

Министерство цифрового развития, связи и массовых коммуникаций
Российской Федерации Сибирский Государственный Университет
Телекоммуникаций и Информатики СибГУТИ

Кафедра вычислительных систем

Курсовая работа
по дисциплине «Сетевое программирование»
на тему
Протокол FTP

Выполнил:
Студент группы ИВ-921
Ярошев Р.А.

Работу проверил:
Ревун А. Л.

Новосибирск 2022

Оглавление

Введение.....	3
Соединение и передача данных.....	4
Аутентификация.....	6
Основные команды.....	7

Введение

FTP (англ. File Transfer Protocol) — протокол передачи файлов по сети. Является одним из старейших прикладных протоколов. Изначально FTP работал поверх протокола NCP, на сегодняшний день широко используется для распространения ПО и доступа к удалённым хостам.

Протокол построен на архитектуре «клиент-сервер» и использует разные сетевые соединения для передачи команд и данных между клиентом и сервером. Пользователи FTP могут пройти аутентификацию, передавая логин и пароль открытым текстом, или же, если это разрешено на сервере, они могут подключиться анонимно. Можно использовать протокол SSH для безопасной передачи, скрывающей (шифрующей) логин и пароль, а также шифрующей содержимое.

Первые клиентские FTP-приложения были интерактивными инструментами командной строки, реализующими стандартные команды и синтаксис. С тех пор были разработаны графические пользовательские интерфейсы для многих используемых по сей день операционных систем. Среди этих интерфейсов как компоненты программы общего веб-дизайна вроде Microsoft Expression Web, так и специализированные FTP-клиенты (например, FileZilla).

Достаточно яркая особенность протокола FTP в том, что он использует множественное (как минимум — двойное) подключение. При этом один канал является управляющим, через который поступают команды серверу и возвращаются его ответы, а через остальные происходит собственно передача данных, по одному каналу на каждую передачу. Поэтому в рамках одной сессии по протоколу FTP можно передавать одновременно несколько файлов, причём в обоих направлениях. Для каждого канала данных открывается свой TCP порт, номер которого выбирается либо сервером, либо клиентом, в зависимости от режима передачи.

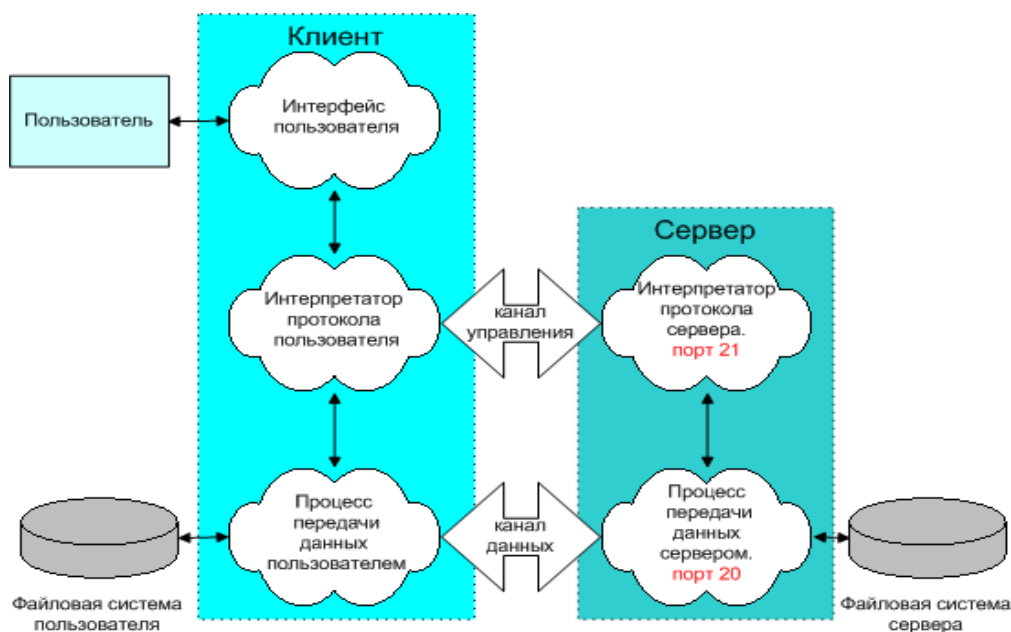
Протокол FTP имеет двоичный режим передачи, что сокращает накладные расходы трафика и уменьшает время обмена данными при передаче больших файлов.

Начиная работу через протокол FTP, клиент входит в сессию, и все операции проводятся в рамках этой сессии (проще говоря, сервер помнит текущее состояние).

Соединение и передача данных

Сервер отвечает по потоку управления трёхзначными ASCII-кодами состояния с необязательным текстовым сообщением. Например, «200» (или «200 OK») означает, что последняя команда была успешно выполнена. Цифры представляют код ответа, а текст — разъяснение или запрос. Текущая передача по потоку данных может быть прервана с помощью прерывающего сообщения, посылаемого по потоку управления.

FTP может работать в активном или пассивном режиме, от выбора которого зависит способ установки соединения. В активном режиме клиент создаёт управляющее TCP-соединение с сервером и отправляет серверу свой IP-адрес и произвольный номер клиентского порта, после чего ждёт, пока сервер запустит TCP-соединение с этим адресом и номером порта. В случае, если клиент находится за брандмауэром и не может принять входящее TCP-соединение, может быть использован пассивный режим. В этом режиме клиент использует поток управления, чтобы послать серверу команду PASV, и затем получает от сервера его IP-адрес и номер порта, которые затем используются клиентом для открытия потока данных с произвольного клиентского порта к полученному адресу и порту.



При передаче данных по сети могут быть использованы четыре представления данных:

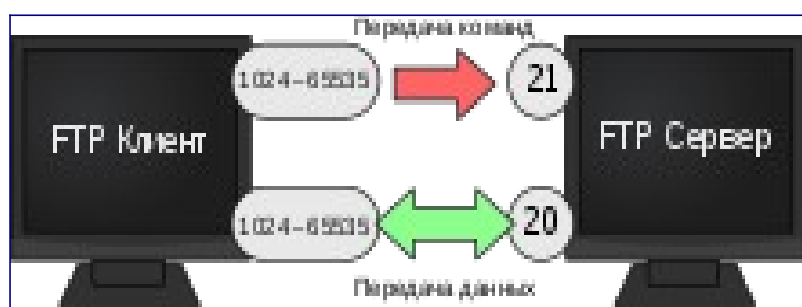
- ASCII — используется для текста. Данные, если необходимо, до передачи конвертируются из символьного представления на хосте-отправителе в «восьмибитный ASCII», и (опять же, если необходимо) в символьное представление принимающего хоста. Этот режим не подходит для файлов, содержащих не только обычный текст.
- Режим изображения (обычно именуемый бинарным) — устройство-отправитель посылает каждый файл байт за байтом, а получатель сохраняет поток байтов при получении. Поддержка данного режима была рекомендована для всех реализаций FTP.
- EBCDIC — используется для передачи обычного текста между хостами в кодировке EBCDIC. В остальном этот режим аналогичен ASCII-режиму.
- Локальный режим — позволяет двум компьютерам с идентичными установками посылать данные в собственном формате без конвертации в ASCII.

Передача данных может осуществляться в любом из трёх режимов:

- Поточный режим — данные посылаются в виде непрерывного потока, освобождая FTP от выполнения какой бы то ни было обработки. Вместо этого вся обработка выполняется ТСП. Индикатор конца файла не нужен, за исключением разделения данных на записи.
- Блочный режим — FTP разбивает данные на несколько блоков (блок заголовка, количество байт, поле данных) и затем передаёт их ТСП.
- Режим сжатия — данные сжимаются единым алгоритмом (обычно кодированием длин серий).

Аутентификация

FTP-аутентификация использует схему имя пользователя/пароль для предоставления доступа. Имя пользователя посылается серверу командой USER, а пароль — командой PASS. Если предоставленная клиентом информация принята сервером, то сервер отправит клиенту приглашение и начинается сессия. Пользователи могут, если сервер поддерживает эту особенность, войти в систему без предоставления учётных данных, но сервер может предоставить только ограниченный доступ для таких сессий.



Хост, обеспечивающий FTP-сервис, может предоставить анонимный доступ к FTP. Пользователи обычно входят в систему как «anonymous» (может быть регистрозависимым на некоторых FTP-серверах) в качестве имени пользователя. Хотя обычно пользователей просят прислать адрес их электронной почты вместо пароля, никакой проверки фактически не производится. Многие FTP-хосты, предоставляющие обновления программного обеспечения, поддерживают анонимный доступ.

Основные команды

- ABOR — Прервать передачу файла
- CDUP — Сменить каталог на вышестоящий.
- CWD — Сменить каталог.
- DELE — Удалить файл (DELE filename).
- EPSV — Войти в расширенный пассивный режим. Применяется вместо PASV.
- HELP — Выводит список команд, принимаемых сервером.
- LIST — Возвращает список файлов каталога. Список передаётся через соединение данных.
- MDTM — Возвращает время модификации файла.
- MKD — Создать каталог.
- NLST — Возвращает список файлов каталога в более кратком формате, чем LIST. Список передаётся через соединение данных.
- NOOP — Пустая операция.
- PASS — Пароль.
- PASV — Войти в пассивный режим. Сервер вернёт адрес и порт, к которому нужно подключиться, чтобы забрать данные. Передача начнётся при введении следующих команд: RETR, LIST и т. д.
- PORT — Войти в активный режим. Например PORT 12,34,45,56,78,89. В отличие от пассивного режима для передачи данных сервер сам подключается к клиенту.
- PWD — Возвращает текущий каталог.
- QUIT — Отключиться.
- REIN — Реинициализировать подключение.
- RETR — Скачать файл. Перед RETR должна быть команда PASV или PORT.
- RMD — Удалить каталог.
- RNFR и RNT0 — Переименовать файл. RNFR — что переименовывать, RNT0 — во что.

- SIZE — Возвращает размер файла.
- STOR — Закачать файл. Перед STOR должна быть команда PASV или PORT.
- SYST — Возвращает тип системы (UNIX, WIN, ...).
- TYPE — Установить тип передачи файла (бинарный, текстовый).
- USER — Имя пользователя для входа на сервер.