

## HTTP-сервер

Необходимо реализовать GET-запрос, который отдает клиенту содержимое файла имеющего соответствующее расположение. (jpg, png, txt, html, pdf).

## FTP-сервер

Реализация команд LIST и CWD.

## SSH-сервер

речь идет о подобии ssh, т.к. никакого шифрования использоваться не будет. Обмен между сервером и клиентом осуществляется по простому сокету. Необходимо реализовать аутентификацию (имена и пароли можно хранить в текстовом файле). Должна быть возможность выполнять удаленные команды с выводом результата.

## Модели сетевых сервисов

**fork 1** Данная модель создает новый процесс для каждого нового клиента. При запуске сервиса должна быть возможность указывать максимальное количество одновременно запущенных процессов. В случае, если количество клиентов превышает заданный лимит, сервер принудительно закрывает сокет. По окончании обслуживания клиента процесс уничтожается.

```
accept();  
if (count > n) close()  
fork();
```

**prefork 1** данная модель отличается от предыдущей тем, что головной процесс создает дочерние, которые будут обрабатывать запросы клиентов, заранее. Когда клиент обращается к серверу, то обслуживание передается одному из существующих процессов. Головной процесс должен следить за тем, чтобы количество созданных процессов, ожидающих запросов клиентов, поддерживалось на заданном уровне. По окончании обслуживания клиента процесс уничтожается.

```
while (n) fork();  
if (parent){  
accept();  
file = read();  
}  
else {  
read(fd[0]);  
handle(file);
```

```
return;  
}
```

**prefork 2 (threadpool)** данная модель схожа с предыдущей, но по завершению обслуживания процессы не уничтожаются, а переходят в состояние “ожидания”. При появлении нового запроса клиента, ожидающий процесс пробуждается и выполняет все необходимые действия.

```
while (n) fork();  
if (parent){  
    accept();  
    file = read();  
}  
else {  
    read(fd[0]);  
    handle(file);  
    wait();  
}
```

Модели thread 1, 2 и 3 соотв. моделям fork 1, prefork 1 и prefork 2, но для реализации модели используются нити, а не процессы.

### **Задания**

1. HTTP-сервер. Модель fork 1
2. HTTP-сервер. Модель prefork 1
3. HTTP-сервер. Модель prefork 2
4. HTTP-сервер. Модель thread 1
5. HTTP-сервер. Модель thread 2
6. HTTP-сервер. Модель thread 3
7. FTP-сервер. Модель fork 1
8. FTP-сервер. Модель prefork 1
9. FTP-сервер. Модель prefork 2
10. FTP-сервер. Модель thread 1
11. FTP-сервер. Модель thread 2
12. FTP-сервер. Модель thread 3
13. SSH-сервер. Модель fork 1
14. SSH-сервер. Модель prefork 1
15. SSH-сервер. Модель prefork 2
16. SSH-сервер. Модель thread 1
17. SSH-сервер. Модель thread 2
18. SSH-сервер. Модель thread 3

Источник: <http://ilyakitaev.livejournal.com/2380.html>