

FTP

(File Transfer Protocol)

Vypracoval : Václav Fišer (A02224)

E-mail : vencis@centrum.cz

Dne : 11.4.2006

KIV/PSI

Význam FTP

- **FTP** je protokol aplikační vrstvy z rodiny TCP/IP, je určen pro přenos souborů mezi počítači, na kterých mohou běžet velmi rozdílné operační systémy.
- Je to jeden z nejstarších protokolů, pracuje 8-bitově na principu klient-server na portech TCP/20 a TCP/21.
- Port 20 slouží k vlastnímu přenosu dat, port 21 slouží ke kontrole a jsou jím také přenášeny ftp příkazy.
- Je definován v RFC959. Protokol je interaktivní a umožňuje řízení přístupu (přihlašování login/heslo), specifikaci formátu přenášeného souboru (znakově - binárně), výpis vzdáleného adresáře atd.
- V současné době už není považován za bezpečný a z tohoto důvodu pro něj byla definována některá rozšíření (RFC 2228)

Trocha historie ...

- Na protokolu FTP je dobře patrný jeden význačný rys celého síťového modelu TCP/IP - totiž skutečnost, že nevznikl naráz, od zeleného stolu, ale postupným vývojem a zdokonalováním.
- První verze protokolu FTP pro netransparentní přenos souborů však pochází již z roku 1971, a od té doby prošla dlouhým vývojem, podporovaným širokou diskusí uživatelské veřejnosti
- Od začátku osmdesátých let, kdy i protokol FTP přešel na používání transportního protokolu TCP (místo původního transportního protokolu NCP sítě ARPANET) se ale již zásadněji neměnil.

Řídící spojení (Control Connection)

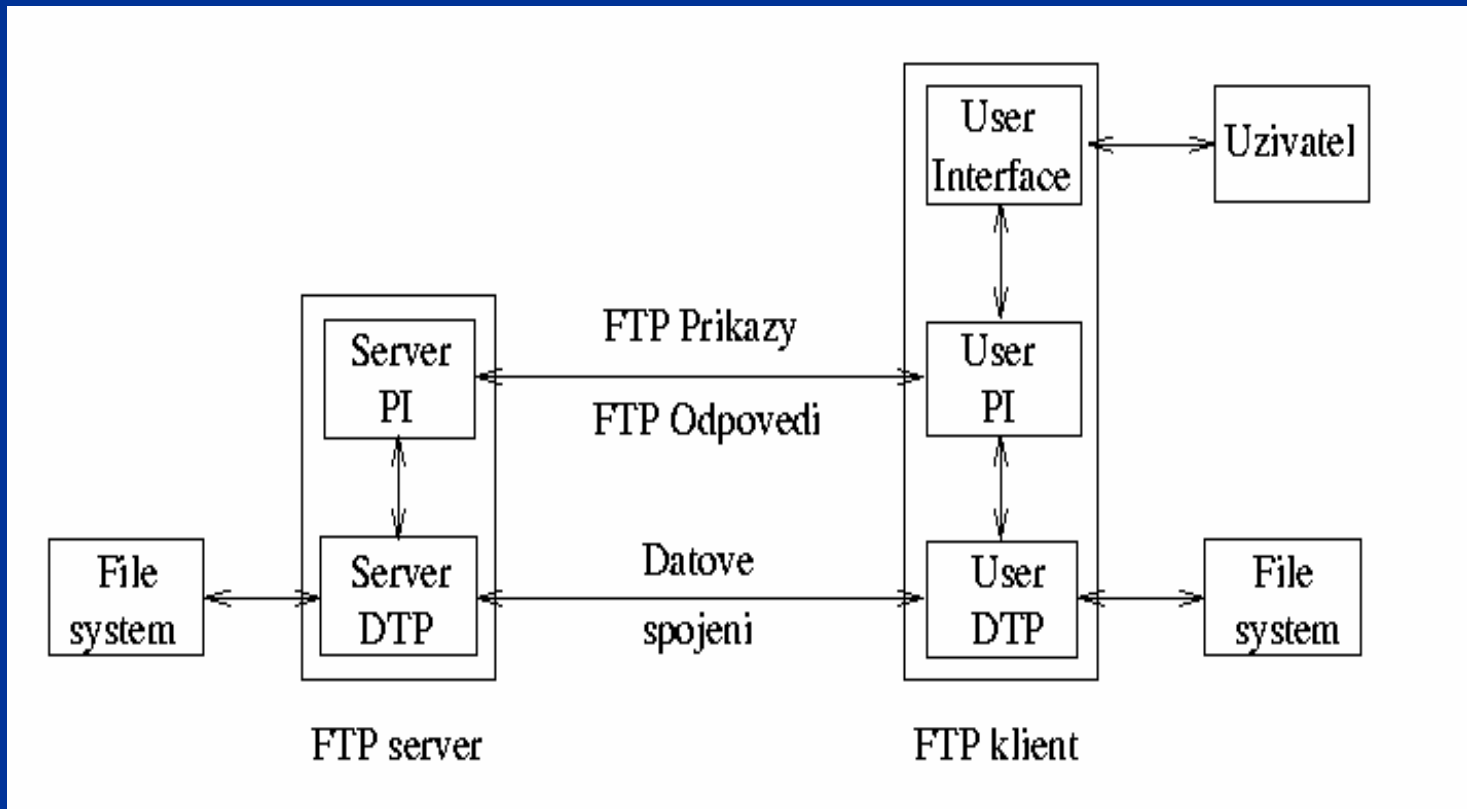
- řídicí spojení navazuje vždy klient
- klient musí znát číslo portu, na kterém server poslouchá - implicitně 21. Port ze kterého klient navázal řídicí spojení, je libovolný
- klient posílá po řídicím spojení FTP příkazy (FTP commands), což jsou ASCII řetězce ukončené telnetovským koncem řádky
- FTP příkazy mají tvar
 - PŘÍKAZ
 - PŘÍKAZ parametry
- kde PŘÍKAZ jsou tři nebo čtyři znaky identifikující konkrétní FTP příkaz

Formát odpovědi serveru

- server posílá po řídicím spojení FTP odpovědi (FTP replies), což jsou rovněž ASCII řetězce ukončené telnetovským koncem řádky, ale na rozdíl od příkazů může být jedna odpověď poslána ve více řádcích
- Např. :
 - pro jednoduché odpovědi
 - XYZ slovní popis odpovědi
 - pro víceřádkové odpovědi, kde XYZ je vždy kód odpovědi složený ze tří číslic
 - XYZ-první řádka
 - druhá řádka
 - ...
 - XYZ poslední řádka

Model použití FTP

- PI - Protocol Interpreter
- DTP - Data Transfer Process



Čísla odpovědí serveru (1)

- komunikace mezi klientem a serverem se skládá z FTP příkazů posílaných klientem a FTP odpovědí posílaných serverem
- číselné kódy odpovědí byly zavedeny proto, aby software na straně klienta mohl rychle rozhodnout, zda se poslaný příkaz povedl nebo ne
- první číslice kódu odpovědi určuje to, jestli se požadovaná akce povedla nebo ne
 - 1YZ - požadovaná akce byla úspěšně započata, čekej další odpověď
 - 2YZ - úspěch, můžeš poslat další příkaz
 - 3YZ - příkaz byl přijat, očekávám další (používáno u sekvencí příkazů, např. při přihlášení - klient pošle jedním příkazem uživatelské jméno, server vrátí odpověď s kódem 331 a očekává, že klient pošle dalším příkazem heslo)
 - 4YZ - akce se nezdařila, ale "je šance, že když to zkusíš za chvíli, tak to půjde"
 - 5YZ - příkaz nebyl přijat a v takové podobě ani nikdy nebude - jinými slovy: nezkoušej to znovu

Čísła odpovědí serveru (2)

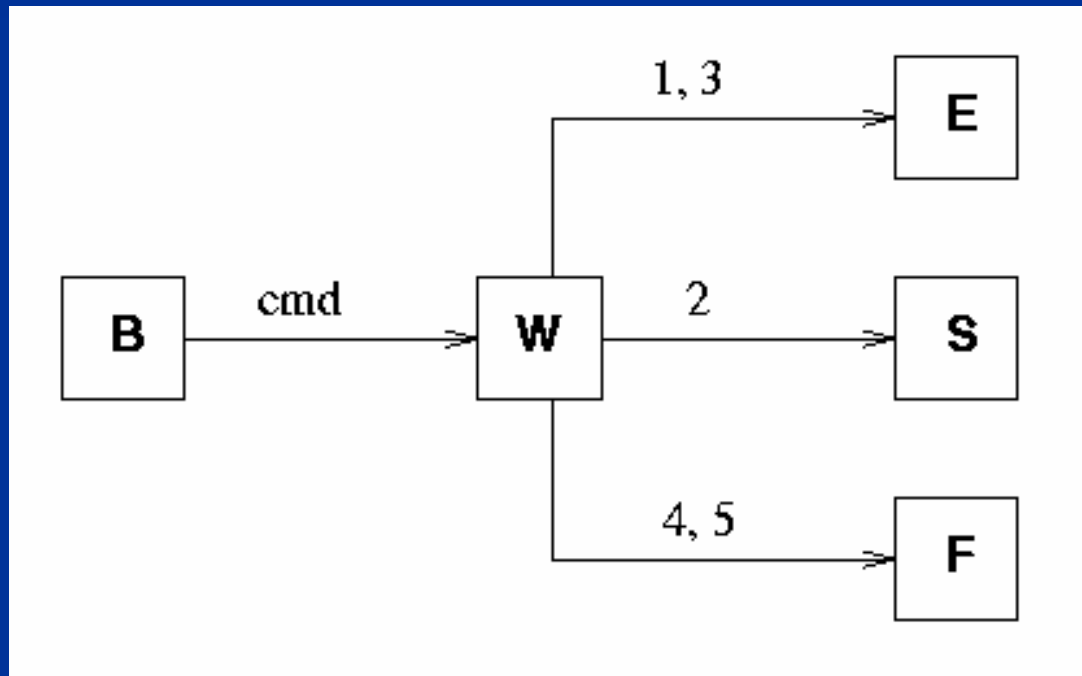
- druhá číslice určuje přibližně druh chyby nebo druh odpovědi
 - X0Z - syntaktická chyba
 - X1Z - odpověď na žádost o informace (např. nápověda)
 - X2Z - odpověď se týká řídicího nebo datového (viz níže) spojení
 - X3Z - odpověď se týká autentikace
 - X5Z - odpověď se týká souborového systému
- doprovodný text není u většiny odpovědí závazný a server může u takových odpovědí posílat cokoliv, nicméně existují odpovědi s přesně definovaným formátem

Zpracování příkazů

- Příkaz se během provádění nachází v různých stavech. Toto zpracování příkazu znázorním stavovým diagramem. Ve stavovém diagramu jsou tyto stavy:
 - B - Begin (počáteční stav)
 - W - Wait For Replay (čekání na odpověď)
 - S - Success (úspěšné provedení)
 - F - Failure (selhání)
 - E - Error (chyba)

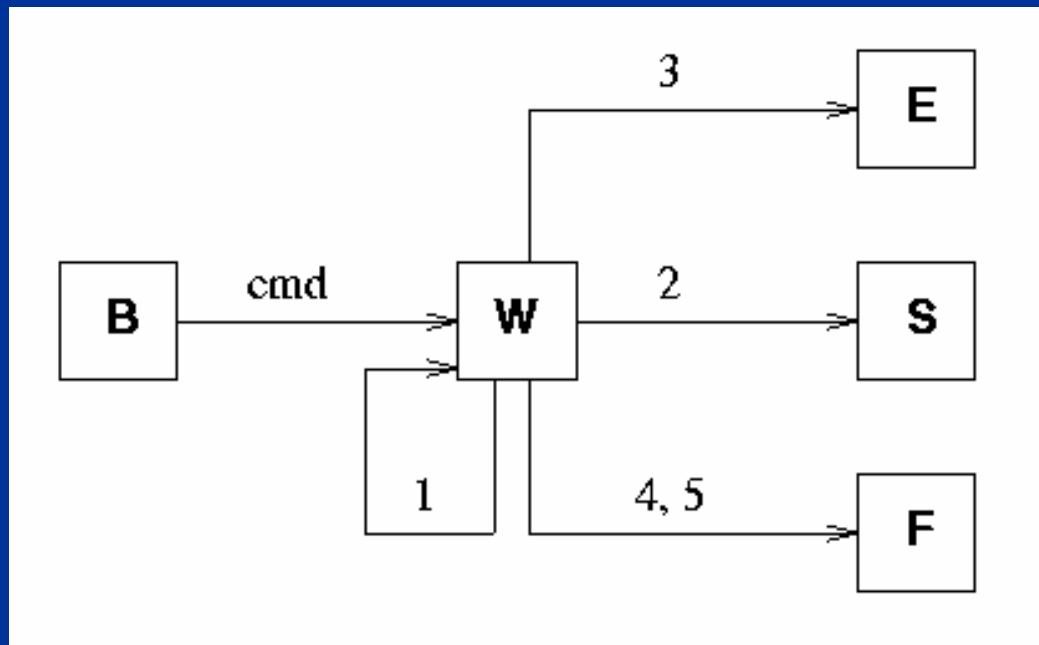
Stavový diagram pro nejjednodušší příkazy

- Pro příkazy: ABORT, DELE, CWD, CDUP, PASV, QUIT, PORT, SYST, RMD, MKD, PWD, TYPE.



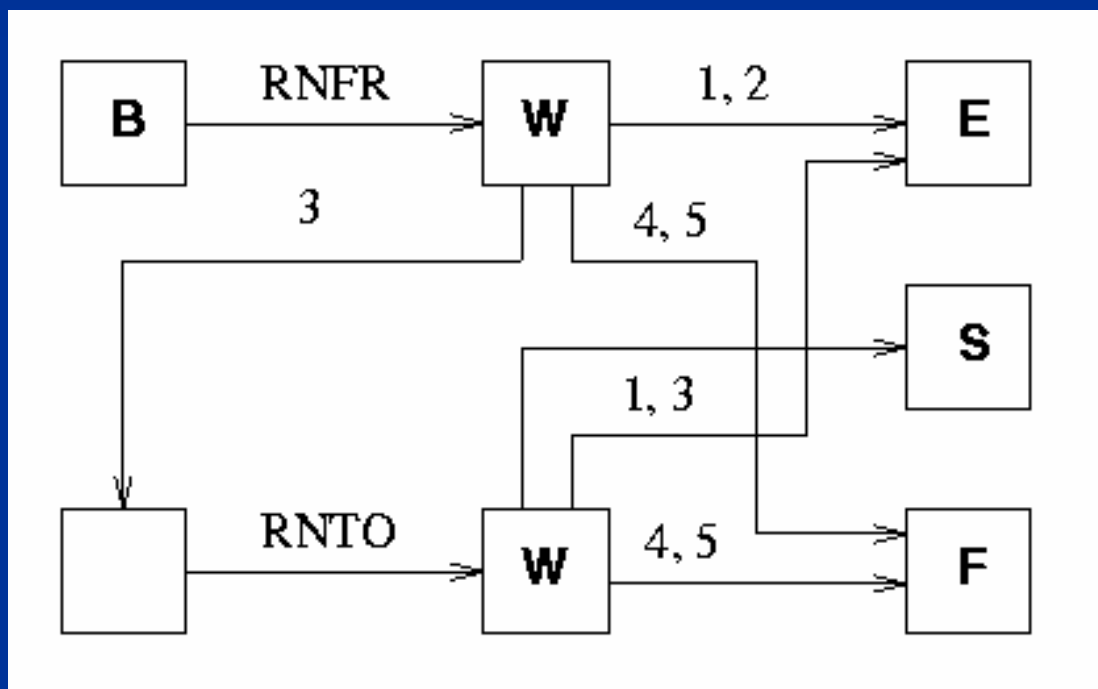
Stavový diagram pro druhou skupinu příkazů

- Druhá skupina příkazů je representována velmi podobným diagramem. Jsou to příkazy LIST, NLST, RETR, STOR



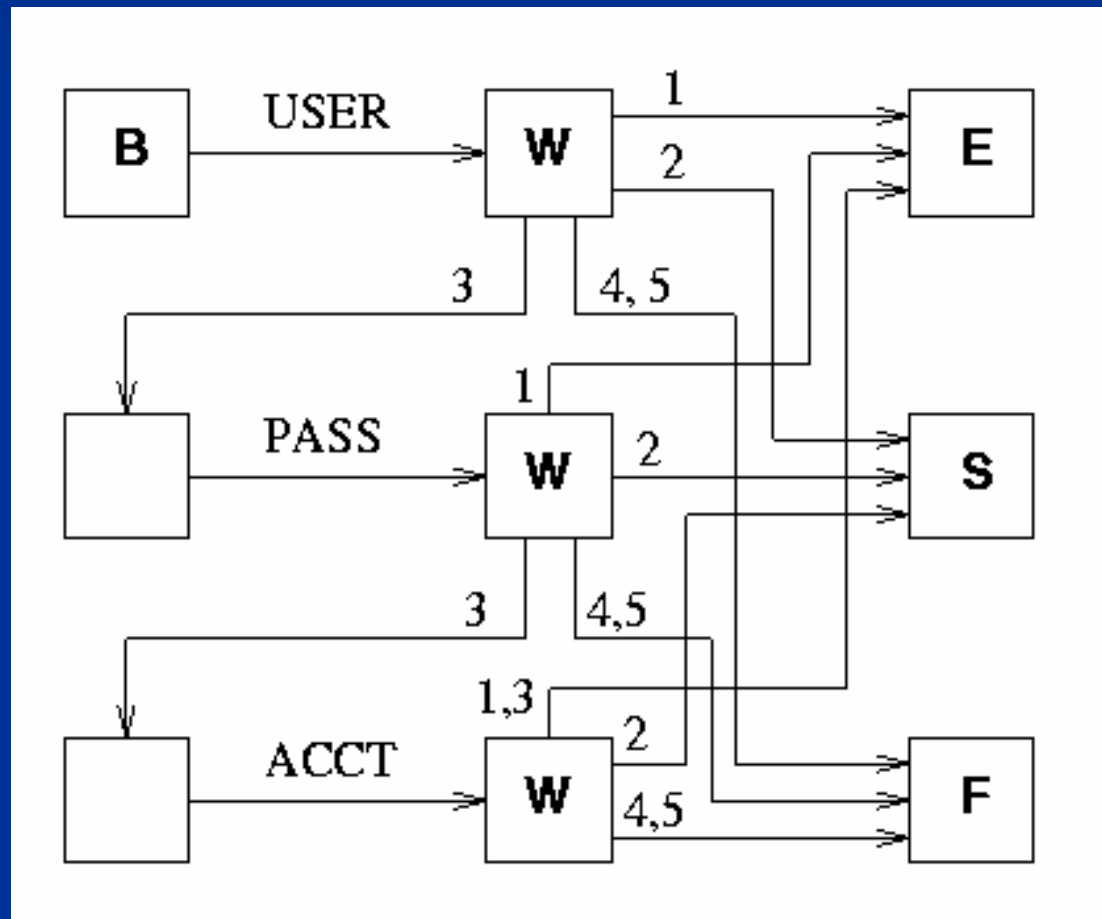
Stavový diagram při přejmenování souboru

- Diagram na obrázku zobrazuje příkazovou sekvenci při přejmenování souboru



Stavový diagram při přihlašování.

- Nejkomplikovanější digram na obrázku zachycuje přihlašování



Datové spojení (Data Connection)

- příkazy pro vytváření, rušení a přepínání se do adresářů, mazání a přejmenování souborů, zjišťování stavu serveru apod. si vystačí s řídicím spojením
- přenos souborů, od kterého dostal protokol FTP svůj název, a o který nám především jde, však vyžaduje ještě tzv. datové spojení
- Pozor! Výpis adresáře se také přenáší po datovém spojení (přenést výpis adresáře po řídicím spojení je možné za určitých okolností příkazem `STAT path`)
- FTP server může pracovat ve dvou módech, v aktivním (active mode, active transfer) a pasivním (passive mode, passive transfer)
- implicitně pracuje server v aktivním módu, do pasivního ho lze přepnout příkazem `PASV`

Aktivní mód

- Při aktivním způsobu klient pošle příkaz PORT, kde jako parametry jsou uvedeny IP adresa, vyšší a nižší byte čísla portu klienta, na kterém bude poslouchat. Server po obdržení tohoto příkazu vytvoří datové spojení právě podle parametrů příkazu PORT. Po vytvoření spojení klient pošle jeden z příkazů RETR, STORE nebo LIST a po datovém spojení se začnou posílat data.
 - PORT 147,32,112,3,10,97
 - 200 PORT command successful.
 - RETR /readme.txt
 - 150 Opening BINARY mode data connection for /readme.txt (107 bytes).
 - 226 Transfer complete.

Pasivní mód

- Při pasivním způsobu klient zašle serveru příkaz PASV. Server na tento příkaz pošle odpověď, ve které bude IP adresa a port, na kterém bude server poslouchat. Klient vytvoří datové spojení na tuto IP adresu a port a pošle příkaz (např. RETR, STORE nebo LIST).
 - PASV
 - 227 Entering Passive Mode (195,113,19,84,143,242)
 - RETR /readme.txt
 - 150 Opening BINARY mode data connection for /readme.txt (2181 bytes).
 - 226 Transfer complete.
- Příkazy PORT a PASV platí jen na následující datové spojení. Pokud je datové spojení ukončeno, musí se před otevřením nového datového spojení poslat nový příkaz PORT nebo PASV.

Důležité FTP příkazy (1)

- **USER** <username> přihlášení, uživatelské jméno
- **PASS** <password> přihlášení, heslo
- **CWD** <pathname> změna pracovního adresáře
- **CDUP** změna pracovního adresáře o úroveň výš
- **PWD** zjištění pracovního adresáře
- **MKD** <pathname> vytvoření adresáře
- **RMD** <pathname> zrušení adresáře
- **LIST** <pathname>] download výpisu adresáře
- **NLST** [<pathname>] download seznamu jmen souborů v adresáři
- **RETR** <pathname> download souboru ze serveru ke klientovi

Důležité FTP příkazy (2)

- **STOR** **<pathname>** upload souboru od klienta do serveru
- **APPE** **<pathname>** upload souboru, existuje-li soubor, jsou data připojena za něj - append
- **PASV** přechod do pasivního módu
- **PORT** **<host-port>** adresa, kam se má server připojovat v aktivním módu
- **RNFR** **<pathname>** přejmenování nebo přesun souboru, parametr je cesta k souboru
- **RNTO** **<pathname>** přejmenování nebo přesun souboru, parametr je nový název nebo nová cesta
- **DELE** **<pathname>** smazání souboru
- **TYPE** **<type-code>** typ přenášených dat type-code může být např. I (image) - binární data, nebo A - ASCII
- **NOOP** no operation - tímto příkazem se často zabraňuje "vytimeoutování" řídicího spojení
- **ABOR** přerušování datového přenosu
- **QUIT** ukončení komunikace

Příklady komunikace mezi klientem a serverem

- download souboru v aktivním módu (server má IP 10.0.0.1, klient má IP 10.0.0.2) :

(klient provede connect() z 10.0.0.2:1121 na 10.0.0.1:21)

220 System ready for new user.

USER anonymous

331 User name okay, need password.

PASS heslo

230 User logged in, proceed.

TYPE I

200 Type set to I.

(klient provede listen() na port 1123)

PORT 10,0,0,2,4,99

200 PORT command successful.

RETR file.zip

(server provede connect() z 10.0.0.1:20 na 10.0.0.2:1123)

150 Opening BINARY mode data connection for /file.zip (10545616 bytes).

(server posílá soubor klientovi)

(server zavře svoji stranu datového spojení, klient v reakci na to také zavře
svoji stranu datového spojení)

226 Transfer complete.

QUIT

221 System closing control connection.

(klient i server zavřou řídicí spojení)

Příklady komunikace mezi klientem a serverem

- download souboru v pasivním módu (server má IP 10.0.0.1, klient má IP 10.0.0.2)

```
(klient provede connect() z 10.0.0.2:1124 na 10.0.0.1:21)
```

```
220 System ready for new user.
```

```
USER anonymous
```

```
331 User name okay, need password.
```

```
PASS heslo
```

```
230 User logged in, proceed.
```

```
TYPE I
```

```
200 Type set to I.
```

```
PASV
```

```
(server provede listen() na port 2529)
```

```
227 Entering Passive Mode (10,0,0,1,9,225)
```

```
RETR file.zip
```

```
(klient provede connect() z 10.0.0.2:1125 na 10.0.0.1:2529)
```

```
150 Opening BINARY mode data connection for /file.zip (10545616 bytes).
```

```
(server posílá soubor klientovi
```

```
(server zavře svoji stranu datového spojení, klient v reakci na to také  
zavře svoji stranu datového spojení)
```

```
226 Transfer complete.
```

```
QUIT
```

```
221 System closing control connection.
```

```
(klient i server zavřou řídicí spojení)
```

Zhodnocení FTP

■ Výhody :

- Rychlost
- Jednoduchost
- Ještě stále hojně používaný
- Jednoduchý nástroj pro sdílení dat v LAN

■ Nevýhody

- Jednoznačně bezpečnost !
- Je použito mnoho TCP/IP spojení
- Firewall může blokovat stahování
- Je možné zachytávat data třetím počítačem
- Nevhodné pro stahování velkého množství malých souborů

Použité zdroje

- RFC 959 (J. Postel, J. Reynolds 1985: FILE TRANSFER PROTOCOL)
 - <http://www.ietf.org/rfc/rfc959.txt>
- Ladislav Prošek, MFF CUNI
 - <http://www.ms.mff.cuni.cz/~prosl0am/ftp.html>
- Diplomová práce - FTP Filesystem, Libor Martínek, 2000, ČVUT
 - <http://remotefs.netbeans.org/project/remotefs/diplomka/index.html>
- Archiv článků a přednášek Jiřího Peterky
 - <http://www.earchiv.cz/a93/a333c110.php3>
- Otevřená encyklopedie, Wikipedia
 - <http://cs.wikipedia.org/wiki/FTP>

Závěr

Děkuji za pozornost a nyní prosím dotazy