

Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií

Zadanie 1 – SIP Proxy (telefónna ústredňa)

Mobilné technológie a aplikácie

Matej Skyčák ID: 111652

Akademický rok: 2021/2022

I. Zadanie

Na vašom počítači (alebo virtuálnom počítači) sprevádzkujte SIP Proxy, ktorá umožní prepájanie a realizáciu hovorov medzi štandardnými SIP klientami.

Na implementáciu vašej SIP Proxy si môžete zvoliť akýkoľvek programovací jazyk a použiť akúkoľvek SIP knižnicu, ktorá pre daný programovací jazyk existuje. Vo výsledku však musíte spúšťať “váš kód”, v ktorom sú zakomponované knižnice, ktoré poskytujú funkcionality SIP Proxy.

Rozsah povinných funkcionalít:

- Registrácia účastníka (bez nutnosti autentifikácie)
- Vytočenie hovoru a zvonenie na druhej strane
- Prijatie hovoru druhou stranou, fungujúci hlasový hovor
- Ukončenie hlasového hovoru (prijatého aj neprijatého)

Doplňkové funkcionality:

- Možnosť zrealizovať konferenčný hovor (aspoň 3 účastníci)
- Možnosť presmerovať hovor
- Možnosť realizovať videohovor
- Logovanie “denníka hovorov” – kto kedy komu volal, kedy bol ktorý hovor prijatý, kedy bol ktorý hovor ukončený, do ľubovoľného textového súboru v ľubovoľnom formáte
- Úprava SIP stavových kódov z zdrojovom kóde proxy, napr. “486 Busy Here” zmeníte na “486 Obsadené”

II. Opis riešenia

Pre riešenie tohto zadania som použil už existujúci kód link: <https://github.com/tirfil/PySipFullProxy>, ktorý som použil ako knižnicu. Z nej som si následne vo svojom kóde, kde som ju použil, volal potrebné funkcie. Pôvodný kód bol písaný na inú verziu, ako som ja používal (Python 3), preto bolo potrebné niektoré riadky opraviť. Rovnako tak som si vytvoril druhý main, odkiaľ pôvodný kód používam ako importovanú knižnicu. Do pôvodného kódu som zasahoval, len pri aktualizácii syntaxe a pri zmene stavových kódov.

Odkaz na môj GitHub repozitár: https://github.com/Matej96/zadanie_1

Z doplnkových funkcionalít som sa snažil splniť všetky. Prvé 3 funkcionality poskytoval klient samotný, len bolo potrebné mierne zmeniť nastavenia napríklad, aby podporoval video. Zadanie som testoval pomocou aplikácie Linphone. Na všetky z povinných, ale

aj doplnkových funkcionalít som urobil pcapy vo Wiresharku, kde je možné vidieť aj zmeny stavových kódov, alebo aj samotnú komunikáciu.

Proxy som sprevádzkoval na svojom notebooku a potom som sa na neho pripájal z mobilu, prípadne dvoch. Tak som otestoval napríklad presmerovanie hovoru alebo aj konferenčný hovor.

```
> 01.03. 19:24:53 :: Call-ID: 9viJW53fVt :: ntb vola mobil
> 01.03. 19:24:59 :: Call-ID: i~Jp7xNZsn :: mobil vola ntb
> 01.03. 19:25:12 :: Call-ID: i~Jp7xNZsn :: ntb zrusil
> 01.03. 19:25:26 :: Call-ID: m3NcEe-BP8 :: mobil vola ntb
> 01.03. 19:25:36 :: Call-ID: m3NcEe-BP8 :: ntb zrusil
```

Hore na obrázku je ukážka toho ako som implementoval denník hovorov, je tam možné vidieť všetko, čo zadanie požadovalo. Ako bonus som tam doplnil, že je vidno aj kto hovor zrušil.

III. Použité knižnice

Pre korektné spustenie môjho riešenia sú potrebné tieto knižnice. Socket server je knižnica na spustenie servera, knižnica re poskytuje prácu s regulárnymi výrazmi, ktoré prevzatá knižnica poskytuje. Sys slúži iba na prácu s argumentami z príkazového riadku, nie je v podstate ani potrebná. Používa sa iba, ak by sme chceli zadať IP, na ktorej má proxy server bežať. Knižnicu logging je potrebné importovať iba pre prehľad vo výpisoch, ale na to môže slúžiť v podstate aj Wireshark. A nakoniec knižnica socket pre klasickú prácu so sieťovými vecami napríklad na získavanie IP adries a podobne.

```
import socketserver
import re
import sys
import time
import logging
import socket
```

IV. Záver

Zadanie mi rozhodne pomohlo pochopiť ako funguje SIP protokol. Ako sa registruje používateľ a ako si navzájom môžu volať. Hlavne pri menení stavových kódov na vlastné som pochopil, aké správy sa medzi klientom a proxy vymieňajú a čo znamenajú.