

Semestrální práce PA2
ČVUT FIT



DummyFTP Server

Implementace FTP serveru

Marek Hakala

Praha 2010

Obsah

Zadání.....	3
Klíčová slova.....	3
Použité technologie.....	3
Qt Framework	3
Oxygen	3
Nastavení.....	4
Hlavní konfigurační soubor.....	4
Seznam uživatel v konfiguračním souboru.....	5
Nastavení pomocí vstupních argumentů.....	6
Implementace FTP protokolu.....	7
Ladění aplikace.....	9
Závěr.....	9
Použitá literatura	10
FTP.....	10
Qt4.....	10
C++.....	10
Obecné.....	10

Zadání

Program bude sloužit k přenosu souboru prostřednictvím protokolu FTP (File Transfer Protocol). Určitě stojí za to zmínit, že program bude používat síťovou architekturu klient-server. Aplikace po startu provede načtení konfiguračního souboru. Na základě konfigurace začne program poslouchat na daném TCP portu a soketu. Jakmile se připojí k serveru klient bude vytvořeno speciální vlákno v rámci, kterého s ním bude komunikováno. Prvním krokem bude autorizační mechanismus, který provede ověření uživatelského jména a hesla oproti speciálnímu souboru. Po řádné autorizaci bude uživateli zpřístupněn obsah sdíleného adresáře a bude mít možnost obousměrného přenosu souborů.

Klíčová slova

FTP, C++, Qt4

Použité technologie

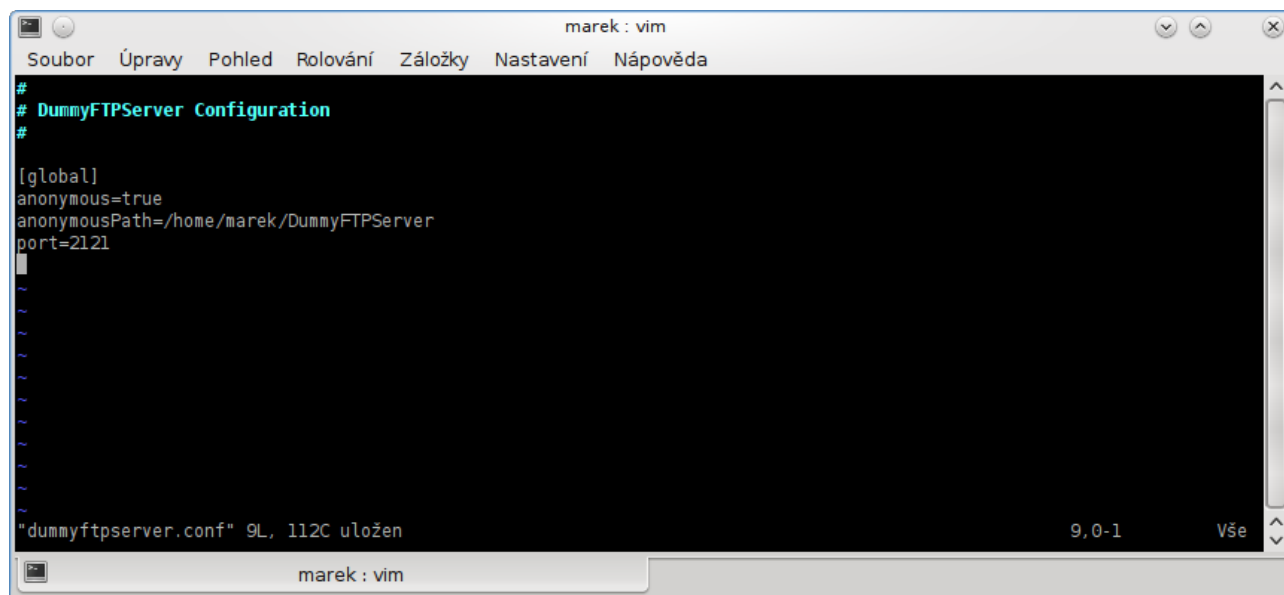
Qt Framework

S Qt frameworku jsem snažím využívat co nejméně věcí. Především jsem ho zvolil k realizaci, protože nabízí zázemí pro síťování, vlákna a práci s adresáři/soubory. Použita byla pouze Console Application varianta aby k překladu nebyly zapotřebí knihovny, které slouží ke kreslení grafiky.

Oxygen

S tohoto open source projektu, používám pouze jednu ikonku. A to hlavní ikonku aplikace, kterou jsem si upravil dle svých potřeb jak mi umožňuje licence LGPL.

Hlavní konfigurační soubor



```
# DummyFTPServer Configuration
#

[global]
anonymous=true
anonymousPath=/home/marek/DummyFTPServer
port=2121
```

Hlavní konfigurační soubor ftp serveru

Slouží k nastavení vlastností ftp serveru. Jelikož se jedná o první verzi jsou zde pouze 3 položky, které můžete měnit.

Konfigurační soubor je rozdělen do kategorií, momentálně je využita pouze globální kategorie „global“. Kategorie dále obsahují položky do, kterých je možné ukládat data a to v párech klíč + hodnota. Na úrovni kódu je tato část ošetřena pomocí mapy. Výhodou tohoto řešení je velice pohodlný přístup k nastavení pomocí názvu sekce a klíče. Pro zrychlení jsou data, ze souboru načtena přímo do paměti po zapnutí aplikace.

Dále konfigurační soubor umožňuje zápis komentářů a to pomocí mřížek ale pouze na novém řádku.

Hodnoty v sekci „global“

anonymous	Povoluje anonymní přístup.
anonymousPath	Nastavuje domovský adresář pro anonymní přístup.
port	Nastavuje port na, kterém bude FTP server poslouchat.

Zápis kategorie

Vzor : [Název kategorie]

Zápis sekce musí být na novém řádku a nesmí obsahovat nic jiného ani komentáře.

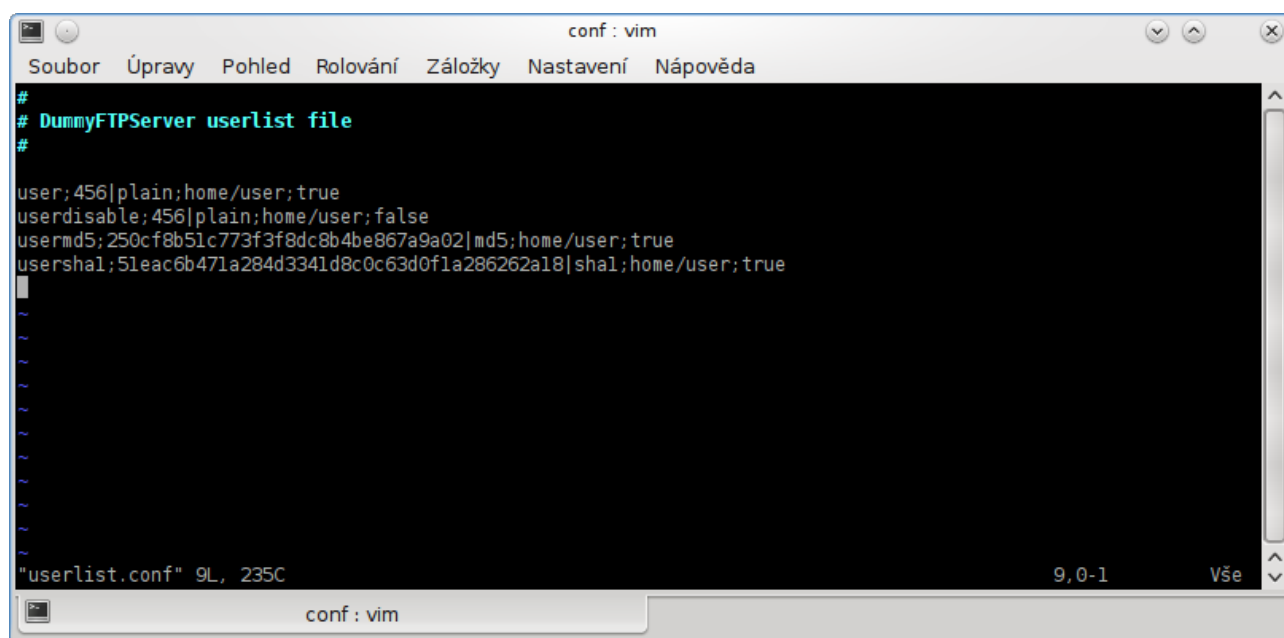
Při špatném zápisu je načítání sekce zahazeno a položky budou načteny do jiné dříve používané kategorie.

Zápis hodnoty

Vzor : klíč=hodnota

Zápis klíče a hodnoty musí být na novém řádku. A nesmí obsahovat nic jiného ani komentáře. Při špatném zápisu je hodnota zahozena.

Seznam uživatel v konfiguračním souboru



Konfigurační soubor s uživatelskými jmény

V tomto souboru jsou záznamy o uživatelských účtech. Jeden záznam odpovídá jednomu řádku. Jednotlivé položky jsou odděleny pomocí středníku. I zde platí stejné pravidlo ohledně komentářů. Mohou být použity pomocí mřížky ale pouze na novém řádku.

Zápis záznamu

Vzor : jméno;heslo|formát;/domovský adresář;blokování účtu

Zápis záznamu musí být na novém řádku. A nesmí obsahovat nic jiného ani komentáře. Při špatném zápisu je hodnota zahozena.

Zápis hesla

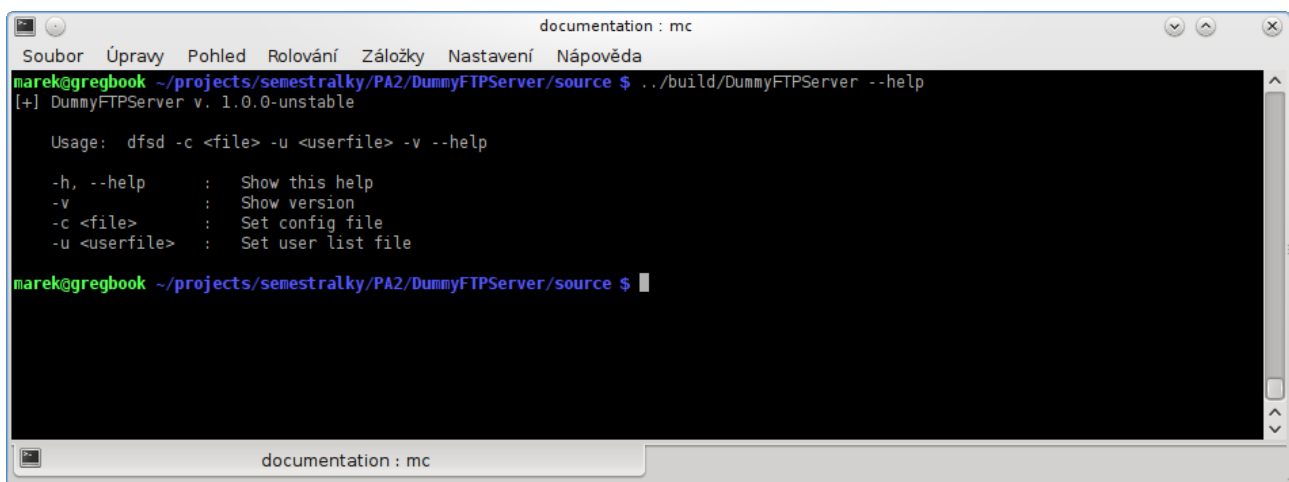
Vzor : heslo|formát

Heslo se zapisuje ve vzorovém formátu. První položka je hash hodnota nebo v případě plain formátu heslo v textové podobě. Formát určuje v jak je zapsána hodnota hesla.

Jsou podporovány celkem tři formáty a to dva hashovací a jeden textový.

plain	Plně textový zápis.
md5	Hash hodnota vytvořená pomocí MD5 algoritmu.
sha1	Hash hodnota vytvořená pomocí SHA1 algoritmu.

Nastavení pomocí vstupních argumentů



```
documentation : mc
Soubor  Úpravy  Pohled  Rolování  Záložky  Nastavení  Nápověda
marek@gregbook ~/projects/semestralky/PA2/DummyFTPServer/source $ ../build/DummyFTPServer --help
[+] DummyFTPServer v. 1.0.0-unstable

Usage: dfdsd -c <file> -u <userfile> -v --help

-h, --help      : Show this help
-v              : Show version
-c <file>       : Set config file
-u <userfile>    : Set user list file

marek@gregbook ~/projects/semestralky/PA2/DummyFTPServer/source $
```

Výpis nápovědy.

Aplikaci je možné pomocí vstupních argumentů nastavit cesty k jednotlivým konfiguračním souborům.

Nastavení cesty ke konfiguračnímu souboru

Vzor : -c <Cesta/ke/konfiguračnímu/souboru>

Cesta k seznamu uživatel

Vzor : -u <Cesta/k/seznamu/uživatel>

Parametry je možné kombinovat. Pro více informací slouží výpis nápovědy pomocí parametru -h.

Implementace FTP protokolu

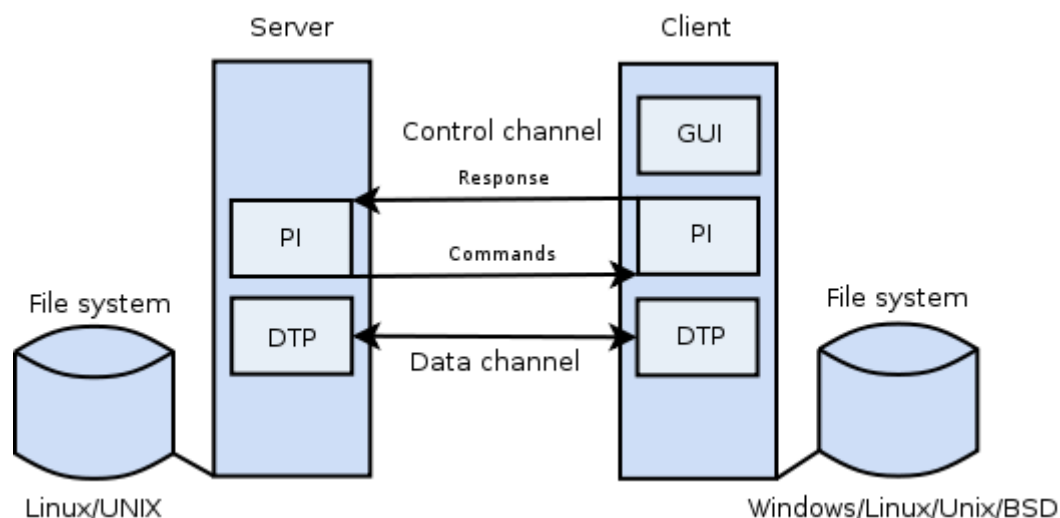


Diagram FTP relace

Protokol FTP patří mezi nejstarší protokoly využívající modelu klient-server. Komunikace je rozdělena do dvou komunikačních kanálů. Kontrolního a Datového.

GUI	Grafické klientské rozhraní.
PI	Interpret FTP protokolu. (Protocol Interpreter)
DTP	Správce datových přenosů. (Data Transfer Process)

Implementace FTP

V semestrální práci jsem implementoval minimum potřebné pro chod FTP serveru. Základem celého FTP serveru je několik tříd. Mezi nejzákladnější objekty patří **FTPServer**, **ServerThread** a **WorkPath**.

Vytvořená implementace by měla zvládat aktivní i pasivní režim přenosu.

FTPServer slouží k obsluze serveru. A je prvním objektem, který začne pracovat s nově přichozím klientem. Vytvoří pro klienta speciální vlákno.

ServerThread reprezentuje serverové vlákno. Obsahuje metody na zpracování a zasílání zpráv. Pro nového klienta je vytvořena instance této třídy a v rámci instance je obsluhován. Nejzajímavější metodou třídy je *parseCommand*. Provádí zpracování příchozích zpráv. Na rozsáhlejší operace se používají pomocné třídy *FTPCommandLIST*, *FTPCommandRETR* a *FTPCommandSTOR*.

WorkPath zajišťuje abstraktní vrstvu mezi souborovým systémem a FTP serverem. Veškeré FS operace jsou vyřizovány pomocí této třídy.

Tabulka implementovaných FTP příkazů

ABOR	Ruší přenos souborů.
CWD	Mění pracovní adresář.
DELE	Maže soubory.
LIST	Vrací seznam vzdálených soubor a adresářů.
MDTM	Vrací čas poslední modifikace souboru.
MKD	Vytvoří nový adresář.
PASS	Posílá uživatelské heslo.
PASV	Zapíná pasivní režim přenosu.
PORT	Nastavuje port pro datový přenos.
PWD	Vrací aktuální pracovní adresář.
QUIT	Ukončuje spojení mezi klientem a serverem.
RETR	Získává vzdálený soubor.
RMD	Odstraní adresář.
RNFR	Nastavuje soubor, který se bude přejmenovávat.
RNTO	Přejmenuje soubor.
SIZE	Vrací velikost souboru v bajtech.
STOR	Ukládá soubor.
TYPE	Nastavuje typ přenosu.
USER	Nastavuje uživatelské jméno.
APPE	Ukládá soubor. S tím, že umí pokračovat.
NOOP	Nevykonává žádnou činnost. Slouží k udržování spojení.
SYST	Vrací typ systému na, kterém server běží.

Ladění aplikace

K testování jsem používal hned několik nástrojů. A to Wireshark, Netcat, gdb a výpisky na stderr.

Wireshark

Software sloužící k sledování síťové komunikaci mezi server a klientem. Wireshark mě v mnoha případech ušetřil spoustu času. Díky velmi propracovanému rozhraní na odposlouchávání tcp komunikace.

Netcat

Jednoduchá aplikace sloužící k debuggování síťových aplikací. Umí se připoj k vzdálenému serveru a v textovém režimu s ním komunikovat. Perfektní věc k zkoušení reakcí serveru na různých ftp příkazů. Pomocí netcatu jsem odhalil několik bugu.

GDB

Nástroj určený k ladění programů. Debugger mi přijde jako základní věc programátora.

Závěr

Myslím si, že se mi povedlo zimplementovat základní podporu pro ftp protocol. Práce byla ale poměrně časově dosti náročná. Nejvíce času jsem strávil za debuggerem dále následoval paketový sniffer a na posledním místě bez pochyby bylo RFC k FTP protokolu. Na druhou stranu si myslím, že jsem při řešení nasbíral spoustu užitečných znalostí a dovedností. Aktuální stav aplikace není určitě ideální ale nebojím se program označit za použitelný. V rámci časových možností mám v plánu aplikaci nadále vyvíjet.

Použitá literatura

FTP

FTP Protocol RFC959 <www.faqs.org/rfcs/rfc959.html>

List of raw FTP commands <www.nsftools.com/tips/RawFTP.htm>

Qt4

Jasmin Blanchette, Mark Summerfield

C++ GUI Programming with Qt 4 (2nd Edition)

Prentice Hall, ISBN: 978-0131872493

Oficiální dokumentace k Qt4

Qt Assistant

C++

Stephen Prata

Mistrovství v C++ 2. aktualizované vydání

Computer Press, ISBN: 80-251-0098-7

Obecné

Wikipedia - Otevřená encyklopedie <<http://cs.wikipedia.org>>

Wikipedia - The free encyclopedia <<http://en.wikipedia.org>>