*Федеральное государственное бюджетное образовательное учреждение* *высшего образования*

|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | ***«Московский государственный технический университет  имени Н.Э. Баумана***  ***(национальный исследовательский университет)»***  ***(МГТУ им. Н.Э. Баумана)*** |

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

**Реферат**

**Дисциплина:** Сети и телекоммуникации

**Название реферата:**

FTP-сервер. Особенности работы FTP сервера, основные алгоритмы и команды.

Студент гр. ИУ6-52  **\_\_\_\_\_\_01.12.2018 Бурлаков А.С.**

(Подпись, дата) (И.О. Фамилия)

Преподаватель  **\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_**

(Подпись, дата) (И.О. Фамилия)

Москва, 2018

Оглавление

[Введение 3](#_Toc532127278)

[*1.* FTP-протокол. 3](#_Toc532127279)

[ОСНОВНАЯ ЧАСТЬ 4](#_Toc532127280)

[2. Модели работы FTP. 4](#_Toc532127281)

[3. Команды протокола. 6](#_Toc532127282)

[4. Утилита FTP. 9](#_Toc532127283)

[ЗАКЛЮЧЕНИЕ 11](#_Toc532127284)

[Список использованной литературы **Ошибка! Закладка не определена.**](#_Toc532127285)

# Введение

## FTP-протокол.

Протокол передачи данных FTP является одним из самых старых протоколов в Интернете. С момента создания протокола FTP претерпел множество изменений и значительно расширил свои возможности. Задача FTP - решить проблему удаленного доступа к файлам, прямого или косвенного использования ресурсов удаленного компьютера, обеспечения независимости клиента от файловых систем на удаленных серверах, эффективной и надежной передачи данных. Протокол может использоваться как в пользовательских программах, так и в специальной утилите операционной системы.

Задача FTP - решить проблему удаленного доступа к файлам, прямого или косвенного использования ресурсов удаленного компьютера, обеспечения независимости клиента от файловых систем на удаленных серверах, эффективной и надежной передачи данных.

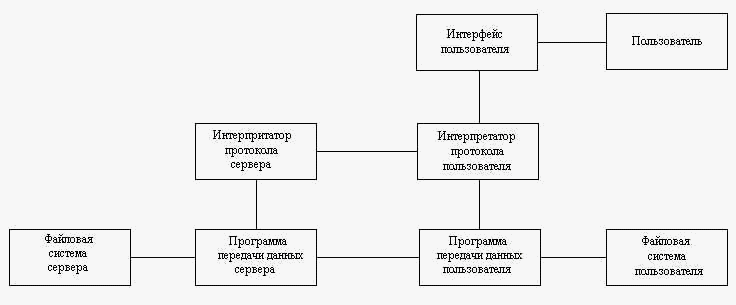
Данные FTP передаются по каналу TCP с использованием технологии клиент-сервер. Однако этот протокол не защищает передаваемые данные и передает незащищенный открытый текст между сервером и клиентом. FTP-сервер может требовать аутентификации клиента. Однако пароль, как и любая другая информация, передается открытым текстом.

Протокол имеет двоичный режим передачи, который снижает стоимость трафика и уменьшает время передачи данных для больших файлов. Когда вы начинаете работать с протоколом FTP, клиент входит в сеанс, и все операции передачи выполняются во время этого сеанса.

# ОСНОВНАЯ ЧАСТЬ

## Модели работы FTP.

Простейшая модель работы протокола FTP представлена на рисунке 1. Соединение создается интерпретатором протокола пользователя. Обменом управляет канал управления в стандарте протокола TELNET. FTP-команды генерируются интерпретатором протокола пользователя и передаются на сервер. По каналу управления также и отправляются ответы сервера. В общем случае пользователь имеет возможность установить контакт с интерпретатором протокола сервера и отличными от интерпретатора протокола пользователя средствами.

Рисунок 1 – Принципиальная схема работы FTP-сервера

Команды FTP устанавливают параметры канала передачи данных, а также параметры самого процесса передачи. Помимо этого они определяют характер работы с локальной и удаленной файловыми системами.

При организации канала передачи данных устанавливается отличная от процесса организации канала управления последовательность действий, при этом сервер создает обмен данными в соответствии с установленными в сессии параметры. Сама же сессия инициализирует канал передачи данных.

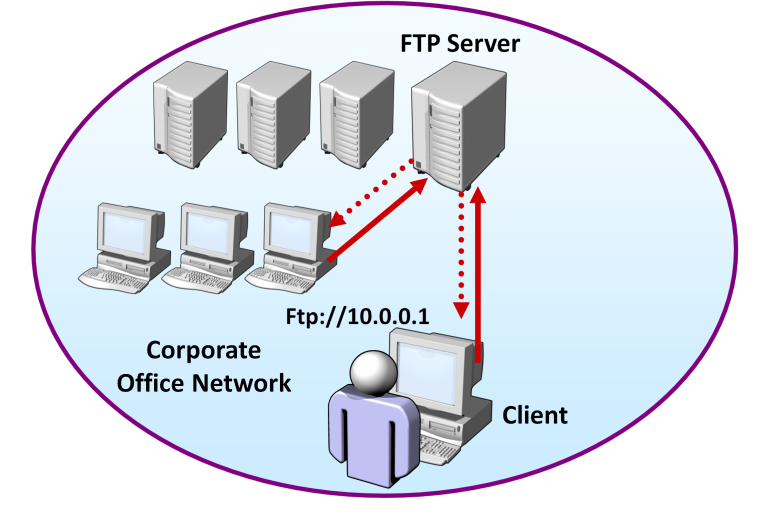


Рисунок 2 – Связь клиента и FTP-сервера

Канал данных устанавливается для того же хоста, что и канал управления, через который ведется настройка канала данных. Канал данных может быть использован как для приема, так и для передачи данных.

Алгоритм работы протокола FTP состоит в следующем:

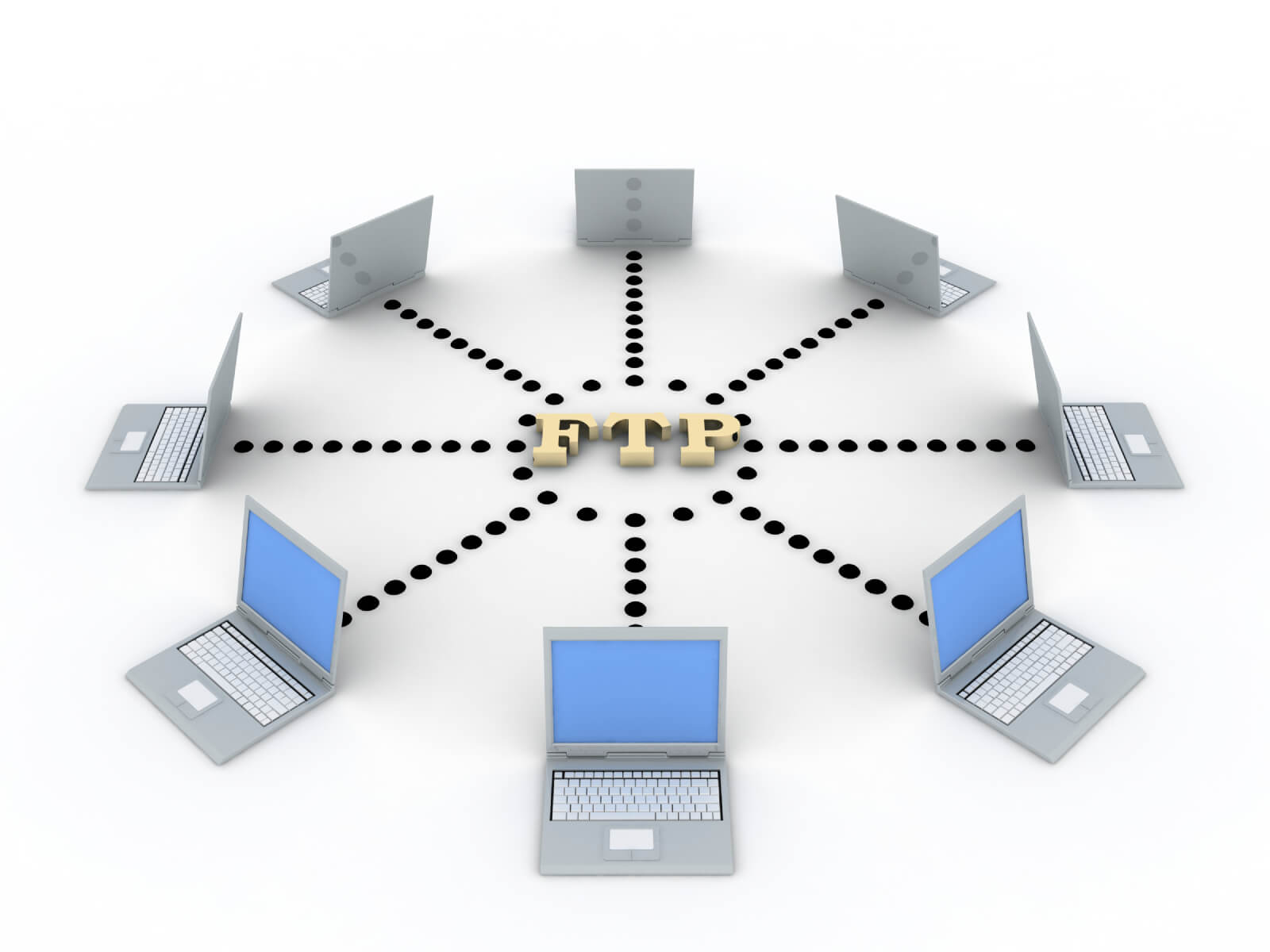
1. Порт TCP 21 постоянно находится в состоянии возникновения соединения со стороны пользователя FTP, которое используется в качестве управляющего соединения.
2. После установления управляющего соединения пользователь получает возможность отправлять на сервер команды. Сами же эти команды определяют параметры соединения передачи данных.
3. После согласования всех параметров канала передачи данных тот из участников соединения, который становится пассивным, становится в режим ожидания открывания соединения на порт, заданный для передачи данных. После чего этот активный модуль открывает соединение и начинает передачу данных.
4. После конца передачи данных соединение закрывается, однако управляющее соединение остается открытым, поэтому пользователь может еще раз открыть канал передачи данных не закрывая сессии FTP.

Механизм установления соединения между портами и выбор параметров передачи составляют основу передачи данных FTP. Каждый из участников FTP-соединения поддерживает порт передачи данных. Однако, участниками эти порты используются, выбранные для них интерпретатором протокола пользователя, так как из управляющих процессов, участвующих в соединении, только этот может изменить порты передачи данных.

Сервер FTP берет ответственность за открытие и закрытие канала передачи и поэтому должен сам закрывать канал передачи данных в указанных ниже случаях:

1. Если сервер окончил передачу данных в том формате, который требует закрытия соединения;
2. Если сервер получил команду «прерви соединение»;
3. В том случае, если пользователь каким-либо образом изменил параметры порта передачи
4. Если пользователь закрыл управляющее соединение.
5. В случае возникновения ошибок, при который возобновить передачу данных не получается.

## Команды протокола.



Все команды управления передачи данных, с помощью которых обмениваются информацией “Интерпретатор протокола сервера” и “Интерпретатор протокола пользователя”, можно разделить на три группы, а именно на команды:

* Управления доступом к системе;
* Управления потоком данных;
* FTP-сервиса.

Для каждой группы рассмотрим самые характерные команды. Среди команд управления доступом к системе следует отметить следующие:

*USER*. Эта команда открывает сессию FTP между клиентом и сервером. Параметры, передаваемые команде – это имя пользователя, или его идентификатор для работы с файловой системой. Команда может использоваться в начале или в середине сессии, к примеру, если пользователь хочет изменить идентификатор, от которого будут проводиться все действия. В этом случае все переменные, которые относятся к старому имени пользователя освободятся.

*PASS*. Эта команда используется после ввода имени пользователя и использует в качестве аргумента пароль пользователя. Следует помнить, что данные аутентификации FTP передаются по сети незащищенным текстом, поэтому пользователю предварительно следует использовать методы шифрования для защиты пароля.

*CWD*. Данная команды позволяет пользователю работать с каталогами удаленной файловой системы. Команде передается строка, которая указывает путь каталога удаленной ФС, в котором будет работать пользователь.

*REIN*. Команда реинициализации. Эта команда очищает все переменные текущего пользователя, сбрасывает параметры соединения. Если в момент подачи команды происходит передача данных, передача продолжается и завершается с прежними параметрами.

*QUIT*. Команда закрывает управляющий канал. Если в момент подачи команды происходит передача данных, канал закрывается после окончания передачи данных.

Команды управления потоком же устанавливают все параметры передачи данных, поэтому все параметры, которые описываются этими командами, имеют значение по умолчанию. Команды управления потоком же используются в том случае, когда нужно поменять значение параметров передачи, которые используются по умолчанию. Среди всех команд управления потоком данных следует выделить следующие:

PORT. Команда назначает адрес и порт хоста, который будет использоваться как активный участник соединения по каналу передачи данных. Аргументами команды являются 32-битный IP адрес и 16-битный номер порта соединения. Эти значения разбиты на шесть 8-битных полей и представлены в десятичном виде: h1, h2, h3, h4, p1, p2, где hN - байты адреса (от старшего к младшему), а pN - байты порта (от старшего к младшему).

PASV. Эта команда отправляется модулю, который будет играть пассивную роль в передаче данных (“слушать” соединение). Ответом на данную команду должна быть строка, содержащая адрес и порт хоста, находящиеся в режиме ожидания соединения в формате команды PORT — “h1, h2, h3, h4, p1, p2”.

Команды TYPE, STRU, MODE определяют, соответственно, тип передаваемых данных (ASCII, Image и другие), структуру или формат передачи данных (File, Record, Page), способ передачи (Stream, Block и другие). Использование этих команд очень важно при построении взаимодействия в гетерогенных средах и весьма отличающихся операционных и файловых систем взаимодействующих хостов.

Все команды сервиса FTP определяют действия, которые нужно сделать с указанными файлами. Параметры команд этой группы – это путь к конкретному файлу. Синтаксис передаваемого в качестве параметра пути должен удовлетворять требованиям формата ФС обработчика данной команды. Из всех команд FTP-сервиса можно выделить следующие:

RETR. Данная команда указывает модулю программы передачи данных сервера передать копию файла, который задан как параметр данной команды модулю передачи данных на том конце соединения.

STOR. Эта команда указывает модулю “Программа передачи данных сервера” получить данные по указанному каналу передачи данных и говорит сохранить в качестве файла, имя которого задано аргументом этой команды. Однако, если такой файл уже существует, то он будет замещен новым, в случае, если он не существует, будет создан новый.

Команды RNFR и RNTO должны следовать одна за другой. Первая команда содержит в качестве аргумента старое имя файла, вторая - новое. Последовательное применение этих команд переименовывает файл.

ABOR. Эта команда указывает серверу прервать выполнение предшествующей сервисной команды (к примеру, передачу файла) и закрыть канал передачи данных.

Команда DELE удаляет указанный в качестве параметра файл.

Команды MKD и RMD, в свою очередь, создают и удаляют указанный в аргументе каталог.

При помощи команд LIST и NLST можно получить список файлов в указанном каталоге.

Все команды FTP-протокола отправляются “Интерпретатором протокола пользователя” в текстовом виде - по одной команде в строке. Каждая из строк команды - идентификатор и параметры - заканчиваются символами <CRLF>. Имя же самой команды отделяется от параметров символом пробела, а именно - <SP>.

Обработчики команд возвращают коды обработки каждой команды, состоящие из трёх цифр. Коды обработки же составляют иерархическую структуру и, в целом, определенная команда может возвратить лишь определенный набор котов. За кодами обработки команды следуют символы пробела, после чего следует текст пояснения. Так, к примеру, строка успешного завершения операции следуют следующим образом: “200 Command okay”.

## Утилита FTP.

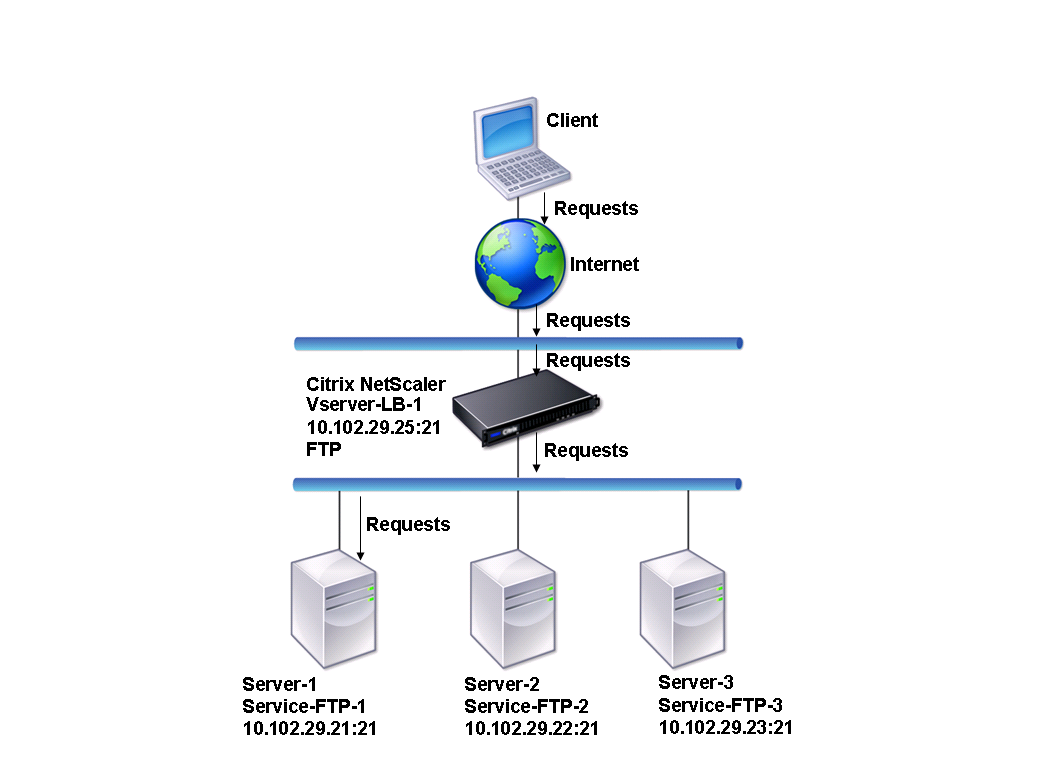


Рисунок 3 – Схема связи сервера и клиента

Некоторые утилиты используют службу командной строки, некоторые имеют графический интерфейс, но, как правило, они поддерживают все стандарты протокола FTP и определяют основные команды, которые должны поддерживаться всеми вариантами приложения FTP. Типичный пользователь FTP чаще известен как утилита, позволяющая обмениваться файлами между серверами FTP и клиентскими компьютерами.

Утилита FTP предоставляет пользователю оболочку командной строки. Команды Утилиты FTP обычно построены по принципу работы с локальной файловой системой и полностью отличаются от команд FTP. В зависимости от приложения и версии приложения FTP оно может поддерживать различные функции для работы с удаленной файловой системой.

Пользователи информируются обо всех командах и событиях передачи данных в соответствии с интерпретацией этих событий утилитой FTP. По сути, утилита FTP является интерпретатором команд оболочки (и их последовательностей) протокола FTP. Отправка команд FTP, анализ ответов FTP-сервера, открытие канала передачи данных, процесс передачи данных - все это в модуле FTP.

# ЗАКЛЮЧЕНИЕ

Протокол передачи файлов (File Transfer Protocol – FTP) – это приложение клиент-сервер протокола TCP/IP для копирования файлов из одного хоста к другому.

FTP требует двух соединений для передачи данных: соединение управления и соединение данных.

FTP использует *NVT* ASCII для связи между непохожими системами.

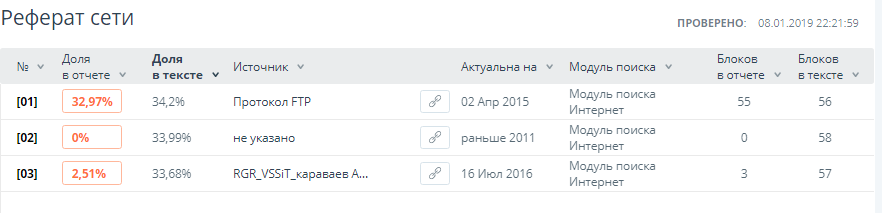
Перед активной передачей файла клиентом определяется тип файла, структура данных и режим передачи.

Имеется пять классов команд, посылаемых клиентом для установления связи с сервером:

* Команды доступа.
* Команды управления файлом.
* Команды определения порта.
* Команды передачи файла.
* Прочие команды.

Реферат проверен на плагиат с помощью сервиса www.antiplagiat.ru:





# сПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Курс лекций по предмету “Сети и телекоммуникации”.
2. НОУ Институт. Протоколы передачи файлов (FTP и TFTP)[Электронный ресурс]. – Режим доступа: URL: https://www.intuit.ru/studies/courses/2/2/lecture/52 (09.12.2018).