

SIP Proxy - telefónna ústredňa

Dokumentácia

Adrián Ondov

Marec 2022

1 Znenie zadania

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 funkcionality 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 dvoma 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

- 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”

2 Použitá knižnica a použitý SIP klient

Na realizáciu zadania bola použitá knižnica **PySipFullProxy** [1]. Knižnica nebola použitá tradičným spôsobom, keďže momentálne nie je možné jej priame stiahnutie a implementovanie v jazyku Python. Naopak bola implementovaná ako dodatočný .py súbor, z ktorého boli použité už existujúce funkcie. Niektoré z nich boli v minimálnej miere upravené pre potreby splnenia povinnej, alebo doplnkovej funkcionality. Na realizáciu samotných hovorov bol použitý softvér **Linphone**.

Výsledný program a súbory s odchytenými paketmi je možné nájsť na Github repozitári na nasledujúcom linku: https://github.com/aondov/MTAA_Z1_Ondov

3 Povinná funkcionality programu

Funkcionality je rozdelená do viacerých typov správ, ktoré sa pri protokole SIP používajú. Tie budú opísané vo vlastných sekciách. Je nutné poznamenať, že pre komunikáciu akéhokoľvek typu bol použitý **protokol UDP** a predvolený port pre SIP protokol, teda **port 5060**. Hovory boli realizované buď medzi dvoma mobilnými zariadeniami (tablet a mobil), alebo medzi mobilným zariadením a osobitným notebookom.

3.1 Registrácia

Program plne podporuje registráciu nových zariadení do ústredne. Zariadenie pritom musí byť správne nakonfigurované, aby používalo primárne SIP Proxy server. V tomto prípade je SIP Proxy server reprezentovaný vytvorenou ústredňou. Jej IP adresa bola nakonfigurovaná v zariadeniach, ktoré sa na ňu pripájali. Tie sa pravidelne snažia na ústredňu pripojiť pomocou **REGISTER správ**. Ak ústredňa správu prijme, odpovie na ňu správou s **kódom 200 OK**, čím potvrdí úspešnú registráciu zariadenia. Následne si ústredňa uloží potrebné informácie o registrovanom zariadení (v tomto prípade do bežného listu). Program sa na tento list odkazuje počas spojenia dvoch zariadení, pričom odtiaľ získava potrebné informácie.

3.2 Pozývanie do hovoru a zvonenie

Ak sa jedno z registrovaných zariadení snaží kontaktovať iné registrované zariadenie, musí najprv odoslať tzv. **INVITE správu**, ktorou pozve požadované zariadenie do hovoru. Táto správa je preposlaná cez ústredňu na cieľové zariadenie. Ak INVITE správa nie je odmietnutá ústredňou, zariadenie pokračuje v pokuse o ustanovenie hovoru. Status odosielateľa sa mení na status **Trying**, čo znamená, že sa snaží dostať až k cieľovému zariadeniu. Keď je toto spojenie úspešné, mení sa status na **Ringing**, takže cieľové zariadenie začne vyzvárať. Ďalšie kroky závisia už iba od reakcie cieľového zariadenia na zvonenie.

3.3 Volanie

Počas vyzvãania môže cieľové zariadenie hovor buď odmietnuť, alebo prijať. Ak ho odmietne, tak odosielateľ dostane správu **Decline**, na ktorú sa ešte odpovedá správou **ACK**, teda potvrdením na odpoveď cieľového zariadenia. Ak ale cieľové zariadenie prijme hovor, odošle sa zdrojovému zariadeniu potvrdenie cez správu **200 OK**. Tým je spojenie ustanovené a môžu sa začať prenášať pakety, ktoré nesú hlasové informácie. Pravidelným posielaním správ 200 OK od zdrojového zariadenia a ACK odpoveďami od cieľového zariadenia sa realizuje keep-alive proces, ktorý potvrdzuje, že obe zariadenia sú stále aktívne zapojené do komunikácie.

3.4 Ukončenie hovoru

Prebiehajúci hovor sa ukončuje správou **BYE**, ktorá je odoslaná zo zariadenia, ktoré ukončilo spojenie ako prvé. Zariadenie na druhej strane následne odpovie správou **ACK**, čím sa spojenie považuje za ukončené a hovor už ďalej neprebieha.

4 Doplnková funkcionalita programu

4.1 Videohovor

Program podporuje aj prenos videa popri prenose hlasových informácií. Zariadenia si posielajú video informácie po dohode cez **protokol SDP (Session Description Protocol)**, ktorý ustanovuje konkrétne parametre pre prenos audio informácií, ale aj video informácií (napr. kodeky). Po dohode použitých parametrov pre komunikáciu sa následne tieto parametre aplikujú v oboch koncových zariadeniach a podľa nich sú dané médiá posielané po sieti.

4.2 Konferečný hovor

Program taktiež podporuje konferenčný hovor medzi viacerými zariadeniami, ktoré medzi sebou môžu komunikovať v skupine. Proces je podobný ako pri bežnom hovore - zariadenia sa registrujú, jedno z nich potom pozve ostatné zariadenia do hovoru cez správu INVITE, prejde sa do stavu Trying a Ringing, až nakoniec cieľové zariadenia hovor prijmu. Pri ukončení je znova použitá správa BYE, na ktorú sa odpovie so správou ACK.

5 Odchytené SIP pakety

Pre každý scenár boli za pomoci programu Wireshark odchytené SIP pakety, ktoré prenášali spomínané správy. Súbor opisujú nasledovné scenáre:

- **"register_Ondov.pcapng"** - súbor obsahuje REGISTER správy z osobitnej komunikácie dvoch zariadení

- **"invite_trying_ringing_Ondov.pcapng"** - súbor obsahuje správy o statusoch Trying, Ringing a taktiež správy INVITE z osobitnej komunikácie dvoch zariadení
- **"calling_endcall_Ondov.pcapng"** - súbor obsahuje priebeh a ukončenie komunikácie medzi dvoma zariadeniami z pohľadu protokolu SIP, teda je možné v ňom nájsť aj BYE správy
- **"decline_Ondov.pcapng"** - súbor opisuje scenár, kedy cieľové zariadenie odmietne hovor
- **"videocall_Ondov.pcapng"** - súbor opisuje scenár, kedy dve zariadenia používajú na komunikáciu aj video, nie iba audio
- **"conference_call_Ondov.pcapng"** - súbor opisuje scenár, kedy sú v komunikácii zapojené až tri zariadenia, teda ide o konferenčný hovor
- **"full_comm_Ondov.pcapng"** - súbor obsahuje kompletnú komunikáciu medzi dvoma zariadeniami (správy REGISTER, INVITE, Trying, Ringing,...)

6 Úprava pôvodnej knižnice

Použitá knižnica bola pre účely tohto zadania jemne upravená, aby boli splnené potrebné požiadavky. Prvou zo zmien bola **úprava kódov odpovedí** na rôzne udalosti na serveri. V súbore bol vytvorený slovník, ktorý obsahuje nové kódy, resp. ktorý obsahuje nový opis týchto kódov. Číslovanie odpovedí totiž ostalo nezmenené, upravené boli iba ich slovné popisy.

Druhou úpravou bola zmena syntaxe pre funkciu **join** pri znakových reťazcoch. V pôvodnom súbore bola použitá zastaralá syntax, preto bolo nutné ju upraviť tak, aby nespôsobovala v programe neželané chyby.

Treťou úpravou bolo pridanie vlastnej funkcie, ktorá nastavuje globálne premenné **record_route** a **top_via**. Táto funkcia bola potrebná z dôvodu, že v pôvodnom kóde boli tieto premenné nastavené v rovnakom súbore z funkcie **main**. Keďže bola funkcia **main** presunutá do iného súboru, je nutné k týmto premenným pristupovať korektne, teda cez volanie použitej funkcie.

Poslednou úpravou bolo odstránenie pôvodnej **main** funkcie a jej presun do osobitného .py súboru. Main funkcia bola manuálne prepísaná, aby sa nejednalo iba o čisté kopírovanie tej pôvodnej.

Ďalšie zmeny v pôvodnom súbore sa týkali len minimálnych opráv kódu. Tieto zmeny sa realizovali iba v prípade, že program vyhadzoval chybné správy, čo sa našťastie udialo len veľmi zriedkavo.

7 Záver

Program poskytuje všetky základné funkcionality pre korektné fungovanie protokolu SIP a protokolu SDP, prípadne iných protokolov, ktoré sa podieľajú na VoIP komunikácii. Zariadenia sa vedia úspešne registrovať do ústredne a vedia sa pomocou stanoveného procesu navzájom kontaktovať. Výmena audio/video informácií je taktiež plne funkčná. Spojenie sa počas testovania javilo byť stabilné a nedochádzalo k žiadnej povšimnuteľnej strate paketov. Z toho je možné usúdiť, že zadanie spĺňa minimálne (a niektoré doplnkové) požiadavky na dostatočne vysokú úroveň na to, aby bolo možné realizovať VoIP komunikáciu medzi dvoma, alebo viacerými koncovými zariadeniami.

Literatúra

- [1] Philippe Thirion. Pysipfullproxy. <https://github.com/tirfil/PySipFullProxy>, 2014.