

Zadanie 1 – SIP Proxy (telefónna ústredňa) - dokumentácia

Autor: Marek Oravec

AIS ID: 103077

Hlavná myšlienka zadania:

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

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.

Doplňkové 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”

Každá doplnková funkcionality predstavuje plus 1 bod.

Počas prezentácie zadania musíte byť schopní na zariadení, kde beží ústredňa urobiť SIP trace a otvoriť ho pomocou tcpdump alebo Wireshark, a v primeranom rozsahu vysvetliť cvičiacemu, ako daná signalizácia prebieha.

Forma odovzdania:

- Dokumentácia, kde opíšete, ako ste vaše riešenie implementovali, aké knižnice ste použili, a ako ste ústredňu sprevádzkovali, vo formáte PDF do miesta odovzdania v AIS. V PDF dokumente musí byť odkaz na váš repozitár v Gite (verejný repozitár, ľubovoľná služba)
- Vo vašom repozitári sa musí nachádzať PCAP trace z každého scenára, ktorý vaša implementácia poskytuje.

Opis implementácie:

Pre účely splnenia tohto zadania som použil „knižnicu“ z GitHubu:

<https://github.com/tirfil/PySipFullProxy/blob/master/sipfullproxy.py>

Tento zdrojový kód nie je knižnica sama o sebe, ale ja som k nej tak pristupoval. V kóde bolo nutné prispôsobiť určité časti aktuálnej verzii Pythonu, pretože originálna verzia obsahovala už nie fungujúce prvky. (napr. pri výbere kľúčov zo slovníka bola použitá metóda **has_key()**, ktorá už v aktuálnej verzii Pythonu nefunguje, preto bolo nutné ju nahradiť)

Celé riešenie bolo závislé na triede **UDPHandler**, v ktorej sa nachádzali overrideované metódy pre funkciu programu. (výpis stavových kódov a pod.)

Sada mnou použitých stavových kódov bola inšpirovaná spoločnosťou Apple. Je ich tam pomerne veľa, ale reálne sa pre účely tohto zadania stretneme len s niekoľkými, ako sú napríklad „**200 V poriadku**“ a pod.

Ústredňa funguje pomocou môjho mobilného telefónu a počítača, na ktorom spustím ústredňu, a po zadaní správnych IP adries som schopný **hovoru** aj **videohovoru**.

Zhodnotenie:

Výsledkom mojej práce je táto dokumentácia, môj kód zverejnený na GitHub repozitári spolu so 4 Wireshark/tcpdump súbormi, ktoré zahŕňajú všetky scenáre z povinných funkcionalít a z doplnkových **videohovor** a **úprava SIP stavových kódov**.

Link na repozitár:

https://github.com/Pucko99/fiit_mtaa_oravec

