

Síťové aplikace a správa sítí
Programování síťové služby – POP3 server

Obsah

1	Úvod	2
2	Uvedení do problematiky	2
2.1	IMF - Internet message format	2
2.2	Maildir	2
2.3	Odeslání zprávy klientovi	2
2.4	POP3 příkazy	3
2.5	stavy serveru	4
3	Návrh programu	4
4	Implementace	4
5	Návod k použití	4
5.1	Překlad aplikace	5
5.2	Spuštění serveru	5
5.3	Příklady	5
6	Závěr	5

1 Úvod

Program popser je POP3 server umožňující připojení více klientů najednou. Klienti mohou komunikovat se serverem pomocí protokolu POP3. Server pracuje s emaily ve formátu IMF v adresářové struktuře typu Maildir. Server pracuje pouze s jednou poštovní schránkou, do které může vstoupit pouze jeden uživatel.

2 Uvedení do problematiky

V této části dokumentu je stručně popsán formát IMF a Maildir. Dále je zde popsána komunikace a funkčnost POP3 serveru.

2.1 IMF - Internet message format

Internetová zpráva ve formátu IMF není nic jiného, než série znaků. Znaků mohou nabývat US-ASCII hodnot 1-127 [2].

Zprávu tvoří jednotlivé řádky. Každá řádka je ukončena dvěma po sobě jdoucími znaky - CR (ASCII hodnota 13) a LF (ASCII hodnota 10) [2].

Zpráva se dále skládá z hlavičky, po které může následovat tělo. Hlavička je sekvence řádků. Na každém řádku hlavičky je název pole a jeho hodnota oddělená ":". Hlavička je ukončena prázdným řádkem. Tělo je opět sekvence řádků [2].

2.2 Maildir

Adresářová struktura Maildir je vhodná pro třídění emailů. Každý uživatel má vlastní složku Maildir a v ní jsou složky cur, new a tmp. Se složkou tmp pracuje pouze SMTP server.

2.3 Odeslání zprávy klientovi

Pokud klient použije jeden z příkazů k získání emailu, POP3 server daný email odešle. Aby klient věděl, kdy server odeslal celou zprávu, odešle na konci emailu osamocenou tečku na řádku a dále nic neposílá. Pokud řádek zprávy začíná tečkou, server přidá další tečku na začátek řádku.

2.4 POP3 příkazy

Níže jsou uvedeny všechny možné dotazy, včetně volitelných, na POP3 server. Všechny tyto příkazy jsou implementovány v programu popser.

- USER jmeno_uzivatele
- PASS heslo
- APOP jmeno_uzivatele heslo
- STAT
- LIST [message_id]
- RETR message_id
- DELE message_id
- NOOP
- RSET
- QUIT
- TOP message_id lines
- UIDL [message_id]

Pro přihlášení na server slouží příkazy USER, PASS a APOP. APOP využívá md5 k zakódování hesla. Klíč obdrží od serveru na začátku spojení a ke klíči připojí své heslo a výsledný řetězec znaků zašifruje md5 algoritmem. USER a PASS slouží k nešifrovanému přenosu. Pokud server podporuje APOP, nesmí podporovat příkazy USER ani PASS a obráceně. V programu popser lze přepínat mezi podporou těchto příkazů pomocí argumentů programu při jeho spuštění [1].

K přečtení zprávy lze použít příkazy TOP a RETR. Pomocí příkazu TOP klient získá hlavičku zprávy a předem zadaný počet řádků zprávy (parametr lines), pokud je zadané větší číslo, než je celkový počet řádků, odešle se celý email [1].

K získání identifikátorů emailů lze použít příkazy LIST a UIDL. List vypíše všechny emaily, které nejsou označeny ke smazání a každému přidělí ID a vedle ID vypíše jeho velikost v oktetech. UIDL k ID zprávám vypíše jejich unikátní ID v rámci celého Maildiru. ID zpráv se na rozdíl od tohoto unikátního ID může měnit (ID je unikátní pouze v rámci jednoho sezení) [1].

Pro zjištění velikosti Maildiru (bez složky tmp) lze použít příkaz STAT. Uvedená velikost je v oktetech [1].

DELE slouží k mazání zpráv. Zprávy se mažou až po zadání příkazu QUIT. Do té doby je možné obnovit zprávy označené ke smazání příkazem RSET [1].

Na příkaz NOOP server pouze odpoví [1].

2.5 stavy serveru

POP3 server má 3 základní stavy.

- AUTHORIZATION
- TRANSACTION
- UPDATE

Ve stavu AUTHORIZATION lze pouze používat příkazy USER, PASS, APOP a QUIT, tento stav slouží k autorizaci uživatele, poté co se uživatel prokáže platnými přihlašovacími údaji, přejde se do stavu TRANSACTION. Ve stavu TRANSACTION lze použít všechny příkazy, vyjma USER, PASS a APOP. Stav TRANSACTION trvá dokud klient nezadá QUIT, poté se přejde do stavu UPDATE. Ve stavu UPDATE se provede mazání zpráv a následné ukončení komunikace [1].

3 Návrh programu

Po připojení klienta na server, aplikace popser odešle uvítací zprávu s klíčem pro šifrování. Klíč se nachází na konci zprávy a je ve formátu `<process-ID.clock.socket_descriptor@hostname>`. Následně program očekává příkaz APOP nebo USER a PASS, podle argumentů programu. Po úspěšném přihlášení uživatele server přistoupí k Maildir a přesune všechny zprávy ze složky new do cur. Následně server očekává další příkazy od klienta. Pokud klient zadá příkaz QUIT, server smaže všechny soubory označené ke smazání. Pokud se klient odpojí bez užití příkazu QUIT nebo se spojení přeruší, emaily označené ke smazání zůstanou nedotčeny.

4 Implementace

Program je implementován v jazyce C a C++. Program využívá objektového programování, má deklarované třídy Mail_file, Mail_dir a Arguments a využívá některé objekty i ze standardních knihoven C++, například vector nebo string. Pro md5 algoritmus byla použita třída md5 [3].

Aplikace je vytvořena pro operační systém Linux a byla vyvíjena na Ubuntu 16.04 LTS a testována na systémech Ubuntu 16.04 LTS a CentOS 7.4

Program je překládán pomocí překladače g++ a pro překlad slouží soubor Makefile.

5 Návod k použití

V této části dokumentu je uveden návod k použití a pár příkladů užití.

5.1 Překlad aplikace

Aplikaci lze přeložit příkazem `make` ve složce se souborem `Makefile`.

5.2 Spuštění serveru

Server má 3 režimy běhu.

- `popser -r`
- `popser -h`
- `popser -a AUTH_FILE -d MAILDIR -p PORT [-c] [-r]`

Argument `-h` vypíše stručný návod k obsluze programu. Argument `-r` uvede složku `Maildir` do původního stavu (kromě obnovení smazaných emailů). `AUTH_FILE` je cesta k souboru s uživatelským jménem a heslem. `MAILDIR` je cesta ke složce, která obsahuje minimálně složky `new` a `cur`. S touto složkou bude následně `popser` pracovat. `PORT` je číslo portu, na kterém se budou moci připojit klienti. Pokud je zadán parametr `-c`, tak se přihlašovací údaje zadávají pomocí `USER` a `PASS` a heslo je na síti přenášeno nešifrovaně. Pokud `-c` zadán není, autorizace probíhá pomocí příkazu `APOP`.

5.3 Příklady

- `popser -a ./user.txt -p 8888 -d ./Maildir/`

Ze souboru `./user.txt` budou načteny uživatelské údaje. Ve složce `./Maildir/` budou očekávány složky `cur` a `new` ve kterých se budou nacházet soubory reprezentující emaily. Na portu 8888 bude probíhat komunikace a autorizovat se bude muset uživatel pomocí příkazu `APOP`.

- `popser -r`

Soubor uvede `Maildir` do původního stavu, tj. zvrátí veškeré operace se soubory (kromě odstranění).

6 Závěr

Aplikaci `popser` lze použít jako POP3 server pro správu jediného účtu. Jsou v něm implementovány všechny povinné i volitelné funkcionality a byl vyvinut pro operační systém Linux.

Reference

- [1] JOHN G. MYERS, M. T. R. *Post Office Protocol - Version 3* [online]. Poslední změna 1996 [vid. 17. listopadu 2017]. Dostupné z: <<https://tools.ietf.org/html/rfc1939>>.
- [2] RESNICK, P. W. *Internet Message Format* [online]. Poslední změna 2008 [vid. 17. listopadu 2017]. Dostupné z: <<https://tools.ietf.org/html/rfc5322>>.
- [3] THILO, F. *C++ md5 function* [online]. Poslední změna 1991 [vid. 17. listopadu 2017]. Dostupné z: <<http://www.zedwood.com/article/cpp-md5-function>>.