdallah; SAMMOUR, George. Detection of wangiri telecommunication fraud using ensemble learning. In: *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*. 2019, s. 330–335.
8. ČESKÁ SPOŘITELNA, a.s. *Vishing* [online] [cit. 2022-01-07]. Dostupné z: `https://www.csas.cz/cs/o-nas/bezpecnost-ochrana-dat/vishing`.
9. TANG, Jin; CHENG, Yu; HAO, Yong; SONG, Wei. SIP flooding attack detection with a multi-dimensional sketch design. *IEEE Transactions on Dependable and Secure Computing*. 2014, roč. 11, č. 6, s. 582–595.
10. VOZŇÁK, Miroslav; ŘEZÁČ, Filip; ROZHON, Jan. *Bezpečnost v komunikacích*. Ostrava: Vysoká škola báňská - Technická univerzita Ostrava, 2013.

11. SAHIN, Merve; FRANCILLON, Aurélien. Over-the-top bypass: Study of a recent telephony fraud. In: *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. 2016, s. 1106–1117.

12. SUBEX. *PBX Hacking* [online] [cit. 2022-01-07]. Dostupné z: `https://www.subex.com/pdf/Point_of_View/PBX-Hacking-White-Papers.pdf`.

13. EL-MOUSSA, Fadi; MUDHAR, Parmindher; JONES, Andy. Overview of SIP attacks and countermeasures. In: *International Conference on Information Security and Digital Forensics*. 2009, s. 82–91.

14. ASSOCIATION, Communications Fraud Control. *Fraud Loss Survey Report 2021* [online] [cit. 2023-04-21]. Dostupné z: `https://cfca.org/wp-content/uploads/2021/12/CFCA-Fraud-Loss-Survey-2021-2.pdf`.

15. APIFY. *Contact Details Scraper* [online] [cit. 2022-01-07]. Dostupné z: `https://apify.com/vdrmota/contact-info-scraper#results`.

16. CONVERTCSV.COM. *Phone Extractor For Web Pages and Text* [online] [cit. 2022-01-07]. Dostupné z: `https://www.convertcsv.com/phone-extractor.htm`.

17. SCRAPEBOX. *Scrape Box - Phone Number Scraper* [online] [cit. 2022-01-07]. Dostupné z: `http://www.scrapebox.com/phone-number-scraper`.

18. WINTECHSOFT. *Phone Number Extractor Files* [online] [cit. 2022-01-07]. Dostupné z: `https://www.wintechsoft.com/file-number-grabber.html`.

19. SOLUTION, Technocom. *Phone Number Web Extractor* [online] [cit. 2022-01-07]. Dostupné z: `https://www.technocomsolutions.com/phone-web-extractor.html`.

20. SOFTWARE, Ahmad. *Phone Number Grabber* [online] [cit. 2022-01-07]. Dostupné z: `https://www.ahmadsoftware.com/72/9/phone-number-grabber.html`.

21. SIPP. *SIPp* [online] [cit. 2022-01-07]. Dostupné z: `http://sipp.sourceforge.net/`.

22. VALID8. *SIP Loader* [online] [cit. 2023-04-20]. Dostupné z: `https://www.valid8.com/datasheets/sip-loader`.

23. STARTRINITY.COM. *StarTrinity SIP Tester (call generator, simulator) - VoIP monitoring and testing tool* [online] [cit. 2022-01-07]. Dostupné z: `http://startrinity.com/VoIP/SipTester/SipTester.aspx`.

24. VOZŇÁK, Miroslav; ŘEZÁČ, Filip. *Asterisk - teorie a praxe*. Ostrava: Vysoká škola báňská - Technická univerzita Ostrava, 2011.

25. ASTERISK. *Asterisk* [online] [cit. 2022-01-07]. Dostupné z: `https://www.asterisk.org/`.

26. PROJECT, The Kamailio SIP Server. *Kamailio* [online]. 2023 [cit. 2023-04-23]. Dostupné z: `https://www.kamailio.org/w/`.

27. TELENOR. *Norwegians Hit by a Wave of Fraud: 200,000 Calls from North Korea* [online]. 2021 [cit. 2023-04-22]. Dostupné z: https://www.mynewsdesk.com/uk/telenor/pressreleases/norwegians-hit-by-a-wave-of-fraud-200-dot-000-calls-from-north-korea-3065401.

28. EUROPOL, European Cybercrime Center. *WANGIRI – A TELEPHONE SCAM* [online]. 2021 [cit. 2023-04-25]. Dostupné z: https://www.europol.europa.eu/cms/sites/default/files/documents/wangiri_final_2.pdf.

# Příloha A

# Příloha v IS Edison

Práce rovněž obsahuje elektronickou přílohu s konfiguračními soubory a skripty.

- **extension.conf** - konfigurační soubor Asterisku
- **generator_pjsip_uctu.py** - skript pro generování uživatelských účtů
- **pjsip.conf** - konfigurační soubor Asterisku
- **proxy.xml** - soubor pro generování SIP žádostí
- **webscraper.py** - skript autonomního vyhledávače telefonních čísel