

**Protokoly POP3 a IMAP4**  
(referát)

---

**Predmet:** Počítačové siete III  
**Autor:** Marek Kubran  
**Školský rok:** 2001/02

# OBSAH

<b>Post Office Protocol – Version 3 .....</b>	<b>3</b>
Úvod.....	3
Základná charakteristika.....	3
Stav AUTHORIZATION .....	4
Príkazy poskytované serverom v transakčnom stave .....	5
Stav UPDATE.....	7
Doplňkové POP3 príkazy .....	7
Bezpečnostné riziká protokolu .....	8
<b>Internet Message Access Protocol - Version 4 .....</b>	<b>9</b>
Úvod.....	9
Základná charakteristika.....	9
Stavový diagram.....	9
Formáty údajov .....	10
Názvy schránok.....	10
Príkazy klienta.....	10
Príkazy klienta v ľubovoľnom stave .....	11
Príkazy klienta v stave NON-AUTHENTICATED .....	11
Príkazy klienta v stave AUTHENTICATED .....	11
Príkazy klienta v stave SELECTED.....	13
Experimentálne a rozšírené príkazy klienta.....	15
Bezpečnostné riziká protokolu .....	15
Revízia 1 protokolu IMAP4 .....	15
<b>Poštovní klienti a WEB mail.....</b>	<b>16</b>
<b>Použitá literatúra .....</b>	<b>17</b>

# Post Office Protocol – Version 3

## Úvod

POP3 protokol sa používa na prístup k poštovej schránke na vzdialenom počítači (napr. pracovná stanica číta poštu zo servera, na ktorom je uložená prichádzajúca elektronická pošta). To má výhodu, že počítač nemusí byť nepretržite pripojený na počítačovú sieť, aby dostával elektronickú poštu. Elektronická pošta prichádza na server, ktorý vykonáva funkciu MTS (Message Transport System) a poskytuje služby pre prístup do odkazovej schránky (POP3 server). POP3 protokol teda dovoľuje pracovným staniciam získavať poštu zo servera, ktorý ju uchováva.

POP3 protokol nie je určený na rozsiahlu manipuláciu s poštou na serveri, bežne sa používa iba na načítanie pošty a jej následné vymazanie. Rozsiahlejšie a všeobecnejšie možnosti poskytuje zložitejší protokol IMAP4.

POP3 protokol je protokol založený na architektúre klient/server. V ďalšom texte sa bude pojmom **klient** označovať počítač, ktorý využíva služby protokolu POP3 a pojmom **server** počítač, ktorý služby POP3 poskytuje. V príkladoch bude klient označený C: a server S:.

## Základná charakteristika

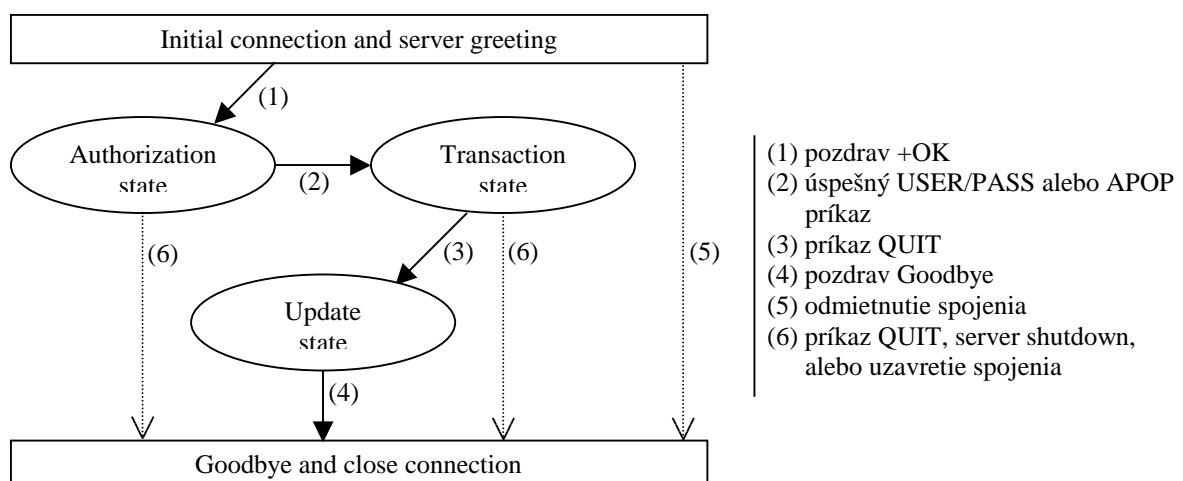
POP3 protokol na spojenie klienta so serverom používa TCP. Server počúva na porte 110. Po pripojení sa klienta, server pošle pozdrav (napr. +OK POP3 server ready). Klient a server si vymieňajú príkazy a odpovede až kým sa neuzavrie alebo nezruší spojenie.

Príkazy klienta pozostávajú z kľúčových slov (nerozlišujú sa veľké a malé písmena), za ktorými nasledujú prípadné argumenty. Všetky príkazy sú ukončené dvojicou znakov CR-LF. Príkazy sa skladajú iba z viditeľných ASCII znakov, argumenty sú od seba oddelené medzerou. Každý argument je dlhý max. 40 znakov.

Odpovede servera pozostávajú z indikátora stavu (+OK, -ERR), kľúčového slova príkazu a prípadných ďalších informácií. Odpovede môžu byť dlhé až 512 znakov a sú ukončené CRLF.

Odpovede na určité príkazy môžu byť viacriadkové. V tomto prípade po odoslaní prvého riadku a CRLF nasledujú všetky ostatné riadky, každý ukončený párom CRLF. Po odoslaní posledného riadku odpovede sa zašle ukončujúci znak (‘.’, 46 decimálny kód) a CRLF. Ak niektorý riadok odpovede začína znakom ‘.’, na začiatok riadku sa vloží ešte raz ukončovacia znak. Takže viacriadková odpoveď sa vždy končí päťcou znakov “CRLF.CRLF“. Pri získavaní viacriadkovej odpovede klient kontroluje, či riadok začína ukončovacím znakom. Ak áno a nasledujúca dvojica znakov je iná ako CRLF, prvý znak riadku (ukončovacia znak ‘.’) sa odsekne. Ak riadok začína ukončovacím znakom a za ním hneď nasleduje CRLF, odpoveď z POP3 servera je ukončená (posledné “CRLF“ nie je súčasťou viacriadkovej odpovede).

POP3 relácia (*session*) sa počas doby svojej existencie (od nadviazania spojenia až po jeho ukončenie) nachádza v niekoľkých stavoch.



Obr. Stavy v POP3 relácii

Po nadviazaní TCP spojenia (otvorenie relácie) a odoslání pozdravu POP3 serverom sa relácia dostáva do stavu AUTHORIZATION. V tomto štádiu sa musí klient identifikovať. Po úspešnej autentifikácii server zabezpečí prístup k odkazovej schránke (INBOX klienta) a relácia sa dostane do stavu TRANSACTION. V tomto stave klient môže požadovať akcie zo strany POP3 servera. Keď klient zadá príkaz QUIT, relácia sa dostane do stavu UPDATE – POP3 server uvoľní prostriedky, ktorými zabezpečoval prístup k odkazovej schránke klienta a odpovie „goodbye“. TCP spojenie sa následne uzavrie.

Na nerozpoznané, neimplementované alebo syntakticky nesprávne príkazy server odpovedá negatívnou odpoveďou (indikátor stavu –ERR). Podobne aj v situácii, keď klient zadá príkaz v nesprávnom stave POP3 relácie, server odpovie negatívne. Klient nemá možnosť zistiť, či server neimplementuje doplnkový (*optional*) príkaz alebo server ho odmieta vykonať, prípadne ho nie je schopný vykonať.

POP3 server môže mať nastavený časovač automatického odhlásenia po určitom čase neaktívnosti klienta (*inactivity autologout timer*). Prijatie ľubovoľného príkazu od klienta anuluje časovač (*reset autologout timer*). Ak uplynie nastavený čas, relácia neprejde do stavu UPDATE - server by mal zrušiť spojenie bez vymazania odkazov z poštovej schránky a bez zaslania odpovede klientovi.

## Stav AUTHORIZATION

Po otvorení TCP spojenia POP3 klientom, server pošle jednoriadkovú správu s pozdravom. Ním môže byť ľubovoľná pozitívna odpoveď, napr.:

```
S:      +OK POP3 server ready
```

Relácia sa nachádza v stave AUTHORIZATION. Klient sa teraz musí autentifikovať. Na to mu slúži niekoľko mechanizmov, napr. kombinácia príkazov USER a PASS alebo príkaz APOP alebo niektorý z ďalších mechanizmov, ktoré sú opísané v dokumente RFC-1734. POP3 server musí podporovať aspoň jeden z autentifikačných mechanizmov.

## Autorizácia pomocou kombinácie USER/PASS

Syntax: USER meno  
PASS heslo

Klient zašle príkaz USER s parametrom meno. Meno je reťazec, ktorý identifikuje poštovú schránku (*mailbox*). Následne klient zadá príkaz PASS s argumentom typu reťazec, ktorý predstavuje heslo. Heslo sa prenáša v nešifrovanej podobe ⇒ nebezpečie odchytenia.

Server odpovie buď pozitívne (+OK) alebo negatívne (–ERR) v závislosti, či je kombinácia mena a hesla správna, či používateľ existuje a zároveň či sa podarilo serveru uzamknúť prístup k schránke.

```
Príklad:  C: USER Gates
          S: +OK name is a valid mailbox
          C: PASS microsoft
          S: -ERR maildrop already locked
          ...
          C: USER mrose
          S: +OK password required for mrose
          C: PASS tanstaaf
          S: +OK mrose's maildrop has 2 messages (320 octets)
```

□

## Autorizácia pomocou príkazu APOP

Syntax: APOP meno číslice

Argument meno je rovnaké ako pri príkaze USER, číslice je číselný reťazec MD5 pozostávajúci zo 16 číslic v šestnástkovej sústave (lower case ASCII). Autorizácia pomocou príkazu APOP prináša zvýšenú bezpečnosť prihlasovania sa klienta, nakoľko heslo sa prenáša už v šifrovanej forme.

POP3 server, ktorý v sebe implementuje príkaz APOP, pri každom pripojení sa klienta zahrmie do pozdravu aj časový údaj (*timestamp*). Časový údaj sa zhoduje s *msg-id* podľa RFC-822 a musí byť zakaždým rôzny (vždy keď sa klient pripojí na POP3 server, časový údaj je unikátny). Napr. na UNIX implementácii kedy sa pre každú inštanciu POP3 servera používa separátny proces, syntax časového údaju môže byť:

`<PID.clock@hostname>`

kde `PID` je identifikačné číslo procesu, `clock` je hodnota systémového času v desiatkovej sústave a `hostname` je plné doménové meno servera, na ktorom sa vykonáva POP3 server.

Klient si poznačí tento časový údaj a parameter číslice príkazu APOP vypočíta tak, že aplikuje algoritmus MD5 na tento reťazec (časový údaj), za ktorým nasleduje heslo. Heslo pozná iba klient a POP3 server.

Keď POP3 server prijme príkaz APOP, zverifikuje obidva parametre. Ak parameter číslice je správny, server pošle pozitívnu odpoveď a POP3 relácia sa dostane do stavu TRANSACTION. Ináč sa pošle negatívna odpoveď a server zostáva v stave autorizácie.

Príklad:     S: +OK POP3 server ready <1896.697170952@dbc.mtview.ca.us>  
              C: APOP mrose c4c9334bac560ecc979e58001b3e22fb  
              S: +OK mailbox has 1 message (369 octets)

V tejto ukážke bolo heslo `tanstaaf`. Takže MD5 algoritmus sa aplikoval na reťazec

`<1896.697170952@dbc.mtview.ca.us>tanstaaf`

čo dáva výsledný parameter číslice s hodnotou

`c4c9334bac560ecc979e58001b3e22fb`

□

Po úspešnej autentifikácii POP3 server sprístupní poštovú schránku klientovi a relácia sa dostane do stavu TRANSACTION. Prístup k schránke je exkluzívny, pričom schránka je uzamknutá, aby sa nemohli odkazy modifikovať alebo vymazať. Uzamknutie trvá až do času, kým sa relácia nedostane do stavu UPDATE, alebo sa nezruší spojenie.

Po otvorení schránky POP3 server priradí každému odkazu číslo a zaznamená si veľkosť každého odkazu. Čísla sú priradované vo vzostupnom poradí od 1 do *n*, kde *n* je počet odkazov v schránke.

## Príkazy poskytované serverom v transakčnom stave

### STAT

Parametre:   žiadne

Odpoveď:    +OK nn mm

POP3 server zašle pozitívnu odpoveď s dvoma číselnými hodnotami. *nn* je počet odkazov v poštovej schránke a *mm* je veľkosť schránky v osembitových slabikách (ďalej len oktet).

Príklad:     C: STAT  
              S: +OK 2 320

### LIST [msg]

Parametre:   voliteľné číslo odkazu, ktoré ak je uvedené, nesmie ukazovať na odkaz označený ako zmazaný

Odpoveď:    +OK scan listing follows  
              -ERR no such message

Ak bol zadaný parameter a POP3 server pošle pozitívnu odpoveď, server pošle odpoveď s riadkom obsahujúcim informácie pre tento odkaz v rovnakom tvare ako pri príkaze STAT (číslo odkazu a jeho veľkosť v oktoch).

Ak nebol uvedený argument v príkaze LIST, server pošle viacriadkovú odpoveď. Po identifikátore pozitívnej odpovedi (+OK) server pošle pre každý odkaz v schránke riadok obsahujúci informácie pre tento odkaz. Ak nie sú v schránke žiadne odkazy, server pošle iba pozitívnu odpoveď.

Príklad: C: LIST  
S: +OK 2 messages (320 octets)  
S: 1 120  
S: 2 200  
S: .  
  
C: LIST 2  
S: +OK 2 200  
C: LIST 3  
S: -ERR no such message

### RETR msg

Parametre: číslo odkazu, ktoré sa nesmie odvolávať na odkaz označený ako zmazaný

Odpoveď: +OK message follows  
-ERR no such message

POP3 server pošle viacriadkovú odpoveď, ktorá predstavuje vlastný odkaz.

Príklad: C: RETR 1  
S: +OK 120 octets  
S: <tu sa nachádza celý odkaz>  
S: .

### DELE msg

Parametre: číslo odkazu, ktoré sa nesmie odvolávať na odkaz označený ako zmazaný

Odpoveď: +OK message deleted  
-ERR no such message

POP3 server označí odkaz s číslom msg ako zmazaný. Všetky ďalšie referencie na toto číslo odkazu v príkazoch POP3 vygenerujú chybu. POP3 server nevymaže odkaz ihneď po prijatí príkazu DELE, ale až sa relácia dostane do stavu UPDATE.

Príklad: C: DELE 1  
S: +OK message 1 deleted  
C: DELE 1  
S: -ERR message 1 already deleted

### NOOP msg

Parametre: žiadne

Odpoveď: +OK

POP3 server nevykoná žiadnu akciu. Príkaz slúži na *reset autologout timer*.

Príklad: C: NOOP  
S: +OK

### RSET

Parametre: žiadne

Odpoveď: +OK

Ak niektoré odkazy boli označené ako zmazané, POP3 server ich odznačí a vráti pozitívnu odpoveď.

Príklad: C: RSET  
S: +OK mailbox has 2 messages (320 octets)

## Stav UPDATE

Keď klient pošle v transakčnom stave príkaz QUIT, POP3 server vstúpi do stavu UPDATE. Ak klient zadá príkaz QUIT v stave AUTHORIZATION, server reláciu ukončí, ale nevstúpi do stavu UPDATE.

Ak relácia končí z nejakého iného dôvodu ako je zaslanie príkazu QUIT klientom, POP3 server nevstúpi do stavu UPDATE a nevymaže žiadny odkaz zo schránky. Na vstup do stavu UPDATE teda slúži príkaz QUIT.

### QUIT

Parametre: žiadne

Odpoveď: +OK  
-ERR some deleted messages not removed

Poznámka: Príkaz je možné zadať v ktoromkoľvek stave.

POP3 server vymaže všetky odkazy zo schránky, ktoré boli označené ako zmazané a pošle odpoveď s výsledkom tejto operácie. Následne server odomkne schránku (viď. stav AUTHORIZATION) a uzavrie TCP spojenie.

Príklad:

```
C: QUIT
S: +OK POP3 server signing off (mailbox empty)
...
C: QUIT
S: +OK POP3 server signing off (2 messages left)
```

## Doplnkové POP3 príkazy

Všetky doteraz uvedené príkazy okrem APOP musia byť súčasťou každého POP3 servera. Nasledovné príkazy nemusia byť implementované každým POP3 serverom. Tieto doplnkové príkazy (*optional commands*) poskytujú klientovi väčšie možnosti pri spracovaní odkazov. Príkazy je možné použiť iba v transakčnom stave.

### TOP msg n

Parametre: číslo odkazu, ktoré sa nesmie odvolávať na odkaz označený ako zmazaný  
nezáporné číslo n určujúce počet riadkov

Odpoveď: +OK top of message follows  
-ERR no such message

Ak server odpovie pozitívne, pošle viacriadkovú odpoveď. POP3 server pošle hlavičku odkazu s číslom msg, za ktorými nasleduje prázdny riadok oddeľujúci hlavičku od tela odkazu a následne pošle n riadkov z tela odkazu. Server teda pošle prvých n riadkov z tela odkazu. Ak počet riadkov uvedený v príkaze TOP je väčší ako telo odkazu, server pošle celý odkaz.

Príklad:

```
C: TOP 1 10
S: +OK
S: <POP3 server pošle hlavičku správy,
   prázdny riadok, a prvých 10 riadkov
   z tela odkazu>
S: .
```

### UIDL [msg]

Parametre: voliteľné číslo odkazu, ktoré ak je uvedené, nesmie sa odvolávať na odkaz označený ako zmazaný

Odpoveď: +OK unique-id listing follows  
-ERR no such message

Ak bol zadán parameter a POP3 server pošle pozitívnu odpoveď, riadok s odpoveďou obsahuje informácie pre tento odkaz. Informácie sú v tvare číslo odkazu a unikátne ID odkazu.

Ak nebol uvedený argument v príkaze UIDL, server pošle viacriadkovú odpoveď. Po identifikátore pozitívnej odpovedi (+OK) server pošle pre každý odkaz v schránke riadok obsahujúci informácie pre tento odkaz.

Ak nie sú v schránke žiadne odkazy, server pošle iba pozitívnu odpoveď. Pre odkazy označené ako zmazané sa neposiela unikátne ID odkazu.

Unikátne ID odkazu (UID) je reťazec pozostávajúci z jedného až 70 znakov v rozsahu 0x21 až 0x7E, ktorý server priradí každému odkazu a ktorý jednoznačne identifikuje odkaz v schránke. Toto UID je konštantné a pretrváva medzi reláciami (po ukončení POP3 relácie a opätovnom nadviazaní TCP spojenia na POP3 server odkazy v rovnakej schránke budú mať rovnaké UID, ale číslo odkazu už nemusí byť totožné). Teda pomocou príkazu UIDL klient dokáže vykonávať operácie nad odkazmi, ak došlo k prerušeniu spojenia resp. relácia bola z nejakého dôvodu ukončená (klient pomocou uchovaných UID odkazov a pomocou príkazu UIDL identifikuje čísla odkazov, nad ktorými bude ďalej vykonávať operácie).

Príklad: C: UIDL  
S: +OK  
S: 1 whqtsqO00WBw418f9t5JxYwZ  
S: 2 QhdPYR:00WBw1Ph7x7  
S: .



Mnoho POP3 klientov má možnosť ponechať prečítanú poštu na serveri. Z hľadiska systémovej politiky by klienti mali používať príkaz DELE (ináč veľkosť schránky ustavične narastá). Server však môže byť nakonfigurovaný tak, že po stiahnutí odkazov klientom (RETR) automaticky označí odkaz za vymazaný. Keď klient štandardne ukončí reláciu príkazom QUIT, odstráni sa vymazané odkazy zo schránky. Servery realizujúce politiku *download-and-delete* často limitujú alebo znemožňujú použitie príkazu TOP, nakoľko ten sa dá použiť ako alternatívny mechanizmus k načítaniu celého odkazu.

## Bezpečnostné riziká protokolu

- Servery, ktoré odpovedajú správou –ERR na príkaz USER, môžu poslúžiť pri útoku, pretože poskytujú informáciu: „ktoré mená v systéme sú platné“.
- Heslo uvedené v príkaze PASS sa prenáša sieťou v otvorenej forme.
- Príkazy RETR a TOP prenášajú odkazy tiež v otvorenej forme.



# Internet Message Access Protocol - Version 4

## Úvod

Internet Message Access Protocol, Version 4 (IMAP4) dovoľuje klientovi prístup k odkazom poštovej schránky a manipulovať s nimi priamo na serveri. IMAP4 poskytuje príkazy na manipuláciu schránok s odkazmi elektronickej pošty na vzdialenom serveri (nazývanými *mailboxes*) spôsobom, ktorý je funkčne ekvivalentný s prácou na schránkach uložených lokálne. IMAP4 taktiež poskytuje pre *offline* klientov možnosť opätovnej synchronizácie so serverom.

IMAP4 zahŕňa operácie vytvárania, mazania a premenovávania schránok; kontrolu na príchod nových odkazov do schránky; nastavovanie a rušenie príznakov; RFC-822 a MIME analýzu; vyhľadávanie a selektívny výber atribútov odkazu, tela odkazu a jeho častí. V IMAP4 sú odkazy sprístupňované pomocou sekvenčných čísel alebo unikátnych identifikátorov (rozdiel oproti POP3).

## Základná charakteristika

IMAP4 používa TCP spojenie. Server počúva na porte 143. Relácia v IMAP4 pozostáva z vytvorenia klient/server spojenia, úvodného pozdravu a klient/server komunikácie. Komunikácia sa skladá z klientských príkazov, údajov zasielaných serverom a konečných odpovedí servera.

Všetky interakcie medzi klientom a serverom prebiehajú vo forme zasielania riadkov, t.j. reťazcov ukončených CRLF. Protokolový prijímač klienta alebo servera číta riadok, alebo číta postupnosť osembitových slabík s vopred známou dĺžkou (uvedená na začiatku riadku v tvare {počet znakov nasledujúcej postupnosti}).

Príkazy klienta začínajú jedinečnými identifikátormi (typicky krátky alfanumerický reťazec, napr. A001, A002, atď.) nazývanými značky (angl. *tag*). Server nerozlišuje v identifikátoroch príkazov malé a veľké písmená. Riadky zasielané klientom nemusia predstavovať úplné príkazy. Napr. príkaz s argumentom typu literál najprv zašle dĺžku posielaného reťazca a čaká, kým je server pripravený prijať vlastný reťazec. Iný prípad je, keď server pošle požiadavku na pokračovanie v príkaze (je pripravený prijať ďalšie údaje od klienta resp. celý zvyšok príkazu). V takýchto prípadoch sú výzvy servera uvádzané s prefixom "+".

Server zasiela odpovede dvoch typov. Buď pošle konečnú odpoveď na príkaz (začína rovnakou značkou, akú klient poslal spolu s príkazom ⇒ klient môže poslať naraz viacero požiadaviek s príkazmi na vykonanie – každý príkaz má inú značku) alebo posieľa neukončené odpovede (napr. viacriadková odpoveď), ktoré začínajú prefixom "\*".

Server môže zaslať údaje klientovi buď ako reakciu na príkaz, alebo aj bez vyžiadania (klient teda musí byť vždy pripravený prijať údaje od servera). Syntakticky nie sú tieto odpovede nijako rozlíšené.

Každá konečná odpoveď indikuje úspech alebo zlyhanie operácie. Server odosiela 3 možné typy konečných odpovedí:

- OK – indikuje úspech
- NO – indikuje zlyhanie
- BAD – indikuje chybu v protokole ako napr. neznámy príkaz, zlý počet parametrov, ...

## Stavový diagram

IMAP4 server sa môže nachádzať v jednom zo štyroch stavov. Väčšina príkazov od klienta je platná iba v určitom stave. Ak sa klient pokúsi o vykonanie príkazu v nesprávnom stave (napr. chce čítať odkaz, no ešte sa neautentifikoval), server zašle konečnú odpoveď BAD alebo NO (závisí od implementácie).

### 1. Stav NON-AUTHENTICATED

V neautentifikovanom stave musí používateľ (klient) zadať identifikačné údaje (napr. meno a heslo), aby mu bola sprístupnená väčšina príkazov. Do tohto stavu sa server dostane ihneď po zostavení TCP spojenia, pokiaľ spojenie nebolo pre-autorizované (autentifikácia prebieha ešte počas spojenia).

## 2. Stav AUTHENTICATED

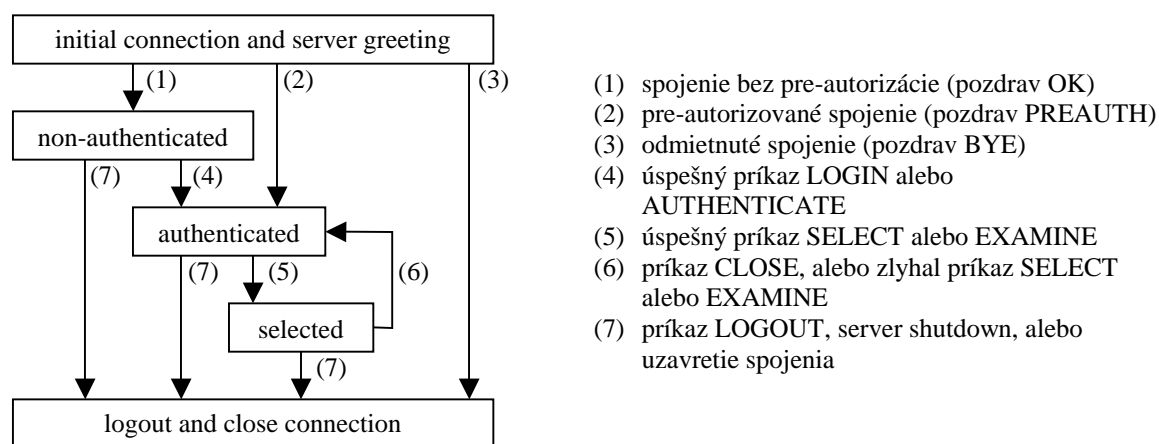
V autentifikovanom stave je používateľská identifikácia overená a klient si musí vybrať schránku, ktorú chce sprístupniť. Po sprístupnení môže používať príkazy, ktoré sa aplikujú na odkazy v schránke. V tomto stave sa server nachádza buď po zostavení pre-autorizovaného spojenia, alebo po overení identity klienta, alebo po chybe pri výbere schránky.

## 3. Stav SELECTED

V stave vybraný sa server nachádza po tom, čo si klient úspešne vybral schránku, s ktorou bude pracovať.

## 4. Stav LOGOUT

V stave odhlasovania sa relácia ukončí a server uzavrie spojenie. Tento stav je možné dosiahnuť na požiadanie klienta (príkaz LOGOUT) alebo neočakávane (napr. chyba pri spojení).



Obr. Stavový diagram IMAP4 servera

## Formáty údajov

Údaje prenášané v IMAP4 sú v textovom tvare a môžu byť v jednej z nasledovných foriem:

- Atóm – jeden alebo viacero nešpeciálnych znakov
- Číslo – pozostáva z jednej alebo viacerých číslic
- Reťazec – buď literál (postupnosť osembitových slabík, ktorým predchádza dĺžka uvedená v zložených zátvorkách) alebo reťazec v úvodzovkách
- Zoznam – postupnosť položiek ohraničená zátvorkami (klasický LIST z Lispu)
- NIL – špeciálny atóm, ktorý reprezentuje nejestvovanie údajovej položky

## Názvy schránok

Názvy schránok sú závislé od implementácie. Schránka s označením INBOX reprezentuje primárnu schránku používateľa na danom serveri. Názvy schránok môžu byť hierarchické, tzn. že možno členiť štruktúru schránok podobne ako adresáre v súborovom systéme. Na oddeľovanie jednotlivých úrovní sa používa špeciálny znak (oddeľovač, napr. '/' vo väčšine UNIX systémov), ktorý sa dá zistiť pomocou príkazu LIST.

## Príkazy klienta

IMAP4 príkazy opísané v nasledujúcich odsekoch sú členené podľa stavu, v ktorom je príkaz povolený. Príkazy, ktoré je možné použiť vo viacerých stavoch, sú uvedené v minimálnom povolenom stave (napr. príkazy, ktoré sú platné aj pre stav AUTHENTICATED aj pre stav SELECTED, sú uvedené v zozname príkazov pre stav AUTHENTICATED).

## Príkazy klienta v ľubovoľnom stave

**CAPABILITY** – vyžiada zoznam možností a rozšírení servera, ktoré poskytuje.

Príklad: C: a001 CAPABILITY  
S: \* CAPABILITY IMAP4 IMAP4rev1 AUTH=KERBEROS\_V4 XPIG-LATIN  
S: a001 OK CAPABILITY completed

**NOOP** – nevykoná žiadnu operáciu.

Príklad: C: a002 NOOP  
S: a002 OK NOOP completed

**LOGOUT** – ukončenie relácie.

Príklad: C: B023 LOGOUT  
S: \* BYE IMAP4 server logging out  
S: B023 OK LOGOUT completed  
(Server and client then close the connection)

## Príkazy klienta v stave NON-AUTHENTICATED

**AUTHENTICATE** – príkaz indikuje serveru mechanizmus autentifikácie, ktorý sa použije na overovanie identity klienta (môže byť napr. aj LOGIN autentifikácia, ktorá bude prebiehať v kódovaní BASE64 – prenos však nie je šifrovaný).

Príklad: S: \* OK KerberosV4 IMAP4 Server  
C: A001 AUTHENTICATE KERBEROS\_V4  
S: + AmFYig==  
C: BAcaQU5EUkVXLkNNVS5FRFUAOCasho84kLN3/IJmrMG+25a4DT  
+nZImJjnTNHJUtxAA+o0KPKfHEcAFs9a3CL5Oebe/ydHJUWYFd  
WwuQlMWiy6IesKvjL5rL9WjXUb9MwT9bpObYLGOKilQh  
S: + or//EoAADZI=  
C: DiAF5A4gA+oOIALuBkAAmw==  
S: A001 OK Kerberos V4 authentication successful

**LOGIN** – príkaz identifikuje používateľa serveru, pričom heslo sa prenáša v nekryptovanej forme.

Príklad: C: a001 LOGIN smith sesame  
S: a001 OK LOGIN completed

## Príkazy klienta v stave AUTHENTICATED

**SELECT** – príkaz zvolí schránku, ktorej odkazy sa sprístupnia klientovi. Ak má klient povolené modifikovať schránku, server by mal poslať konečnú odpoveď s prefixom [READ-WRITE] (ináč musí poslať prefix [READ-ONLY]). Pred odoslaním odpovede OK, server pošle klientovi nasledovné neoznačené odpovede:

- FLAGS – príznaky definované v schránke
- <n> EXISTS – počet odkazov v schránke
- <n> RECENT – počet nových odkazov v schránke, ktoré boli pridané od posledného výberu schránky
- OK [UIDVALIDITY <n>] – hodnota unikátneho identifikátora schránky

Príklad: C: A142 SELECT INBOX  
 S: \* 172 EXISTS  
 S: \* 1 RECENT  
 S: \* OK [UNSEEN 12] Message 12 is first unseen  
 S: \* OK [UIDVALIDITY 3857529045] UIDs valid  
 S: \* FLAGS (\Answered \Flagged \Deleted \Seen \Draft)  
 S: \* OK [PERMANENTFLAGS (\Deleted \Seen \\*)] Limited  
 S: A142 OK [READ-WRITE] SELECT completed

**EXAMINE** – príkaz je identický s príkazom SELECT a vracia rovnaký výstup, no schránka je vybraná ako READ-ONLY (iba na čítanie). Žiadne zmeny v schránke nie sú povolené.

**CREATE** – príkaz vytvorí schránku s názvom zadaným ako parameter príkazu. Príkaz vygeneruje chybu, ak sa klient pokúša vytvoriť INBOX alebo schránku s existujúcim menom. Ak má názov schránky na konci uvedený znak oddeľovača, je to deklarácia, že klient môže (v budúcnosti) vytvárať schránky pod týmto menom v hierarchii.

Príklad: C: A003 CREATE posta/  
 S: A003 OK CREATE completed  
 C: A004 CREATE posta/odoslana  
 S: A004 OK CREATE completed  
 (v tomto príklade predpokladáme, že / je znak oddeľovača  
 vrátený príkazom LIST)

**DELETE** – príkaz natrvalo vymaže schránku s názvom zadaným ako parameter príkazu. Novovytvorená schránka s rovnakým menom sa líši v hodnote UID (unikátneho identifikátora).

Príklad: C: A683 DELETE posta  
 S: A683 OK DELETE completed

**RENAME** – príkaz slúži na premenovanie schránky.

Príklad: C: Z4S9 RENAME posta mailbox  
 S: Z4S9 OK RENAME completed

**SUBSCRIBE/UNSUBSCRIBE** – príkaz pridá/odoberie špecifikovanú schránku do/zo skupiny „aktívnych“ schránok, ktoré vracia príkaz LSUB. Využitie v news-och.

Príklad: C: A002 SUBSCRIBE #news.comp.mail.mime  
 S: A002 OK SUBSCRIBE completed  
 ...  
 C: A009 UNSUBSCRIBE #news.comp.mail.mime  
 S: A009 OK UNSUBSCRIBE completed

**LIST** – príkaz vráti množinu názvov schránok prístupných používateľovi, ktoré vyhovujú zadaným kritériám (parametre príkazu). Server vráti niekoľko neoznačených odpovedí, obsahujúcich množinu atribútov, oddeľovač používaný v hierarchii schránok a názov schránky. Prvým parametrom príkazu je názov referencie (cesta v hierarchii schránok, kde sa budú hľadať schránky) a druhým parametrom je názov schránky s možnými zástupnými znakmi (žolík \* znamená 0 alebo viac znakov na tejto pozícii. Žolík % je podobný \*, ale nezahŕňa znak oddeľovača ⇒ prehľadáva iba na rovnakej úrovni v hierarchii).

Príklad: C: A004 LIST "~/Mail/" "%"  
 S: \* LIST (\Noselect) "/" ~/Mail/family\_subfolder  
 S: \* LIST () "/" ~/Mail/sent-mail  
 S: \* LIST (\Noinferiors) "/" ~/Mail/postponed-messages  
 S: A004 OK LIST completed

**LSUB** – príkaz vráti podmnožinu názvov (ktorá vyhovuje zdaným kritériám) z množiny názvov schránok, ktoré používateľ označil ako „aktívne“ (pomocou príkazu SUBSCRIBE). Parametre sú totožné ako pri príkaze LIST. Výstup je tiež podobný (odlišnosť je v znaku oddeľovača v hierarchii).

Príklad: C: A005 LSUB "#news." "comp.mail.\*"  
 S: \* LSUB () "." #news.comp.mail.mime  
 S: \* LSUB () "." #news.comp.mail.misc  
 S: A005 OK LSUB completed

**APPEND** – príkaz pridá do schránky nový odkaz, ktorý tvorí argument príkazu (argument je typu literál). Nový odkaz musí spĺňať formát definovaný v štandarde RFC-822. Príkaz môže byť zadáný s ďalšími dvoma voliteľnými parametrami, a to zoznam príznakov a dátum/čas zaznamenania odkazu.

Príklad: C: A011 APPEND saved-messages (\Seen \Answered) {310}  
C: Date: Mon, 7 Oct 2001 21:45:25 -0800 (PST)  
C: From: Fred Fooobar <fooobar@blurdybloop.com>  
C: To: mooch@owatagu.siam.edu  
C: Subject: Afternoon meeting  
C: Message-ID: <Q27397-0100000@blurdybloop.com>  
C: MIME-Version: 1.0  
C: Content-Type: TEXT/PLAIN; Charset=US-ASCII  
C:  
C: Hello Joe, do you think we can meet at 3:30 tomorrow?  
C:  
S: A011 OK APPEND completed

## Príkazy klienta v stave SELECTED

**CHECK** – príkaz vytvorí kontrolný bod (*checkpoint*) vybranej schránky. Kontrolný bod slúži na organizovanie schránky, ktoré je implementačne závislé a ktoré sa normálne vykonáva ako súčasť každého príkazu.

**CLOSE** – príkaz uzavrie otvorenú schránku, tzn. že vymaže všetky odkazy, ktoré sú označené ako zmazané (príznak \Deleted) a vráti server do stavu AUTHENTICATED. Príkaz nevracia žiadnu neoznačenú odpoveď (rozdiel oproti EXPUNGE, viď. ďalej). Príkaz je potrebné zadávať pred príkazom SELECT, EXAMINE alebo LOGOUT (v tomto prípade sa schránka implicitne uzavrie, no odkazy označené ako zmazané sa neodstránia).

Príklad: C: A431 CLOSE  
S: A431 OK CLOSE completed

**EXPUNGE** – príkaz odstráni zo schránky odkazy označené ako zmazané. Pred zaslaním konečnej odpovede server pošle neoznačenú odpoveď pre každý odkaz.

Príklad: C: A202 EXPUNGE  
S: \* 3 EXPUNGE  
S: \* 3 EXPUNGE  
S: \* 5 EXPUNGE  
S: \* 8 EXPUNGE  
S: A202 OK EXPUNGE completed

V tomto príklade odkazy 3, 4, 7 a 11 mali príznak \Deleted. Poradové číslo odkazu pre každý nasledujúci odkaz sa po úspešnom odstránení dekrementuje o 1, čo sa ihneď prejaví v ďalších odpovediach servera. Ak server odstraňuje odkazy odzadu (od vyšších čísel po nižšie), poradové čísla práve vymazávaných odkazov sa nemenia.

**SEARCH** – príkaz hľadá odkazy v schránke, ktoré vyhovujú zadaným kritériám. Vyhľadávacie kritériá môžu pozostávať z niekoľkých kľúčov, ktoré sú spojené konjunkciou. Kľúče môžu byť typu zoznam a môžu obsahovať aj logické spojky OR, NOT a pod. Server vracia poradové čísla odkazov, ktoré vyhovujú vyhľadávacím kritériám.

Príklad: C: F282 SEARCH SMALLER 1000 SINCE 1-Oct-2001 NOT FROM "Smith"  
S: \* SEARCH 2 84 99  
S: F202 OK SEARCH completed

**FETCH** – príkaz vyberie zadané údaje, ktoré sú priradené odkazom v schránke. Vybrané údaje môžu byť typu buď jednoduchý atóm alebo zoznam.

Príklad: C: A343 FETCH 4:6 RFC822  
S: \* 4 FETCH ...  
S: \* 5 FETCH ...  
S: \* 6 FETCH (RFC822 {457})  
S: Return-Path: <root>  
S: Received: (from root@localhost)  
S: by localhost.localdomain (8.9.3/8.9.3) id UAA00434  
S: for dumb; Mon, 15 Oct 2001 20:35:39 GMT  
S: Date: Mon, 15 Oct 2001 20:35:39 GMT  
S: From: root <root@localhost.localdomain>  
S: Message-Id: <200110152035.UAA00434@localhost.localdomain>  
S: To: dumb@localhost.localdomain  
S: Subject: Slovenske nazvy v menach suborov  
S:  
S: 6.29 Dlouhá jména na partition s Windows-95/NT  
S: Ma-li byt v Linuxu správnì vidìt diakritika v dlouhych  
jménech souborù na  
S: partition s Windows-95, je tøeba v souboru /etc/fstab uvést  
S: ...  
S: FLAGS (\Seen))  
S: A343 OK FETCH completed

Príznamy odkazu vrátené v odpovedi FLAGS môžu byť:

- \Seen – odkaz bol prečítaný
- \Answered – na odkaz sa odpovedalo (*reply*)
- \Flagged – odkaz má príznak „urgentný“ alebo „špeciálne upozornenie“
- \Deleted – odkaz je označený ako zmazaný (po zadaní EXPUNGE sa odstráni)
- \Draft – písanie odkazu nie je ukončené (odkaz je označený ako koncept)
- \Recent – nový odkaz, ktorý bol zapísaný do schránky od posledného prístupu k schránke.

**PARTIAL** – funkcia tohto príkazu je ekvivalentná s funkciou príkazu FETCH (dokonca výstup majú rovnaký), ale rozdiel je v ďalších dvoch pridaných parametroch – *offset* a počet oktetov, ktoré sa vrátia (teda podobne ako je dvojica príkazov RETR a TOP v protokole POP3).

Príklad: C: A0026 PARTIAL 6 RFC822.HEADER 33 19  
S: \* 6 FETCH (RFC822.HEADER "from root@localhost")  
S: A0026 OK PARTIAL completed

**STORE** – príkaz zmení príznaky pre daný odkaz (skupinu odkazov) v schránke. Server vráti aktuálne hodnoty príznakov zaktualizovaných odkazov. Ak sa v príkaze použije prípona “.SILENT”, server nevracia žiadnu odpoveď (iba konečnú). Parameter FLAGS nahradí súčasné príznaky novými, parameter +FLAGS pridá nové príznaky k súčasným a parameter -FLAGS odoberie príznaky.

Príklad: C: A0027 STORE 2:4 +FLAGS (\Deleted)  
S: \* 2 FETCH FLAGS (\Deleted \Answered \Seen)  
S: \* 3 FETCH FLAGS (\Deleted \Seen)  
S: \* 4 FETCH FLAGS (\Deleted)  
S: A0027 OK STORE completed

**COPY** – príkaz skopíruje špecifikovaný odkaz(y) do zadanej schránky. Príznamy a dátum prijatia odkazu by sa mali zachovať v kópii.

Príklad: C: A0032 COPY 4:6 inbox  
S: \* 9 EXISTS  
S: \* 0 RECENT  
S: A0032 OK COPY completed

**UID** – príkaz vykoná príkaz uvedený ako parameter (aj so všetkými jeho argumentmi) nad množinou odkazov so špecifikovanými UID (unikátne číslo odkazu v schránke). Unikátne identifikátory odkazov sa zachovávajú medzi jednotlivými reláciami  $\Rightarrow$  klient sa môže znovu zosynchronizovať s predchádzajúcim stavom relácie (napr. zrušenie spojenia alebo *offline* prístup klienta). Ak by sa z nejakého dôvodu zmenil unikátny identifikátor medzi dvoma reláciami, musí sa zmeniť aj príznak UIDVALIDITY pre túto schránku (viď. príkaz SELECT). Povolené príkazy uvedené ako parametre príkazu UID sú: COPY, FETCH, STORE a SEARCH.

Príklad: C: A0037 FETCH 6:7 UID  
S: \* 6 FETCH (UID 29)  
S: \* 7 FETCH (UID 30)  
S: A0037 OK FETCH completed  
C: A0038 UID FETCH 29:30 FLAGS  
S: \* 6 FETCH (UID 29 FLAGS (\Seen \Answered))  
S: \* 7 FETCH (UID 30 FLAGS (\Seen))  
S: A0038 OK UID FETCH completed

## Experimentálne a rozšírené príkazy klienta

**X<atom>** – všetky príkazy s prefixom X označujú príkazy, ktoré nie sú súčasťou špecifikácie protokolu IMAP4. Ak napríklad klient potrebuje vykonať niektorý príkaz z rozšírení uvedených v odpovedi príkazu CAPABILITY, musí tento príkaz uviesť s predponou X.

Príklad: C: A441 CAPABILITY  
S: \* CAPABILITY IMAP4 XPIG-LATIN  
S: A441 OK CAPABILITY completed  
C: A442 XPIG-LATIN  
S: \* XPIG-LATIN ow-nay, eaking-spay ig-pay atin-lay  
S: A442 OK XPIG-LATIN ompleted-cay

## Bezpečnostné riziká protokolu

- Údaje vymieňané medzi serverom a klientom nie sú nijako zabezpečené voči tretej strane, pokiaľ sa ne-zvolila špeciálna ochrana pomocou príkazu AUTHENTICATE.
- Heslo uvedené v príkaze LOGIN sa prenáša sieťou v otvorenej forme.

## Revízia 1 protokolu IMAP4

Posledná verzia protokolu (IMAP4rev1) sa od popisu protokolu v tomto dokumente odlišuje iba veľmi málo. Bol napr. pridaný nový príkaz STATUS, ktorý zistí stavové informácie zadanej schránky bez toho, že by zmenil stav schránky alebo niektorý z jej odkazov (teda zachováva sa príznak \Recent). V tejto verzii bol ďalej zrušený príkaz PARTIAL a niektoré atribúty príkazu FETCH boli nahradené novými. Príkaz CAPABILITY už vracia aj poskytované autentifikačné mechanizmy. Schránkam je možné pridať popis (*description*). Ostatné zmeny sa týkajú najmä dokumentácie RFC-1730.

## Poštoví klienti a WEB mail

Prakticky všetky klientské programy, ktoré čítajú poшту uloženú na vzdialenom počítači, podporujú protokol POP3. Väčšina z nich podporuje aj IMAP4. Ak sa k Internetu pripájame pomocou modemu a poшту máme uloženú u svojho poskytovateľa Internetových služieb (ISP – Internet Service Provider), je výhodné aby server poskytoval príkazy na čiastočné načítanie poštových odkazov (napr. TOP v POP3, alebo PARTIAL v IMAP4), pretože potom je možné načítať zo servera iba hlavičky s niekoľko prvými riadkami odkazu. Ak je odkaz veľmi veľký (napr. väčší ako 1 MB), klient môže zmazať odkaz na serveri (príkaz DELE v POP3, alebo STORE v IMAP4), a teda preťaží linku pri čítaní pošty. Príkladom takéhoto správania sa je funkcia *Selective mail download* v poštovom klientovi *Pegasus mail*.

Väčšina poštových klientov používa *offline* režim pre prácu s POP3/IMAP4 serverom, tzn. jednotlivé činnosti (ako napr. spomínané načítanie hlavičiek odkazov, vlastné čítanie pošty, zmazanie odkazov na strane servera, a pod.) sa vykonávajú ako samostatné relácie (samostatné TCP spojenia). Keďže poradové čísla odkazov a aj samotný počet odkazov sa v jednotlivých reláciách môžu meniť, poštoví klienti najskôr musia identifikovať odkazy pomocou *Unique-ID*, ktoré sa zachovávajú medzi jednotlivými reláciami. Klient teda po spojení najskôr zistí poradové čísla odkazov pomocou UID z predchádzajúcej relácie a až potom pracuje s odkazmi v schránke pomocou týchto čísel. Toto má význam pri činnostiach, ktoré navzájom súvisia a ktoré sa vykonávajú v samostatných reláciách – napr. selektívne načítanie odkazov.

WEB mail je obdoba klasickej pošty, tzn. doručenie pošty prebieha na rovnakom princípe, iba prístup k pošte sa vykonáva cez WWW rozhranie. Väčšina súčasných WEB poštových serverov poskytuje aj prístup k pošte pomocou protokolu POP3 a/alebo IMAP4, a teda na čítanie odkazov z poštovej schránky je možné použiť ktorékoľvek poštového klienta podporujúceho rovnaký protokol ako ponúka server, na ktorom je pošta uložená.



## Použitá literatura

- [1] Myers, J., Rose, M., “Post Office Protocol – Version 3”, RFC 1939, Carnegie Mellon, Dover Beach Consulting, Inc., May 1996.
- [2] Crispin, M., “Internet Message Access Protocol – Version 4”, RFC 1730, University of Washington, December 1994.
- [3] Crispin, M., “Internet Message Access Protocol – Version 4rev1”, RFC 2060, University of Washington, December 1996.
- [4] Myers, J., “POP3 AUTHentication command”, RFC 1734, Carnegie Mellon, December 1994.
- [5] Crocker, D., “Standard for the Format of ARPA-Internet Text Messages”, RFC 822, University of Delaware, August 1982.