

به نام خدا

شبکه‌های کامپیوتری
تمرین کامپیوتری شماره یک

اعضای گروه:

مرضیه باقری‌نیا امیری / 810197682

آبتین هیدجی / 810197607

بهار 1400

• ایجاد Connection

ایجاد اتصال همزمان چند Client با کمک thread انجام می‌پذیرد. ابتدا سوکت را می‌سازیم و بعد با تابع `bind` سوکت را به آدرس مربوطه اتصال متصل می‌کنیم؛ سپس در پورت `listen` می‌کنیم تا یک `client` درخواستی ارسال کند. سپس با ایجاد یک thread جدید، آن را `handle` می‌کنیم و وارد محیط `ftp` می‌شود.

• احراز هویت

اطلاعات کاربران عضو سیستم (نام کاربری، رمز عبور و ادمین بودن) در فایل `config.json` ذخیره شده است. آن را با استفاده از `parser` با نام `nlohmann` می‌خوانیم. در ابتدا ورود به `ftp`، کاربر تنها می‌تواند با استفاده از دستور `user` و `pass` احراز هویت خویش را انجام داده و بعد از آن که `login` او تایید شد، می‌تواند از سایر دستورات استفاده کند. قسمت `handle` این دو دستور، از سایر دستورات جدا می‌باشد. هرگاه دستور `user` توسط کاربر ارسال شود، در کلاس `handler` یک متد `do_user` هست که اگر کاربر قبلاً نام کاربری‌اش را وارد نکرده بود و یا اشتباه وارد کرده بود، نام کاربری دریافتی در نام همه کاربران ذخیره شده در `server` جست‌وجو می‌شود و پس از تایید نام کاربری، کاربر باید رمز عبور خود را وارد نماید که هندل کردن دستور `pass` نیز توسط متدی با نام `do_pass` انجام می‌گیرد که `password` وارد شده توسط کاربر به `server` ارسال شده، با اطلاعات مربوط به `username` تطابق داده شده و در صورت تایید، `login` کاربر نهایی شده و از این پس می‌تواند از سایر دستورات هم استفاده نماید.

• دستورات

قبل اجرای هر دستور بررسی می‌کند که کاربر وارد شده، اگر وارد نشده باشد ارور داریم.

- دستور `pwd`: با استفاده از تابع `getcwd` مسیر دایرکتوری فعلی را دریافت کرده و چاپ می‌کند.
- دستور `mkdir`: بررسی می‌کند که دایرکتوری به این اسم (اسمی که همراه با دستور وارد شده است) وجود دارد یا خیر. اگر نه، با تابع `mkdir` (در کتابخانه `stat.h`) دایرکتوری را می‌سازد و `Response` مناسب را می‌فرستد و اگر هم که موجود باشد، با پیغامی مناسب این را به کاربر اطلاع می‌دهد.
- دستور `cwd`: با تابع `chdir` این کار را انجام می‌دهد، اگر مسیر داده شده مجاز و تغییر دایرکتوری موفق باشد این کار انجام شده و اعلام می‌شود وگرنه اعلام عدم موفقیت اتفاق می‌افتد.
- دستور `ls`: ابتدا اتصال مسیر داده را بررسی می‌کند (و اگر برقرار نباشد ارور می‌دهد)؛ یک پردازش ایجاد می‌کند و آن را `fork` می‌کند. با `pipe` کردن آن اطلاعات داخل دایرکتوری را گرفته و اگر بتواند اطلاعات را دریافت کند، ریسپانس مربوطه که حاوی تمامی اطلاعات درون دایرکتوری فعلی است را می‌فرستد.
- دستور `rename`: ابتدا چک می‌کند که فایلی که نام آن در ورودی داده شده در مسیر داده شده موجود هست یا خیر! اگر موجود بود و اگر اندازه مناسبی داشته باشد، با استفاده از تابع `rename` نام آن را تغییر می‌دهد.
- دستور `delete`: بعد از بررسی وجود داشتن فایل یا دایرکتوری مورد نظر در مسیر داده شده و قابل حذف بودنش، با تابع `remove` در کتابخانه `stdio.h` آن را حذف می‌کند.

- دستور **retr**: ابتدا بررسی می‌کنیم که اگر کاربر مربوطه Admin نبود، به فایل غیرمجازی دسترسی نخواسته باشد؛ سپس فایل را می‌خوانیم و ذخیره می‌کنیم؛ ابتدا بررسی می‌کنیم که آیا کاربر مربوطه، حجم کافی برای دانلود فایل را دارد یا خیر! اگر داشت، fd سوکت Data را بازیابی می‌کند و محتوای فایل را به آن send می‌کند. در کاربر هم ابتدا با استفاده از دستور PASV پورت و IP سوکت Data را از سرور می‌گیرد و سپس به آن متصل شده و از روی آن اطلاعات فایل read می‌کند و فایل را باز کرده و اطلاعات را در آن بازنویسی می‌کند.
- دستور **help**: در صورت دریافت این دستور، server یک document آماده (string) که حاوی اطلاعات مختصر درباره تمامی دستورات است را برای کاربر ارسال می‌کند.
- دستور **quit**: Response مربوطه را می‌فرستد و برنامه را به صورت موفقیت‌آمیز می‌بندد.

• نحوه Logging

- Client ID: در ابتدا به هر کاربر یک شماره منحصر به فرد اختصاص داده می‌شود.
- Client ## quit: اگر اتصال برقرار نشود یا قطع شود.
- Bad command: دستور وارد شده ناموجود باشد.
- Client ## [cmd: args]: هر دستور دریافتی را با ID منحصر به فرد فرستنده log می‌کنیم.
- Send List Data Error: اگر اجرای دستور ls با مشکل مواجه شود.
- Enter PASV Mode Error: در ورود به حالت PASV (که برای دریافت و ارسال port و IP سوکت مربوط به داده از آن استفاده می‌شود)، با مشکلی برخورد.
- Data connection accept error: اتصال سوکت داده قطع شود.
- send error: ارسال فایل با مشکل مواجه شود.
- send file data error: ارسال فایل از مسیر داده به مشکل برخورد.
- transfer completed: ارسال فایل انجام شد.
- accept error: پذیرش ناموفق یک Client.
- connection from: اعلام منبع اتصال.
- bind error: مشکل در bind کردن سوکت به port.
- port is being used: اتصال به پورت غیر مجاز / پر
- Ftp server is running با موفقیت initialize شد و راه افتاد.

• نحوه اجرا

```
make
./ftp-server 8000
./ftp 127.0.0.1 8000
```