Slovenská technická univerzita v Bratislave

Fakulta informatiky a informačných technológií

Ilkovičova 2, 842 16 Bratislava 4

**Zadanie 1 – SIP Proxy (telefónna ústredňa)**

Mobilné technológie a aplikácie

Patrik Kecera

ID: 110815

ak. rok: 2021/22

Cvičiaci: Ing. Adam Ševčík

**OBSAH**

[1. ZADANIE ÚLOHY 3](#_Toc96976205)

[2. IMPLEMENTÁCIA 4](#_Toc96976206)

[3. ANALÝZA VO WIRESHARKU 5](#_Toc96976207)

[3.1. Štruktúra hlavičky 5](#_Toc96976208)

[3.2. Metóda kontrolnej sumy (checksum) 6](#_Toc96976209)

[3.3. Simulácia chyby 6](#_Toc96976210)

[3.4. Keep-alive metóda 6](#_Toc96976211)

[4. TESTOVANIE 7](#_Toc96976212)

[5. ZÁVER 7](#_Toc96976213)

**DOKUMENTÁCIA**

## ZADANIE ÚLOHY

Zadanie 1 – SIP Proxy (telefónna ústredňa)

Hlavná myšlienka zadania:

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

Doplňujúce informácie k zadaniu:

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 funkcionalitu SIP Proxy. To znamená, že **nemôžete** zobrať existujúcu SIP Proxy ako napr. Asterisk, kde len skompilujete alebo priamo spustíte cudziu binárku… Hovor **musí** byť realizovaný medzi dvomi **fyzickými** zariadeniami v rámci LAN siete.

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)

Ak sú splnené **všetky** tieto podmienky, študent získava 5 bodov, ktoré sú minimom na absolvovanie tohoto zadania.

Doplnkové funkcionality (ktoré môžete, ale nemusíte urobiť):

* 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”

## IMPLEMENTÁCIA

Zadanie bolo implementované v programovacom jazyku python 3.9 a bola použitá knižnica z <https://github.com/tirfil/PySipFullProxy>, konkrétne súbor sipfullproxy.py, ktorý je importnutý do hlavného programu *main.py*. Knižnica sipfullproxy.py je implementovaná ako UDP server, ktorý slúži ako SIP Proxy a spracováva požiadavky, ktoré dostáva. UDP server je implementovaný ako trieda, ktorá má funkcie, ktoré spracovávajú konkrétne požiadavky. Napríklad funkcia processInvite spracováva požiadavku INVITE v protokole SIP.

Knižnica sipfullproxy.py bola upravená, aby fungovala aj na novšej verzii pythonu konkrétne na python 3.9. Ďalej boli odstránené všetky riadky spojené s logovaním do log súboru. Do knižnice bola pridaná funkcia handle\_topvia\_recordroute\_file, ktorá naplní top\_via a record\_route informáciami a z hlavnej funkcie získa meno súboru, kde budú logované registrácie klientov. Ďalej vo funkcii processRegister bolo pridané zapisovanie do súboru, kde budú vypísané registrovania klientov. V knižnici boli taktiež zmenené stavové kódy na svoje vlastné ako je ukázané aj na obrázku 1.

Hlavný program ( súbor main.py) bol implementovaný na to, aby zistil svoju ip svojho PC (servera proxy SIP) a následne zavolal UDP server z knižnice sipfullproxy, ktorý bude bežať do nekonečna, až kým sa neskončí program.

Obrázok, na ktorom je text

Automaticky generovaný popis

Obrázok 1 zmenené kódy a zapisovanie registrácie

Obrázok, na ktorom je text

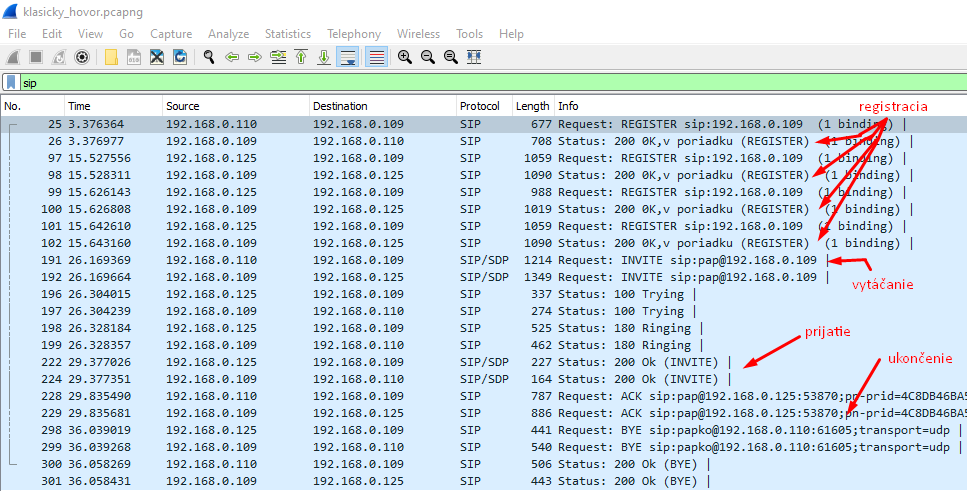
Automaticky generovaný popis

Obrázok 2 hlavný program ( funkcia main)

## ANALÝZA VO WIRESHARKU

Program bol zanalyzovaný v programe Wireshark na ethernete. V rovnakej LAN sieti bol pripojený PC, na ktorom bežal program SIP proxy a ostatné 3 zariadenia, ktoré sa pripájali na tento server pomocou klienta linphone. 3 zariadenia boli pripojené z dôvodu analýzy konferenčného hovoru, teda hovoru, kde medzi sebou komunikujú aspoň 3 zariadenia.

Vo wiresharku boli odchytené pakety pri klasickom hovore, konferenčnom hovore a pri videohovore.



Obrázok 3 klasický hovor

Obrázok, na ktorom je stôl

Automaticky generovaný popis

Obrázok 4 videohovor

Obrázok, na ktorom je stôl

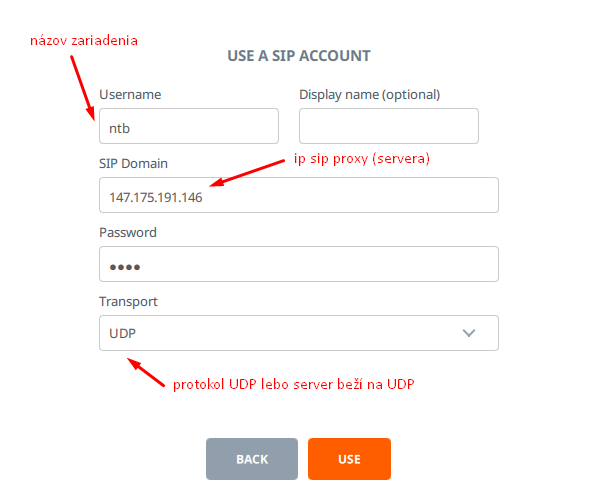
Automaticky generovaný popis

Obrázok 5 konferenčný hovor medzi 3 zariadeniami

Ako sa dá z obrázkov 3,4,5 vidieť pri video hovore a konferenčnom hovore je zachytených omnoho viac paketov ako pri klasickom hovore.

## TESTOVANIE

Testovanie hovorov prebehlo pomocou linphone klienta. V linphone klientovi bolo treba zadať username, SIP domain ( ip, kde nám beží server) a nejaké heslo. Následne sa naše zariadenie objavilo pod adresou username@ip\_servera – konkrétne [ntb@147.175.191.146](mailto:ntb@147.175.191.146) pre obrázok 6.



Obrázok 6 nastavenie sip účtu

## ZÁVER

Program bol úspešne implementovaný v pythone. Program bol testovaný na linphone, kde bol úspešne vykonaný najprv klasicky hovor medzi dvomi zariadeniami v LAN sieti, následne bol vykonaný videohovor a na konci bol vykonaný konferenčný hovor medzi 3 zariadeniami. Všetky tieto hovory boli odchytené vo wiresharku a následne zanalyzované. Denník hovorov nebol implementovaný, bolo len implementované denník registrácie na SIP proxy, ktorý je uložený v súbore history.txt. Úspešne bolo implementovaná aj úprava stavových kódov, kde napr. namiesto OK zahlási OK, v poriadku. Možnosť presmerovať hovor funguje taktiež len táto komunikácia nebola odchytená vo wiresharku.