

Zadanie 1 – SIP Proxy

Mobilné technológie a aplikácie

Alex Macala

Slovenská technická univerzita v Bratislave

Fakulta informatiky a informačných technológií

xmacala@stuba.sk

Zadanie:

Hlavná myšlienka zadania:

Na vašom počítači (alebo virtuálnom počítači) sprevádzajte 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)
- Vytocenie hovoru a zvonenie na druhej strane
- Prijatie hovoru druhou stranou, fungujúci hlasový hovor
- Ukončenie hlasového hovoru (prijatého aj neprijatého)

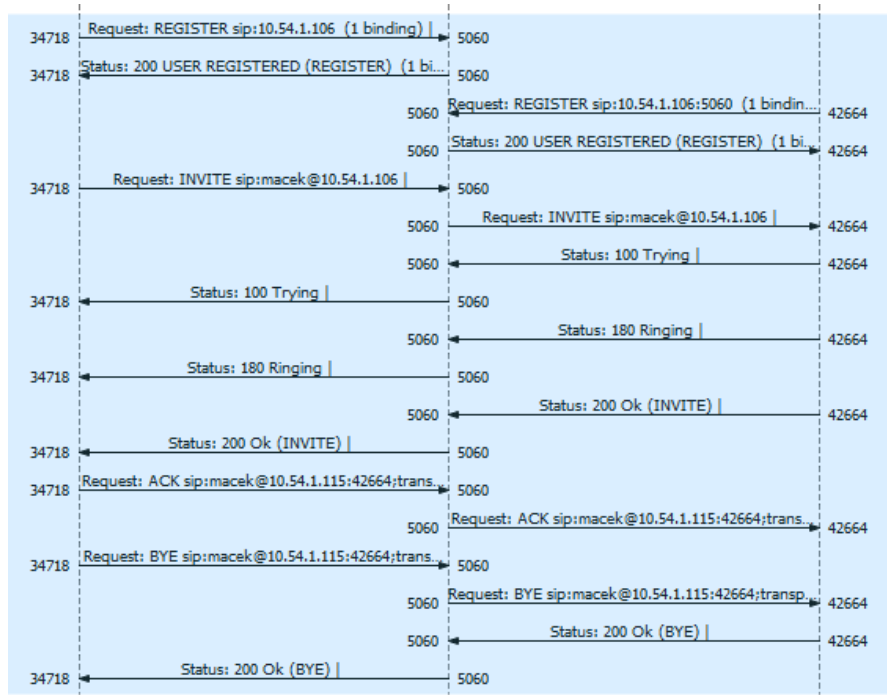
Doplňkové funkcionality:

- Možnosť zrealizovať konferenčný hovor (aspoň 3 účastníci)
- Možnosť presmerovať hovor
- Možnosť realizovať video hovor
- 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:

V programe som implementoval knižnicu sipfullproxy ktorú som našiel na githube (<https://github.com/tirfil/PySipFullProxy/blob/master/sipfullproxy.py>) v ktorej boli použité tieto knižnice: logging, socketserver, re a knižnicu time. Implementoval som ju tak že som ju použil v svojom main.py do ktorého som nakopíroval a následne upravil main z danej knižnice.

Povinné funkcionality som splnil všetky vid' obrázok:



Doplňkové funkcionality som splnil taktiež všetky, prvé tri budú znázornené pomocou .pcap súborov, logovanie už v danej knižnici bolo aplikované len tam nebolo to ktoré sa od nás vyžaduje tak to som tam pridal a tie ktoré tam boli som odstránil a úpravu sip stavových kódov som urobil tak že som stavový kód kde vypisovalo REGISTER pozmenil na USER REGISTERED.

Zmeny ktoré som aplikoval v danej knižnici:

- Import SocketServer som zmenil na import socketserver
- Dictionary.has_key(key) som zmenil na key in dictionary
- Väčšinu premenných „text“ som zmenil na text.encode(„utf-“)
- V def handle(self): data = self.request[0] som zmenil na data = self.request[0].decode(„utf-8“)
- Vymazanie nepotrebných logging.info
- Vamazanie nepotrebných premenných

Odkaz na môj github: <https://github.com/xmacala/MTAA>

Na záver by som dodal že keď som toto zadanie videl prvý krát myslel som si že to je blbosť že máme za úlohu trochu prerobiť nejaký kód ktorý už je vymyslený no potom som zistil že to nie je len tak trochu ale bolo to treba pochopiť no myslím si že sa mi to podarilo.