

IBM

@server

iSeries

FTP





@server

iSeries

FTP

Содержание

FTP	1
Новое в выпуске V5R2	2
Как напечатать этот раздел	3
Сценарии применения FTP	3
Сценарий: Передача файла с удаленного хоста	3
Сценарий: Защита FTP с помощью SSL	4
Сведения о настройке	6
Настройка сервера FTP	9
Сервер FTP в Навигаторе	10
Настройка серверов FTP для работы с клиентами FTP с графическим интерфейсом и средствами Web	10
Настройка анонимного доступа к FTP	11
Подготовка к настройке анонимного доступа к FTP	12
Создание программ выхода для поддержки анонимного доступа к серверу FTP	13
Создание пользовательского профайла OS/400: ANONYMOUS	14
Создание общей библиотеки или каталога	14
Установка и регистрация программ выхода	14
Защищенный FTP	15
Настройка запрета на доступ к серверу FTP	15
Ограничение доступа к FTP	16
Защита сервера FTP с помощью SSL	18
Создание локальной сертификатной компании	18
Выбор сертификата для сервера FTP	19
Настройка обязательной идентификации клиента на сервере FTP (выполняется при необходимости)	20
Настройка SSL на сервере FTP	21
Защита клиента FTP с помощью TLS или SSL	21
Управление доступом с помощью программ выхода FTP	23
Управление доступом с помощью Навигатора	23
Отслеживание пользователей FTP	24
Управление сервером FTP	24
Запуск и остановка сервера FTP	25
Настройка числа доступных серверов FTP	25
Повышение производительности сервера FTP за счет настройки подсистемы	25
Работа с клиентом FTP в системе iSeries	26
Открытие и закрытие сеанса клиента	26
Информация о тайм-ауте сервера	30
Передача файлов с помощью FTP	30
Запуск FTP в пакетном режиме	31
Справочная информация о FTP	39
Команды сервера FTP	40
ADDM (Добавить элемент физического файла)	42
ADDV (Добавить элемент переменной длины в физический файл)	42
APPE (Добавить к существующему файлу)	42
AUTH (Способ идентификации)	43
CRTL (Создать библиотеку)	43
CRTP (Создать физический файл)	44
CRTS (Создать исходный физический файл)	44
CWD (Изменить рабочий каталог или библиотеку)	44
DEBUG (Включить трассировку сервера FTP)	45
DELE (Удалить файл или документ)	45
DLTF (Удалить файл)	45
DLTL (Удалить библиотеку)	45

HELP (Получение справки с удаленного сервера iSeries)	46
LIST (Список файлов)	46
MKD (Создать каталог)	47
MODE (Задать режим передачи)	47
NLST (Список имен)	47
NOOP (Получить ответ сервера)	48
PASS (Пароль)	48
PASV (Установить пассивное информационное соединение)	48
PBSZ (Размер буфера защиты)	48
PORT (Порт данных)	49
PROT (Уровень защиты канала для передачи данных)	49
PWD (Показать рабочий каталог или библиотеку)	50
QUIT (Завершить сеанс сервера FTP)	50
RCMD (Отправить команду CL в систему сервера FTP)	50
REIN (Повторно инициализировать сеанс)	51
RETR (Получить файл)	51
RMD (Удалить каталог)	51
RNFR (Переименовать из)	52
RNTO (Переименовать в)	52
SITE (Отправить информацию о клиенте)	52
STOR (Записать файл)	54
STOU (Записать файл без замены существующего)	55
STRU (Задать структуру файла)	55
SYST (Определить операционную систему)	55
TIME (Задать значения тайм-аутов для сервера FTP)	56
TYPE (Задать способ представления данных)	56
USER (Отправить ИД пользователя на сервер)	58
Команды клиента FTP	58
ACCT (Отправить учетную запись)	60
APPEND (Добавить элемент локального файла в удаленный файл)	61
ASCII (Изменить тип передачи на ASCII)	61
BINARY (Изменить тип передачи на двоичный)	62
CD (Изменить рабочий каталог или библиотеку)	62
CLOSE (Завершить сеанс FTP с удаленной системой)	62
DEBUG (Трассировка клиента и вывод на экран команд сервера, отправляемых в удаленную систему)	63
DEBUG (Изменить значения тайм-аутов клиента)	64
DELETE (Удалить файл в удаленной системе)	64
DIR (Показать содержимое каталога, библиотеки или файлов)	65
EBCDIC (Изменить тип передачи на EBCDIC)	65
GET (Скопировать файл из удаленной системы в локальную)	66
HELP (Справка по командам FTP)	67
LCD (Изменить рабочую библиотеку или каталог в локальной системе)	68
LOCSITE (Задать информацию о локальной системе)	68
LOCSTAT (Показать информацию о состоянии локальной системы)	69
LS (Показать список удаленных файлов)	70
LTYPE (Локальный режим передачи)	70
MDELETE (Удалить несколько файлов из удаленной системы)	71
MGET (Скопировать несколько файлов из удаленной системы в локальную)	71
MKDIR (Создать каталог)	72
MODE (Задать режим передачи данных)	73
MPUT (Отправить несколько элементов файлов из локальной системы в удаленную)	73
NAMEFMT (Выбрать формат имен файлов)	74
NULLFLDS (Разрешить передачу файлов с пустыми полями)	75
OPEN (Подключиться к серверу FTP удаленной системы)	75
PASS (Отправить пароль)	76

PUT (Скопировать элемент файла из локальной системы в удаленную)	76
PWD (Показать текущий каталог, папку или библиотеку)	77
QUOTE (Отправить команду серверу FTP)	77
REINITIALIZE (Повторно инициализировать сеанс)	78
RENAME (Переименовать файл удаленной системы)	79
RESET (Сброс)	79
RMDIR (Удалить каталог)	79
SECData (Настройка защиты данных)	80
SECOpen (Настройка уровня защиты данных)	80
SENDPASV (Задать опцию передачи команды PASV)	81
SENDPORT (Задать опцию отправки команды PORT)	82
SENDSITE (Опция отправки команды SITE)	83
SITE (Отправить информацию о локальном хосте)	83
STATUS (Получить информацию о состоянии от удаленной системы)	84
STRUCT (Задать структуру файла)	84
SUNIQUE (Опция замены файлов)	85
SYSCMD (Передать команду CL в локальную систему iSeries)	85
TYPE (Задать тип передачи файлов)	86
USER (Отправить ИД пользователя в удаленную систему)	88
VERBOSE (Просмотр сообщений об ошибках)	88
Программы выхода FTP	89
Точка выхода для проверки запроса: клиент и сервер	89
Пример: Программа выхода из процедуры проверки запросов к серверу или клиенту FTP на языке CL	91
Пример: Программа выхода из процедуры проверки запросов к серверу FTP на языке ILE RPG	93
Формат точки выхода VLRQ0100	97
Точка выхода из процедуры входа в систему сервера	101
Пример: Программа выхода из процедуры входа в систему сервера FTP на языке CL	103
Пример: Программа выхода из процедуры входа в систему сервера FTP на языке C	104
Пример: Программа выхода из процедуры входа в систему сервера FTP на языке RPG	113
Формат точки выхода TCPL0100	115
Формат точки выхода TCPL0200	119
Формат точки выхода TCPL0300	125
Удаление программ выхода	128
Способы передачи данных	129
Передача файлов, содержащих упакованные десятичные данные, с одного сервера iSeries на другой	129
Передача файлов *SAVF	130
Передача документов QDLS	131
Передача файлов "root", QOpenSys, QLANSrv, QDLS и QOPT	131
Передача файлов с помощью QfileSvr.400	131
Передача файлов QSYS.LIB	133
Копирование текстовых файлов в библиотеку QSYS.LIB	135
Информация о предварительном создании файлов	135
Преобразование CCSID	136
Задание таблиц преобразования	136
Теги кодовых страниц CCSID для файлов iSeries	137
Информация о NLS, относящаяся к FTP	137
Файловые системы и соглашения о присвоении имен	138
Файловые системы OS/400, поддерживаемые FTP	139
Коды ответа сервера FTP	140
Правила описания формата команд сервера FTP	141
Правила описания формата команд клиента FTP	142
Параметры команд и знаки препинания	143
Имена файлов, применяемые в командах передачи файлов клиента	143

Правила именования передаваемых файлов	145
Устранение неполадок FTP	146
Устранение неполадок FTP	146
Данные, необходимые для создания отчета о неполадках FTP	149
Трассировка сервера FTP	149
Трассировка клиента FTP	152
Работа с заданиями сервера FTP и протоколом задания	152

FTP

Протокол передачи файлов (FTP) позволяет пользователям сервера iSeries™ обмениваться файлами по сети и работать с общими файлами. Кроме того, он позволяет переименовывать, добавлять и удалять файлы в удаленных системах. Перед тем как начать настройку системы для передачи файлов, в ней необходимо запустить и настроить протокол TCP/IP.

Дополнительная информация о работе с FTP на сервере iSeries приведена в следующих разделах:

» Новое в выпуске V5R2

Ознакомьтесь с новыми функциями службы FTP системы iSeries, появившимися в выпуске V5R2.

Как напечатать этот раздел

Ознакомьтесь с информацией о том, как загрузить и напечатать документацию по работе с FTP в формате PDF.

Сценарии применения FTP

Сценарии, содержащие примеры настройки и применения FTP.

Настройка сервера FTP

Информация о настройке сервера FTP для графических клиентов FTP, Web-браузеров и других инструментов с Web-интерфейсом. Дополнительно в этом разделе приведена информация о настройке доступа анонимных пользователей.

Защита FTP

Информация о защите данных, передаваемых с помощью протокола FTP, путем настройки SSL, сбора данных о пользователях FTP и управления доступом к функциям FTP.

Управление сервером FTP

Информация об управлении сервером FTP, в том числе инструкции по запуску и завершению работы сервера, сведения о средствах защиты FTP и сведения о применении SSL.

Работа с клиентом FTP в системе iSeries

Информация об открытии и закрытии сеанса клиента, передаче файлов и запуске пакетных заданий FTP.

Справочная информация об FTP

Информация о командах клиента и сервера, точках выхода FTP, способах передачи данных и т.п.

Устранение неполадок FTP

Ознакомьтесь с информацией об устранении неполадок в работе сервера и клиента FTP.

Примечание:

Ознакомьтесь с важной юридической информацией, приведенной в разделе Отказ от предоставления гарантий на примеры программ.



Новое в выпуске V5R2

В выпуске 5, версии 2 в поддержку FTP на сервере iSeries были внесены следующие изменения:

Добавлены новые функции

- Сервер FTP теперь поддерживает библиотеки, расположенные в независимых пулах вспомогательной памяти (IASP).
- Поддержка протоколов TSL и SSL
В выпуске V5R2 клиент FTP OS/400^R может применять протоколы TSL и SSL (допустима только идентификация сервера). Это дает возможность передавать данные по управляющему соединению FTP (в том числе пароль) в зашифрованном виде. Для поддержки этой функции были внесены следующие изменения:

Добавлены команды клиента:

- SECOpen (Настройка защиты данных)
- SECData (Настройка защиты данных)

Добавлены параметры команды STRTCPFTP:

- Защищенное соединение (SECCNN) (См. стр. 28)
- Защита данных (DTAPROT) (См. стр. 28)

Добавлены параметры команды LOCSITE (См. стр. 69):

- DTAPROT C
- DTAPROT P
- DTAPROT



Пополнена информация


Раздел FTP справочной системы V5R2 Information Center был дополнен новой информацией. Кроме того, в некоторых местах была изменена структура информации, для того чтобы облегчить поиск нужного раздела. Раздел FTP в выпуске V5R2 был дополнен следующей информацией:

- Сценарии FTP - содержит примеры применения FTP. Эти сценарии помогут вам спланировать и задать конфигурацию сервера FTP в системе iSeries.
- Информация о защите сервера FTP с помощью SSL перенесена в этот раздел из раздела, посвященного SSL.
- Примеры программ выхода для настройки доступа анонимных пользователей к FTP, ранее находившиеся на Web-сайте Technical Reference:
 - Точка выхода из процедуры входа в систему сервера - содержит примеры программ выхода из процедуры входа в систему сервера FTP на языках CL, C и ILE RPG.
 - Точка выхода для проверки запросов - содержит примеры программ выхода из процедуры проверки запросов к клиенту и серверу FTP на языках CL и ILE RPG.
- Справочная информация, ранее находившаяся в Справочнике по настройке, добавлена в раздел FTP.

Выделение измененной информации

Все технические изменения, внесенные в данную информацию, отмечены следующим образом:

- Значком  отмечено начало новой или измененной информации.
- Значком  отмечен конец новой или измененной информации.

Более подробная информация о дополнениях и изменениях, внесенных в этот выпуск, приведена в документе Информация для пользователей .

Как напечатать этот раздел

Для просмотра или загрузки документа в формате PDF щелкните на ссылке FTP (около 600 Кб, или 154 страницы).

Сохранение файлов PDF

Для того чтобы сохранить документ в формате PDF на своем компьютере для просмотра или печати, выполните следующие действия:

1. Щелкните правой кнопкой мыши на документе PDF в окне браузера (для этого щелкните правой кнопкой мыши на приведенной выше ссылке).
2. Выберите пункт **Сохранить как...**
3. Перейдите в каталог, в котором нужно сохранить файл PDF.
4. Нажмите кнопку **Сохранить**.

Загрузка продукта Adobe Acrobat Reader

Для просмотра и печати документов в формате PDF применяется программа Adobe Acrobat Reader.

Ее можно загрузить с cWeb-сайта фирмы Adobe(www.adobe.com/products/acrobat/readstep.html) .

Сценарии применения FTP

» Приведенные ниже сценарии помогут вам понять, каким образом работает протокол FTP и для каких целей его можно применять в своей сети. Эти сценарии демонстрируют основные принципы работы FTP, с которыми полезно ознакомиться как начинающим, так и опытным пользователям перед планированием конфигурации и настройкой FTP.

Передача файла с удаленного хоста

Необходимо передать файл в тестовую систему. Для передачи этого файла на удаленный хост воспользуйтесь основной функцией FTP.

Защита FTP с помощью SSL

Необходимо настроить SSL для защиты данных, передаваемых по сети партнеру фирмы.



Сценарий: Передача файла с удаленного хоста



Задача

Предположим, что ваш коллега разрабатывает приложение на Java™ в удаленной системе. В ваши обязанности входит тестирование приложений, поэтому вам нужно передать файл `example.jar` из удаленной системы в локальную тестовую систему. Для передачи файла по сети TCP/IP в двоичном режиме воспользуйтесь обычным FTP. И клиент, и сервер представляют собой системы iSeries, в которых установлена служба FTP OS/400.

Описание

Для передачи файла применяются два соединения: *управляющее соединение* и *информационное соединение*. Управляющее соединение служит для передачи команд с клиента на сервер и получения ответов от сервера. Клиент вызывает команды FTP, которые передаются серверу FTP. Информационное соединение служит для передачи файлов. На клиенте и сервере применяется

файловая система OS/400. Обычно для передачи файлов у пользователя должны быть пользовательские профайлы в обеих системах. Остальные требования перечислены ниже:

- Сервер iSeries с операционной системой OS/400.
- Программа TCP/IP Connectivity Utilities (5722-TC1)
- Настроенный сервер FTP
- Имя хоста удаленной системы
- Имя и пароль пользователя в удаленной системе
- Имя необходимого файла
- Расположение этого файла
- Формат файла (формат, в котором будет передаваться файл, например, текстовый или двоичный)

Задачи настройки

Для передачи файла выполните следующие задачи:

1. Запустите сеанс клиента FTP.
В данном сценарии: В командной строке iSeries введите STRTCPFTP и нажмите **Enter**.
2. Укажите имя удаленной системы, в которую нужно отправить файл.
В данном сценарии: theirco.com
3. Введите имя пользователя для идентификации в удаленной системе.
В данном сценарии:
Введите ID для входа в систему
(свой ID пользователя):
==>ID-пользователя
4. Укажите пароль в удаленной системе.
В данном сценарии:
Введите пароль:
==>пароль
5. Найдите на сервере фирмы TheirCo каталог, в котором расположен необходимый файл.
В данном сценарии: ==>cd /qibm/userdata/os400/dirserv/usertools/windows
6. Перейдите на локальном сервере в каталог, в котором должен быть сохранен файл.
В данном сценарии: ==>lcd /qibm/userdata/os400/dirserv/usertools/windows
7. Укажите тип файла: ASCII или BINARY. По умолчанию тип файла равен ASCII.
Файл .jar должен передаваться как двоичный файл.
В данном сценарии: ==> binary
8. Отправьте запрос на передачу файла из удаленной системы в систему клиента.
В данном сценарии: ==> get example.jar
9. После выполнения команды завершите сеанс FTP.
В данном сценарии: ==> QUIT

Дальнейшие действия

Перейдите к следующему шагу. Вы можете автоматизировать передачу файлов с помощью пакетного задания FTP. <<

Сценарий: Защита FTP с помощью SSL

>> Предположим, что ваша фирма MyCo занимается сбором информации о новых компаниях и продает эту информацию фирмам, работающим в области планирования инвестиций. Одной из таких фирм является фирма TheirCo, которой нужно отправить отчеты с результатами исследований по FTP. Фирма MyCo всегда стремится обеспечить защиту данных, которые она поставляет своим заказчикам, независимо от того, каким способом передаются эти данные. В данном случае фирме MyCo требуется установить защищенный сеанс FTP с фирмой TheirCo.

Задачи

В данном сценарии требуется выполнить следующие задачи:

- Создать локальную сертификатную компанию на сервере iSeries фирмы MyCo
- Разрешить применение SSL на сервере FTP фирмы MyCo
- Экспортировать копию сертификата локальной CA в файл
- Создать хранилище сертификатов *SYSTEM на сервере фирмы TheirCo
- Импортировать сертификат локальной CA фирмы MyCo в хранилище сертификатов *SYSTEM фирмы TheirCo
- Назначить локальную CA фирмы MyCo уполномоченной CA на клиенте FTP фирмы TheirCo

Предварительные требования

MyCo

- Сервер iSeries, на котором установлена операционная система OS/400 версии V5R1 или выше.
- На сервере iSeries должна быть установлена программа TCP/IP Connectivity Utilities (5722-TC1) версии V5R1 или выше.
- На сервере iSeries должна быть установлена программа Cryptographic Access Provider 128-bit for iSeries (5722-AC3).
- На сервере iSeries должен быть установлен Диспетчер цифровых сертификатов IBM^R (DCM) (5722-SS1, компонент 34).
- На сервере iSeries должен быть установлен продукт IBM HTTP Server (5722-DG1).
- Для ограничения доступа к внешним приложениям должны быть созданы сертификаты пользователей (более подробная информация приведена в разделе Сценарий:Применение сертификатов для ограничения доступа к внешним приложениям и ресурсам.

TheirCo

- Сервер iSeries, на котором установлена операционная система OS/400 версии V5R2 или выше.
- На сервере iSeries должна быть установлена программа TCP/IP Connectivity Utilities (5722-TC1) версии V5R2.
- На сервере iSeries должна быть установлена программа Cryptographic Access Provider 128-bit for iSeries (5722-AC3).
- На сервере iSeries должен быть установлен Диспетчер цифровых сертификатов IBM (5722-SS1, компонент 34).
- На сервере iSeries должен быть установлен продукт IBM HTTP Server (5722-DG1).
- Для настройки сеансов FTP должен применять клиент FTP TCP/IP OS/400.

Описание

С помощью клиента FTP OS/400 фирма TheirCo отправляет запрос на защищенную передачу файлов серверу FTP фирмы MyCo. Обратитесь к разделу Защита клиента FTP с помощью TLS и SSL.

На сервере выполняется идентификация.

В защищенном сеансе FTP фирме TheirCo отправляются финансовые отчеты.

Задачи настройки

Для защиты сеанса FTP с помощью SSL на серверах фирм MyCo и TheirCo необходимо выполнить следующие задачи:

Задачи на сервере фирмы MyCo:

1. Создайте локальную сертификатную компанию на сервере iSeries фирмы MyCo (См. стр. 6)
2. Настройте SSL на сервере FTP фирмы MyCo (См. стр. 7)
3. Экспортируйте копию сертификата локальной CA фирмы MyCo в файл (См. стр. 7)

Задачи на сервере фирмы TheirCo:

1. Создайте хранилище сертификатов *SYSTEM на сервере фирмы TheirCo (См. стр. 8)
2. Импортируйте сертификат локальной CA фирмы MyCo в хранилище сертификатов *SYSTEM фирмы TheirCo (См. стр. 8)
3. Назначьте локальную CA фирмы MyCo в качестве уполномоченной CA на клиенте FTP фирмы TheirCo (См. стр. 9)



Сведения о настройке

Выполните описанные ниже задачи, для того чтобы настроить защиту FTP с помощью SSL.

Шаг 1: Создайте локальную сертификатную компанию (CA) на сервере iSeries фирмы MyCo

В данном сценарии предполагается, что Диспетчер цифровых сертификатов (DCM) ранее не применялся для создания сертификатов на сервере iSeries. Для реализации поставленных задач в фирме MyCo было принято решение создать локальную сертификатную компанию (CA) для выдачи сертификата серверу FTP. В качестве альтернативного варианта в фирме MyCo рассматривалась возможность выдать серверу FTP глобальный сертификат, предназначенный для установления соединений SSL.

При создании локальной CA с помощью Диспетчера цифровых сертификатов выполняются все необходимые действия по настройке SSL.

Для создания локальной CA на сервере фирмы MyCo с помощью Диспетчера цифровых сертификатов (DCM) выполните следующие действия:

1. Запустите DCM.
2. На панели навигации DCM выберите пункт **Создать сертификатную компанию (CA)**. Появится последовательность форм. С их помощью вы сможете пошагово выполнить процедуру создания локальной CA и другие задачи, необходимые для применения цифровых сертификатов при работе с SSL и подписями объектов.
3. Заполните все формы. Каждая форма соответствует определенной задаче, которую нужно выполнить для создания локальной CA на сервере iSeries и работы с ней. Требуется выполнить следующие задачи:
 - a. Выбрать способ хранения личного ключа, связанного с сертификатом локальной CA. Это действие требуется выполнять только в том случае, если в системе iSeries установлен Шифровальный сопроцессор IBM 4758-023 PCI. Если в системе нет шифровального сопроцессора, DCM автоматически сохранит сертификат и его личный ключ в хранилище сертификатов локальной CA.
 - b. Указать идентификационную информацию для локальной CA.
 - c. Установить сертификат локальной CA на PC или в браузере. Это позволит приложениям распознавать сертификаты, выданные локальной CA, как надежные.
 - d. Выбрать стратегию для локальной CA.
 - e. Создать с помощью локальной CA сертификат сервера или клиента, который может применяться приложениями для настройки соединений SSL. Если на сервере iSeries установлен Шифровальный сопроцессор IBM 4758-023 PCI, то при выполнении этой задачи вы сможете выбрать способ хранения личного ключа, связанного с сертификатом сервера или

клиента. Если в системе нет шифровального сопроцессора, DCM автоматически разместит сертификат и его личный ключ в хранилище сертификатов *SYSTEM. DCM создает хранилище сертификатов *SYSTEM при выполнении этой задачи.

- f. Выберите приложения, которым разрешено применять сертификат клиента или сервера для настройки соединений SSL. Примечание: Обязательно выберите ИД приложения, связанный с сервером FTP TCP/IP OS/400 (QIBM_QTMF_FTP_SERVER).
- g. С помощью локальной CA создайте сертификат подписи объекта, который может применяться приложениями для создания цифровых подписей объектов. При этом будет создано хранилище сертификатов *OBJECTSIGNING, предназначенное для работы с сертификатами подписи объекта. Примечание: Хотя в данном сценарии сертификат подписи объекта не применяется, выполните эту задачу. Если вы отмените выполнение процедуры на этом этапе, вам потребуется выполнить дополнительные действия для завершения настройки сертификата SSL.
- h. Выберите приложения, которые должны принимать сертификаты, выданные локальной CA. Примечание: Обязательно выберите ИД приложения, связанный с сервером FTP TCP/IP OS/400 (QIBM_QTMF_FTP_SERVER).

После заполнения всех форм можно настроить SSL на сервере FTP.

Шаг 2: Настройте SSL на сервере FTP фирмы MyCo

После назначения сертификата на сервере FTP фирмы MyCo требуется выполнить следующие действия для настройки SSL:

1. В окне программы Навигатор разверните **значок сервера iSeries** → **Сеть** → **Серверы** → **TCP/IP**.
2. Щелкните правой кнопкой мыши на значке **FTP**.
3. Выберите пункт **Свойства**.
4. Перейдите на страницу **Общие**.
5. Выберите следующий вариант применения SSL: **Только защищенные соединения**
Выберите эту опцию, чтобы на сервере FTP можно было устанавливать только сеансы SSL. Соединения могут устанавливаться и с незащищенным портом FTP, однако пользователю будет разрешено войти в систему только после того, как клиент FTP выполнит процедуру согласования SSL.

После выполнения этой задачи сервер FTP фирмы MyCo сможет устанавливать защищенные сеансы SSL и обеспечивать защиту данных, передаваемых в этих сеансах. Однако для настройки клиента FTP, с которым будет устанавливаться сеанс SSL, фирма MyCo должна предоставить клиенту, то есть фирме TheirCo, копию сертификата локальной CA. Для этого копию сертификата локальной CA нужно экспортировать в файл и отправить файл на сервер фирмы TheirCo. На сервере фирмы TheirCo можно воспользоваться DCM для размещения полученного сертификата в хранилище сертификатов *SYSTEM и настройки клиента FTP OS/400 на применение SSL.

Шаг 3: Экспортируйте копию сертификата локальной CA фирмы MyCo в файл

Фирма MyCo должна предоставить фирме TheirCo копию сертификата локальной CA. Для того чтобы с клиентом фирмы TheirCo можно было установить сеанс SSL, этот сертификат должен быть добавлен в число надежных сертификатов.

Для того чтобы экспортировать копию сертификата локальной CA в файл, на сервере фирмы MyCo нужно выполнить следующие действия:

1. Запустите DCM.
2. Нажмите кнопку **Выбрать хранилище сертификатов**.

3. Выберите в качестве хранилища сертификатов значение ***SYSTEM** и нажмите кнопку **Продолжить**.
4. На странице Хранилище сертификатов и пароль укажите пароль, заданный при создании хранилища сертификатов, и нажмите кнопку **Продолжить**.
5. После обновления содержимого панели навигации выберите задачу **Управление сертификатами**, а затем - **Экспортировать сертификат**.
6. Выберите пункт **Сертификатная компания (CA)** и нажмите кнопку **Продолжить**. Появится список сертификатов CA.
7. Выберите в списке сертификат локальной CA фирмы MyCo и нажмите кнопку **Экспортировать**.
8. Укажите, что сертификат нужно экспортировать в **Файл**, и нажмите кнопку **Продолжить**.
9. Укажите полное имя файла, в который нужно экспортировать сертификат локальной CA, и нажмите кнопку **Продолжить**. Сертификат будет экспортирован в файл.
10. Для того чтобы закрыть окно Подтверждение экспорта, нажмите кнопку **ОК**.

Теперь вы можете передать файл с сертификатом в конечные системы iSeries, в которых планируется проверять подписи, созданные с помощью этого сертификата. Этот файл можно отправить по электронной почте или по FTP. Обеспечивать защиту файла во время передачи не требуется.

Теперь на сервере фирмы TheirCo нужно импортировать сертификат локальной CA в хранилище сертификатов *SYSTEM и выбрать локальную CA фирмы MyCo в качестве уполномоченной сертификатной компании (а ее сертификат - в качестве надежного сертификата).

Шаг 4: Создайте хранилище сертификатов *SYSTEM на сервере фирмы TheirCo

Для того чтобы с клиентом FTP OS/400 фирмы TheirCo был установлен сеанс SSL, этот клиент должен распознать и принять сертификат, предоставленный сервером FTP фирмы MyCo во время настройки сеанса SSL. Для этого в хранилище сертификатов *SYSTEM должна находиться копия сертификата сертификатной компании (CA). Хранилище сертификатов *SYSTEM по умолчанию содержит копии сертификатов наиболее известных CA. Однако поскольку сертификат сервера FTP фирмы MyCo был выдан локальной CA, клиент FTP фирмы TheirCo должен получить копию сертификата локальной CA и импортировать ее в хранилище сертификатов *SYSTEM.

В данном сценарии предполагается, что Диспетчер цифровых сертификатов (DCM) ранее не применялся для создания сертификатов и работы с ними. Следовательно, в первую очередь на сервере фирмы TheirCo нужно создать хранилище сертификатов *SYSTEM. Для этого выполните следующие действия:

1. Запустите DCM.
2. На панели навигации Диспетчера цифровых сертификатов (DCM) выберите задачу **Создать хранилище сертификатов**. После этого выберите в качестве хранилища сертификатов значение ***SYSTEM** и нажмите кнопку **Продолжить**.
3. Выберите **Нет**, для того чтобы вместе с хранилищем сертификатов *SYSTEM был создан сертификат, и нажмите кнопку **Продолжить**.
4. Укажите пароль нового хранилища сертификатов и нажмите кнопку **Продолжить**. Появится окно с подтверждением.
5. Нажмите кнопку **ОК**.

Теперь на сервере фирмы TheirCo сертификат локальной CA можно импортировать в хранилище сертификатов и назначить в качестве надежного сертификата.

Шаг 5: Импортируйте сертификат локальной CA фирмы MyCo в хранилище сертификатов *SYSTEM фирмы TheirCo

Для того чтобы импортировать сертификат локальной CA в хранилище сертификатов *SYSTEM и назначить его в качестве надежного сертификата, выполните на сервере фирмы TheirCo следующие действия:

1. На панели навигации DCM щелкните на пункте **Выбрать хранилище сертификатов**, а затем выберите хранилище сертификатов ***SYSTEM**.
2. На странице Хранилище сертификатов и пароль укажите пароль, заданный при создании хранилища сертификатов, и нажмите кнопку **Продолжить**.
3. После обновления содержимого панели навигации выберите пункт **Управление сертификатами**. Появится список задач.
4. Выберите в списке задачу **Импортировать сертификат**.
5. Выберите в качестве типа сертификата значение **Сертификатная компания (CA)** и нажмите кнопку **Продолжить**.
6. Укажите полное имя файла, содержащего сертификат CA, и нажмите кнопку **Продолжить**. Появится сообщение о том, что сертификат был импортирован, либо сообщение об ошибке.

Теперь на сервере фирмы TheirCO можно указать, что клиент FTP считает сертификат локальной CA фирмы MyCo надежным. Это позволит ему устанавливать сеансы SSL с приложениями сервера, применяющими сертификат, выданный локальной CA фирмы MyCo.

Шаг 6: Назначьте локальную CA фирмы MyCo в качестве уполномоченной CA клиента FTP фирмы TheirCo

Для того чтобы клиент FTP фирмы TheirCo мог устанавливать защищенные соединения с сервером FTP фирмы MyCo, на сервере фирмы TheirCo необходимо выбрать CA, которые клиент будет считать надежными. Другими словами, сертификат локальной CA, ранее импортированный на сервер фирмы TheirCo, должен быть назначен надежным сертификатом.

Для того чтобы назначить сертификат локальной CA фирмы MyCo в качестве надежного сертификата для клиента FTP, выполните на сервере фирмы TheirCo следующие действия:

1. Запустите DCM.
2. Щелкните на пункте **Выбрать хранилище сертификатов** и выберите хранилище сертификатов *SYSTEM.
3. На странице Хранилище сертификатов и пароль укажите пароль, заданный при создании хранилища сертификатов, и нажмите кнопку **Продолжить**.
4. На панели навигации выберите пункт **Управление приложениями**. Появится список задач.
5. Выберите в списке задачу **Определить список надежных CA**.
6. Выберите значение **Клиент** в качестве типа приложения, для которого нужно определить список, и нажмите кнопку **Продолжить**.
7. Выберите в списке приложение клиента FTP TCP/IP OS/400 (QIBM_QTMF_FTP_CLIENT) и нажмите кнопку **Продолжить**. Появится список сертификатов CA.
8. Выберите ранее импортированный сертификат локальной CA фирмы MyCo и нажмите кнопку **OK**. Появится сообщение DCM, подтверждающее обновление списка надежных сертификатов.

После выполнения этой процедуры сервер FTP фирмы MyCo сможет устанавливать сеансы SSL с клиентом и сервером FTP фирмы TheirCo. Обратитесь к разделу Защита клиента FTP с помощью TLS и SSL.

Настройка сервера FTP

➤ Лицензионная программа TCP/IP Connectivity Utilities поставляется вместе с настроенными серверами FTP TCP/IP. Сервер FTP запускается вместе с TCP/IP.

Ниже перечислены указания по защите данных, которые необходимо соблюдать при настройке сервера FTP в сети Internet:

- Установите брандмауэр между сервером iSeries и сетью Internet.
- Не применяйте рабочий сервер iSeries в качестве сервера FTP.
- Не подключайте сервер FTP к корпоративным локальным или глобальным сетям.
- Ограничьте доступ к серверу FTP с помощью программ выхода.
- Ежемесячно проверяйте программы выхода FTP с целью убедиться, что они обеспечивают надежную защиту.
- Не предоставляйте анонимным пользователям FTP одновременно права на чтение и запись к одному каталогу. Работу анонимных пользователей с такими правами в Internet отследить невозможно.
- Ведите протокол всех обращений к серверу FTP iSeries и просматривайте протоколы ежедневно или еженедельно, чтобы обнаружить возможные атаки.
- Ежемесячно проверяйте правильность зарегистрированных программ выхода для сервера FTP.
- Ознакомьтесь с информацией о защите сервера FTP iSeries, приведенной в разделе Защита FTP.

Приведенные ниже разделы содержат информацию о различных способах просмотра и настройки серверов FTP:

Сервер FTP в Навигаторе

Этот раздел содержит информацию о настройке и управлении сервером FTP с помощью Навигатора.

Настройка серверов FTP для графических клиентов FTP и средств Web

Этот раздел содержит информацию о настройке на сервере FTP iSeries поддержки клиентов FTP с графическим интерфейсом, Web-браузеров и других средств Web.

Настройка анонимного доступа к FTP

Анонимный доступ к FTP позволяет удаленным пользователям работать с сервером FTP без указания ИД пользователя и пароля.



Сервер FTP в Навигаторе

Навигатор позволяет работать с конфигурацией сервера FTP. Для того чтобы с помощью Навигатора получить доступ к графическому пользовательскому интерфейсу для работы с FTP, выполните следующие действия:

1. В Навигаторе разверните **значок сервера iSeries** → **Сеть** → **Серверы** → **ТСР/IP**.
2. В правой панели щелкните правой кнопкой мыши на пункте **FTP** и выберите **Свойства**.
3. В появившемся окне диалога можно изменить свойства сервера FTP. Для просмотра справочной информации нажмите кнопку справки. Для получения справки по отдельному полю нажмите кнопку с вопросительным знаком, а затем щелкните мышью на интересующем вас поле.

Настройка серверов FTP для работы с клиентами FTP с графическим интерфейсом и средствами Web

» Сервер FTP iSeries поддерживает клиенты FTP с графическим интерфейсом, Web-браузеры и средства разработки Web-страниц.

В большинстве клиентов FTP с графическим интерфейсом для вывода списка файлов применяется формат UNIX^R, а имена файлов задаются в виде путей к файлам. Для работы с этими форматами необходимо выполнить следующие действия по настройке сервера FTP:

1. В Навигаторе разверните **значок сервера iSeries** → **Сеть** → **Серверы** → **TCP/IP**.
2. В правой панели щелкните правой кнопкой мыши на пункте **FTP** и выберите **Свойства**.
3. На странице **Свойства** выберите вкладку **Начальные форматы**.
 - Выберите в качестве формата присвоения имен файлам значение **Путь**.
 - Выберите в качестве формата вывода списка файлов формат **UNIX**.

Примечание:

Задавать значения параметров LISTFMT и NAMEFMT для конкретных сеансов FTP можно с помощью программы выхода, связанной с форматом TCPL0200 или форматом TCPL0300, для точки выхода процедуры входа в систему сервера FTP.

Формат вывода списка файлов можно изменить и *после* открытия сеанса FTP с помощью опций команды сервера FTP SITE (Отправить серверу информацию о клиенте). Эти параметры влияют на результаты, возвращаемые командами сервера FTP LIST (Список файлов) и NLST (Список имен).

Дополнительная информация приведена в следующих разделах:

- Вывод команды LIST в формате iSeries
- Вывод команды LIST в формате UNIX



Настройка анонимного доступа к FTP

Анонимный доступ к серверу FTP позволяет пользователям работать с некоторыми ресурсами удаленной системы, не указывая пароль. Набор общих данных определяется на удаленном сервере. Такие данные доступны любому пользователю. Ответственность за выбор общих ресурсов ложится на владельца ресурсов и администратора системы.

Для работы с общей информацией пользователь при входе в систему указывает в качестве ИД значение ANONYMOUS. Пользователю ANONYMOUS предоставляются права доступа к некоторым файлам и права на выполнение некоторых операций. Обычно анонимному пользователю разрешается выполнять следующие операции:

- Вход в систему сервера FTP
- Просмотр содержимого некоторых каталогов
- Получение файлов из этих каталогов.

Обычно анонимным пользователям запрещается передавать файлы на сервер FTP. Однако в некоторых системах существует специальный каталог, в который анонимные пользователи могут загружать файлы. Для анонимных пользователей часто создается учетная запись, в которой задается некоторая строка в качестве пароля. Обычно паролем служит слово 'guest' или адрес электронной почты. Некоторые серверы, содержащие архивы данных, явно запрашивают у пользователя адрес электронной почты и не поддерживают вход в систему с паролем "guest". Адрес электронной почты дает возможность администратору удаленного сервера получить некоторое представление о пользователе, работающем с сервером.

Анонимный доступ к серверу FTP в системе iSeries

Базовый сервер FTP не поддерживает анонимный доступ. Для настройки анонимного доступа к FTP в системе iSeries нужно создать программы для двух точек выхода сервера FTP: точки выхода из процедуры входа в систему сервера и точки выхода из процедуры проверки запросов.

Анонимный доступ к серверу FTP удобен, а во многих случаях просто необходим. Однако, разрешив анонимный доступ к серверу FTP, вы должны будете настроить защиту сервера iSeries. Дополнительная информация о защите системы приведена в разделе Защита сервера FTP с помощью программ выхода.

Для настройки анонимного доступа к FTP выполните следующие действия:

- 1. Подготовка к настройке анонимного доступа к FTP**
Ознакомьтесь с техническими требованиями и разработайте стратегию защиты.
- 2. Создание программ выхода для поддержки анонимного доступа к серверу FTP**
Создайте две программы выхода для поддержки анонимного доступа к FTP. В данном разделе приведены примеры программ выхода, а также полезные советы и рекомендации.
- 3. Создание пользовательского профайла OS/400 ANONYMOUS**
Создание такого профайла позволит запретить анонимным пользователям входить в систему iSeries *напрямую*.
- 4. Создание общей библиотеки или каталога**
Создайте, заполните и настройте общие библиотеки и каталоги.
- 5. Установка и регистрация программ выхода**
Создайте библиотеку, в которой будут расположены программы выхода и их файлы протокола, а затем зарегистрируйте программы выхода на сервере FTP.

Подготовка к настройке анонимного доступа к FTP

Следующая информация предназначена для настройки анонимного доступа к FTP.

Требования к пользователю

Для настройки анонимного доступа к FTP вы должны уметь:

- Вводить команды с несколькими параметрами и ключевыми словами в командной строке iSeries.
- Создавать в системе iSeries библиотеки, элементы и исходные физические файлы (у вас должны быть права доступа не ниже *SECOFR).
- Предоставлять права доступа к библиотекам, файлам, элементам и программам.
- Писать, редактировать, компилировать и тестировать программы на сервере iSeries.

Рекомендации по защите

Настройку анонимного доступа к FTP следует начать с определения стратегии анонимного доступа к FTP. Эта стратегия описывает способы защиты сервера FTP и способы применения программ выхода. Так как после настройки анонимного доступа любой пользователь сможет получить доступ к данным сервера FTP, необходимо тщательно продумать способы предоставления доступа и защиты данных.

Рекомендации по планированию стратегии защиты сервера FTP:

- Установите брандмауэр между сервером iSeries и сетью Internet.
- Не применяйте рабочую систему iSeries в качестве сервера FTP.
- Не подключайте сервер FTP к корпоративным локальным или глобальным сетям.
- Ограничьте доступ к серверу FTP с помощью программ выхода.
- Проверьте программы выхода FTP и убедитесь, что они обеспечивают надежную защиту.
- Не предоставляйте анонимным пользователям FTP одновременно права на чтение и запись к одному каталогу. Работу анонимных пользователей с такими правами в Internet отследить невозможно.
- Разрешите только анонимный доступ. Запретите доступ пользователей с другими ИД и проверку паролей.
- Разрешите пользователю ANONYMOUS доступ только к одной библиотеке или каталогу. (Где он будет располагаться? Как он будет называться?)

- В общем каталоге или библиотеке размещайте только те файлы, доступ к которым разрешен всем пользователям.
- Разрешить пользователям ANONYMOUS выполнять только команды просмотра и чтения (get, mget). **Никогда не разрешайте пользователям ANONYMOUS запускать команды CL.**
- Заносите в протокол все попытки входа в систему iSeries для работы с сервером FTP.
- Ежедневно или еженедельно просматривайте протоколы сервера FTP, чтобы вовремя обнаруживать попытки взлома или несанкционированного доступа.
- Ежемесячно проверяйте, что на сервере FTP зарегистрирована правильная программа выхода.
- Ежемесячно проверяйте надежность защиты сервера FTP.

Дальнейшие действия:

Создайте две программы выхода FTP для поддержки анонимного доступа к серверу FTP.

Создание программ выхода для поддержки анонимного доступа к серверу FTP

» Для настройки анонимного доступа к FTP нужно создать две программы выхода: программу выхода для процедуры входа в систему сервера FTP и программу выхода для процедуры проверки запросов к серверу FTP. Первая из этих программ присваивает пользователю ИД ANONYMOUS и изменяет текущий каталог на общую библиотеку или каталог. Программа выхода для процедуры проверки запросов к серверу FTP ограничивает набор команд, файлов, каталогов и библиотек, доступных пользователю ANONYMOUS.

Программы выхода и форматы точек выхода

Сервер FTP взаимодействует с программой выхода через особую точку выхода. Она применяется для передачи параметров от сервера программе выхода. Формат передаваемой информации зависит от формата точки выхода. Дополнительная информация о программах выхода для FTP приведена в разделе Программы выхода FTP. Для поддержки анонимного доступа к серверу FTP применяются следующие две точки выхода:

Программа	Точка выхода	Формат
Вход в систему	QIBM_QTMF_SVR_LOGON	TCPL0100, TCPL0200 или TCPL0300. ¹
Проверка запроса	QIBM_QTMF_SERVER_REQ	VLRQ0100

¹ - Существует несколько форматов точек выхода, однако программа выхода может быть зарегистрирована только для одного из них. Ознакомьтесь с информацией о различных форматах и выберите наиболее удобный из них.

Примеры программ

В данном руководстве приведено несколько примеров программ, которые помогут вам настроить анонимный доступ к серверу FTP. Вы можете использовать примеры в качестве основы для создания собственных программ. Кроме того, вы можете скопировать фрагменты кода примеров в свои программы. Примеры программ не рекомендуются запускать в рабочей системе.

Примечание: Эти программы приведены только в качестве примера. В них не заданы конкретные значения для некоторых параметров, поэтому эти программы нельзя применять в рабочей системе. Вы можете использовать их полностью или частично в качестве основы для создания своих собственных программ.

Примеры программ приведены для точки выхода входа в систему и точки выхода проверки запроса.

Дальнейшие действия:

Создание пользовательского профайла OS/400: ANONYMOUS



Создание пользовательского профайла OS/400: ANONYMOUS

Для того чтобы пользователи не могли напрямую подключаться к серверу iSeries с помощью пользовательского профайла ANONYMOUS, настоятельно рекомендуется создать пользовательский профайл ANONYMOUS с паролем *NONE. Этот профайл можно создать с помощью Навигатора.

1. В **Навигаторе** разверните список **Пользователи и группы**.
2. Щелкните правой кнопкой мыши на пункте **Все пользователи** и выберите **Новые пользователи**.
3. В панели Новые пользователи введите следующую информацию:
Имя пользователя = ANONYMOUS и
Пароль = No password.
4. Нажмите кнопку **Задания** и выберите вкладку **Общие**.
5. На странице **Общие** назначьте текущую библиотеку и домашний каталог для пользователя Anonutous.
6. Нажмите кнопку **ОК** и настройте остальные параметры.
7. Нажмите кнопку **Добавить**, чтобы создать профайл.

Дальнейшие действия:

Создание общей библиотеки или каталога

Создание общей библиотеки или каталога

После добавления анонимных пользователей необходимо создать для них общую библиотеку или каталог. Как правило, анонимным пользователям предоставляется доступ только к общим файлам. Рекомендуется предоставлять анонимным пользователям доступ только к одной библиотеке или к одному дереву каталогов, содержащему только “общие” файлы.

1. Создайте общие библиотеки и каталоги, в которых будут храниться файлы, доступные анонимным пользователям FTP.
2. Разместите в общих библиотеках и каталогах общие файлы.
3. Установите для общих библиотек или каталогов и их файлов права доступа PUBLIC *USE.

Дальнейшие действия:

Установка и регистрация программ выхода

Установка и регистрация программ выхода

Установка программы выхода

1. Создайте библиотеку, в которой будут храниться программы выхода и их файлы протокола.
2. Скомпилируйте программу выхода в этой библиотеке.
3. Установите для объектов библиотек, программ и файлов права доступа PUBLIC *EXCLUDE.
Для вызова программы выхода приложение сервера FTP при необходимости примет права доступа.

Регистрация программы выхода

1. В командной строке iSeries введите **WRKREGINF**.
2. Найдите запись для точки выхода из процедуры входа в систему сервера FTP:

```
QIBM_QTMF_SVR_LOGON TCPL0100
QIBM_QTMF_SVR_LOGON TCPL0200
QIBM_QTMF_SVR_LOGON TCPL0300
QIBM_QTMF_SERVER_REQ VLRQ0100
```

3. В поле Опц слева от записи точки выхода укажите **8** и нажмите **Enter**.
4. В меню Работа с программами выхода введите **1(Добавить)**.

5. В поле Программа выхода укажите имя программы выхода.
6. В поле Библиотека укажите имя библиотеки, в которой находится программа выхода.
7. Нажмите **Enter**.
8. Для того чтобы все экземпляры сервера FTP стали применять добавленную программу выхода, перезапустите сервер FTP.
9. Протестируйте программы выхода.

Примечание: Программы выхода будут применяться только в новых сеансах FTP. В уже открытых сеансах они применяться не будут.

Связанный раздел: Удаление установленных программ выхода

Защищенный FTP

Если ваша система iSeries играет роль сервера FTP в Internet, то она доступна из любой точки мира. Для того чтобы скрыть важные конфиденциальные сведения, хранящиеся на сервере FTP, необходимо принять меры по защите самого сервера iSeries. Кроме того, вы можете принять меры по защите клиента FTP.

Информация о различных способах защиты сервера и клиента FTP приведена в следующих разделах:

Предотвращение доступа к серверу FTP

Если FTP не используется, то вы должны отключить запуск FTP, чтобы сделать доступ к серверу iSeries через порт FTP невозможным. В этом разделе приведены инструкции по блокированию порта FTP.

Управление недоступом FTP

При работе с FTP необходимо контролировать доступ пользователей с целью защиты данных и сети. В этом разделе приведены советы и рекомендации по организации защиты.

Защита FTP с помощью Secure Sockets Layer (SSL)

Поддержка протокола SSL позволяет избежать отправки паролей и данных "открытым текстом" по соединению FTP. Для ее применения необходимо, чтобы и клиент, и сервер OS/400 поддерживали SSL.

Управление доступом с помощью программ выхода FTP

В этом разделе рассмотрена защита сервера iSeries с помощью точек выхода FTP.

Управление доступом с помощью Навигатора

Для защиты сервера или клиента FTP iSeries служит функция Ограничение доступа программы Администрирование приложений Навигатора.

Отслеживание пользователей FTP

Позволяет получать информацию обо всех пользователях, работающих с сервером FTP.

Настройка запрета на доступ к серверу FTP

➤ Если вы не хотите, чтобы клиенты подключались к серверу iSeries по FTP, запретите запуск сервера FTP. Для того чтобы запретить доступ к серверу FTP на сервере iSeries, выполните следующие действия:

Запретите автоматический запуск сервера FTP

Для того чтобы задания сервера FTP не запускались автоматически при запуске TCP/IP, выполните следующие действия:


1. В окне программы Навигатор разверните **значок сервера iSeries** → **Сеть** → **Серверы** → **TCP/IP**.
2. Щелкните правой кнопкой мыши на пункте **FTP** и выберите **Свойства**.
3. Отмените выбор опции **Запускать вместе с TCP/IP**.

Запретите доступ к портам FTP

Для того чтобы запретить запуск FTP, а также запретить пользователям связываться с портами FTP свои приложения, например, приложения с API сокетов, выполните следующие действия:

1. В окне программы Навигатор разверните **значок сервера iSeries** → **Сеть** → **Серверы** → **TCP/IP**.
2. Щелкните правой кнопкой мыши на пункте **Настройка TCP/IP** и выберите пункт **Свойства**.
3. В окне **Свойства конфигурации TCP/IP** перейдите на страницу **Запреты на порты**.
4. На странице **Запреты на порты** нажмите кнопку **Добавить**.
5. На странице **Добавить запрет на порт** укажите следующую информацию:
 - **Имя пользователя:** Введите имя защищенного пользовательского профайла сервера iSeries. (Защищенным называется пользовательский профайл, которому не принадлежат программы, принимающие права доступа, и пароль которого не сообщается другим пользователям.) Порт будет выделен указанному пользователю. При этом для остальных пользователей порт станет недоступным.
 - **Начальный порт:** 20
 - **Конечный порт:** 21
 - **Протокол:** TCP
6. Нажмите кнопку **ОК**, чтобы добавить запрет на порт.
7. На странице **Запреты на порты** нажмите кнопку **Добавить** и повторите описанную процедуру для протокола UDP.
8. Нажмите кнопку **ОК**, чтобы сохранить добавленные запреты на порты и закрыть окно **Свойства конфигурации TCP/IP**.
9. Запреты на порты вступят в силу после следующего запуска TCP/IP. Если во время изменения запретов на порты протокол TCP/IP был активен, перезапустите его.

Примечания:

- Запреты на порты вступят в силу после следующего запуска TCP/IP. Если во время изменения запретов на порты протокол TCP/IP был активен, перезапустите его.
- Информация о назначенных номерах портов приведена на Web-сайте организации IANA <http://www.iana.org> .
- Если порты 20 и 21 запрещено использовать всем пользовательским профайлам, за исключением QTCP, то при попытке запуска сервера FTP будет возникать ошибка.
- Такой способ может применяться в том случае, если нужно полностью запретить доступ к приложению, например, серверу FTP. Он не может применяться для ограничения доступа отдельных пользователей. При подключении пользователя к серверу FTP первоначально применяется профайл QTCP. Система переключается на профайл пользователя, когда соединение полностью установлено. Все пользователи сервера FTP применяют те права доступа к порту, которые предоставлены профайлу QTCP.



Ограничение доступа к FTP

» Если вы разрешите клиентам обращаться к системе по FTP, обратите внимание на следующее:

- Для применения FTP в системе необходимо более детально настроить права доступа к объектам. Например, если у пользователя есть права на просмотр файла (права доступа *USE), то он может


скопировать этот файл на PC или в другую систему. Возможно, вам потребуется защитить некоторые файлы от копирования в другие системы.

- С помощью программ выхода FTP можно ограничить набор операций FTP, которые разрешено выполнять пользователям. В частности, это можно сделать с помощью программы выхода из процедуры проверки запроса FTP. Например, вы можете запретить выполнять запросы GET, относящиеся к некоторым файлам базы данных.
- С помощью точки выхода из процедуры входа в систему сервера можно идентифицировать пользователя, подключившегося к серверу FTP. В разделе Настройка доступа анонимных пользователей к FTP описано, каким образом с помощью программ выхода можно разрешить анонимным пользователям работать с сервером FTP.
- Если не применяются протоколы TLS и SSL, пароли FTP передаются по соединению между клиентом и сервером в незашифрованном виде. При некоторых способах подключения такой пароль может быть перехвачен.
- Если системное значение QMAXSGNACN равно 1, системное значение QMAXSIGN относится к TELNET, но не к FTP. Если системное значение QMAXSGNACN равно 2 или 3 (значения, при которых пользовательский профайл отключается после достижения максимального числа попыток входа в систему), то при входе пользователя в систему сервера FTP подсчитывается число неудачных попыток. В этом случае злоумышленник может отключить пользовательский профайл, заданное число раз введя неверный пароль при подключении к серверу FTP.
- При каждой неудачной попытке входа в систему в протокол QHST записывается сообщение CPF2234. Вы можете создать программу, отслеживающую появление сообщений в протоколе QHST. При обнаружении нескольких неудачных попыток входа в систему такая программа может завершать работу сервера FTP.
- Задайте параметр Тайм-аут простоя (INACTTIMO) в конфигурации FTP, для того чтобы посторонние пользователи не могли воспользоваться открытым сеансом FTP на компьютере, оставленном без присмотра. Обязательно ознакомьтесь с информацией о применении параметра INACTTIMO и параметра Тайм-аут подключения, приведенной в документации или электронной справке.

Примечание: Системное значение QINACTITV не относится к сеансам FTP.

- Если применяется пакетное задание FTP, программа должна передавать на сервер как ИД пользователя, так и пароль. Эти значения могут быть заданы в программе, либо в файле, из которого они считываются программой. В обоих случаях пароль и ИД пользователя могут быть получены злоумышленниками. При работе с пакетным заданием FTP обязательно следует защищать ИД пользователя и пароль на уровне объектов. Кроме того, в этом случае рекомендуется применять ИД пользователя, которому предоставлены ограниченные права доступа в целевой системе. У такого пользователя должны быть права на выполнение только необходимых функций, например, на передачу файлов.
- FTP поддерживает функцию удаленного выполнения команд, которая также предусмотрена в APPC и программе iSeries Access. Команда сервера FTP RCMD (Удаленная команда) служит для эмуляции командной строки системы. Перед тем как разрешить использовать службу FTP, убедитесь, что применяется надежная схема защиты объектов. Для того чтобы ограничить доступ к команде RCMD или запретить ее использование, воспользуйтесь программой выхода FTP. Соответствующая точка выхода и примеры программ описаны в разделе Программы выхода FTP.
- С помощью FTP пользователь может работать с объектами интегрированной файловой системы. Перед запуском сервера FTP в системе убедитесь, что в интегрированной файловой системе применяется соответствующая схема прав доступа.
- Одним из популярных занятий хакеров является размещение информации на чужих сайтах. Иногда чужой сайт выбирается в качестве места хранения порнографии или ворованной информации. Хакер подключается к серверу iSeries по FTP и загружает на него такую информацию. После этого он сообщает адрес сервера FTP другим хакерам. Они подключаются к серверу iSeries по FTP и загружают эту информацию.

Защитить сервер от подобных атак можно с помощью программ выхода FTP. Например, все запросы на запись информации могут направляться в каталог, чтение данных из которого запрещено. Этого будет достаточно, чтобы разрушить план хакера, так как его друзья не смогут загрузить информацию из этого каталога. Дополнительная информация об атаках, основанных на загрузке информации в систему по FTP, и способах их предотвращения приведена в руководстве

AS/400^R Internet Security: Protecting Your AS/400 from HARM on the Internet .



Защита сервера FTP с помощью SSL

» Сервер FTP предоставляет расширенные средства защиты, применяемые при передаче файлов по незащищенной сети. С помощью Secure Sockets Layer (SSL) обеспечивается защита паролей и другой конфиденциальной информации во время обмена данными. Сервер FTP поддерживает защищенные сеансы SSL и TLS, обеспечивающие идентификацию клиента и автоматический вход в систему (дополнительная информация о протоколах TSL и SSL приведена в разделе SSL - основные принципы).

В большинстве приложений с поддержкой SSL “незащищенные” сеансы клиентов устанавливаются через один порт TCP, а защищенные - через другой. В FTP предусмотрены более гибкие средства защиты. Клиенту разрешено подключиться через незащищенный порт TCP (обычно это порт 21), а затем согласовать параметры идентификации и шифрования. Кроме того, клиент может подключиться через защищенный порт TCP (обычно это порт 990), который служит для установления соединений SSL. Сервер FTP системы iSeries позволяет выбрать один из этих способов подключения.

Перед настройкой SSL на сервере FTP установите в системе iSeries все необходимые программы и настройте цифровые сертификаты.

Для того чтобы настроить средства защиты SSL на сервере FTP, выполните следующие действия:

1. Создайте локальную сертификатную компанию с помощью DCM, либо настройте сервер FTP на применение глобального сертификата для установления соединений SSL.
2. Назначьте сертификат серверу FTP
3. Настройте обязательную идентификацию клиента на сервере FTP (при необходимости)
4. Разрешите применение SSL на сервере FTP

Дополнительная информация приведена в разделе Защита клиента FTP с помощью TLS/SSL. 

Создание локальной сертификатной компании

» Диспетчер цифровых сертификатов IBM (DCM) позволяет создать на сервере iSeries локальную сертификатную компанию (CA). С ее помощью вы сможете выдавать частные сертификаты приложениям сервера iSeries.

Для создания локальной CA с помощью DCM на сервере iSeries выполните следующие действия:

1. Запустите DCM.
2. На панели навигации DCM выберите пункт **Создать сертификатную компанию (CA)**. Появится последовательность форм. С их помощью вы сможете пошагово выполнить процедуру создания локальной CA и другие задачи, необходимые для применения цифровых сертификатов при работе с SSL и подписями объектов.
3. Заполните все формы. Каждая форма соответствует определенной задаче, которую нужно выполнить для создания локальной CA на сервере iSeries и работы с ней. При заполнении форм можно выполнить следующие действия:

- a. Выбрать способ хранения личного ключа, связанного с сертификатом локальной CA. Это действие требуется выполнять только в том случае, если в системе iSeries установлен Шифровальный сопроцессор IBM 4758-023 PCI. Если в системе нет шифровального сопроцессора, DCM автоматически сохранит сертификат и его личный ключ в хранилище сертификатов локальной CA.
- b. Указать идентификационную информацию для локальной CA.
- c. Установить сертификат локальной CA на PC или в браузере. Это позволит приложениям распознавать сертификаты, выданные локальной CA, как надежные.
- d. Выбрать стратегию для локальной CA.
- e. Создать с помощью локальной CA сертификат сервера или клиента, который может применяться приложениями для настройки соединений SSL. Если на сервере iSeries установлен Шифровальный сопроцессор IBM 4758-023 PCI, то при выполнении этой задачи вы сможете выбрать способ хранения личного ключа, связанного с сертификатом сервера или клиента. Если в системе нет шифровального сопроцессора, DCM автоматически разместит сертификат и его личный ключ в хранилище сертификатов *SYSTEM. DCM создает хранилище сертификатов *SYSTEM при выполнении этой задачи.
- f. Выберите приложения, которым разрешено применять сертификат клиента или сервера для настройки соединений SSL. Примечание: Обязательно выберите ИД приложения, связанный с сервером FTP OS/400 (QIBM_QTMF_FTP_SERVER).
- g. С помощью локальной CA создайте сертификат подписи объекта, который может применяться приложениями для создания цифровых подписей объектов. При этом будет создано хранилище сертификатов *OBJECTSIGNING, предназначенное для работы с сертификатами подписи объекта. Примечание: Хотя в данном сценарии сертификат подписи объекта не применяется, выполните эту задачу. Если вы отмените выполнение процедуры на этом этапе, вам потребуется выполнить дополнительные действия для завершения настройки сертификата SSL.
- h. Выберите приложения, которые должны принимать сертификаты, выданные локальной CA. Примечание: Обязательно выберите ИД приложения, связанный с сервером FTP OS/400 (QIBM_QTMF_FTP_SERVER).

Дополнительная информация о сертификатах приведена в следующих источниках:

Работа с сертификатами пользователей
Информация о применении DCM для получения сертификатов и их назначения пользовательским профайлам iSeries.

Применение API для выдачи сертификатов пользователям, не зарегистрированным в iSeries
Содержит информацию о том, как с помощью локальной CA можно выдать пользователю частный сертификат, не связывая этот сертификат с пользовательским профайлом iSeries.

Получение копии частного сертификата CA
В этом разделе приведена информация о том, как получить копию частного сертификата CA и установить его на PC. Такая копия необходима для идентификации сертификатов серверов, выданных CA.

Дальнейшие действия:

Выбор сертификата для сервера FTP



Выбор сертификата для сервера FTP

» Выполните эту задачу, если вы не назначили сертификат серверу FTP при создании локальной сертификатной компании (CA), либо планируете получать сертификаты от глобальной CA.

1. Запустите Диспетчер цифровых сертификатов IBM. Если вам необходимо создать или получить сертификат, либо внести какие-либо другие изменения в систему обслуживания сертификатов, сделайте это сейчас. За информацией о работе с сертификатами обратитесь к разделу Работа с Диспетчером цифровых сертификатов.
2. Нажмите кнопку **Выбрать хранилище сертификатов**.
3. Выберите ***SYSTEM**. Нажмите кнопку **Продолжить**.
4. Введите пароль для доступа к хранилищу сертификатов *SYSTEM. Нажмите кнопку **Продолжить**.
5. После обновления левой панели навигации разверните значок **Управление приложениями**.
6. Выберите **Назначить сертификат**.
7. В следующем окне выберите приложение **Сервер**. Нажмите кнопку **Продолжить**.
8. Выберите **Сервер FTP TCP/IP OS/400**.
9. Нажмите кнопку **Назначить сертификат**, чтобы назначить сертификат серверу FTP.
10. Выберите в списке сертификат, который нужно назначить серверу.
11. Выберите **Назначить новый сертификат**.
12. На странице **Назначить сертификат** появится подтверждающее сообщение DCM. После выбора всех сертификатов для сервера FTP нажмите кнопку **Готово**.

Дальнейшие действия:

Настройте обязательную идентификацию клиента на сервере FTP (при необходимости)
или
Разрешите применение SSL на сервере FTP



Настройка обязательной идентификации клиента на сервере FTP (выполняется при необходимости)

» Для того чтобы сервер FTP выполнял идентификацию клиентов, измените спецификацию приложения в Диспетчере цифровых сертификатов IBM.

Примечание: Сервер FTP поддерживает идентификацию клиентов, тогда как клиент FTP OS/400 не поддерживает эту функцию. Несмотря на это, вы можете включить функцию идентификации клиентов, однако при этом клиент FTP OS/400 не сможет работать с соединениями SSL.

Даже если на сервере FTP настроена функция идентификации клиентов, при подключении клиент все равно должен отправлять команду USER. После получения информации, переданной в команде USER, сервер FTP проверит, что имя пользователя совпадает с именем профайла, связанного с сертификатом, который был передан клиентом во время процедуры согласования SSL. Если имя пользователя соответствует сертификату клиента, сервер FTP разрешит пользователю войти в систему, не запрашивая пароль. Команду USER требуется отправлять для того, чтобы клиента можно было «уведомить» о входе в систему. Это единственный способ уведомления клиента, предусмотренный в FTP.

1. Запустите Диспетчер цифровых сертификатов IBM. Если вам необходимо создать или получить сертификат, либо внести какие-либо другие изменения в систему обслуживания сертификатов, сделайте это сейчас. За информацией о работе с сертификатами обратитесь к разделу Работа с Диспетчером цифровых сертификатов.
2. Нажмите кнопку **Выбрать хранилище сертификатов**.
3. Выберите ***SYSTEM**. Нажмите кнопку **Продолжить**.
4. Введите пароль для доступа к хранилищу сертификатов *SYSTEM. Нажмите кнопку **Продолжить**.
5. После обновления левой панели навигации разверните значок **Управление приложениями**.

6. Выберите **Обновить определение приложения**.
7. В следующем окне выберите приложение **Сервер**. Нажмите кнопку **Продолжить**.
8. Выберите **Сервер FTP TCP/IP OS/400**.
9. Выберите **Обновить определение приложения**.
10. В появившейся таблице включите опцию обязательной идентификации клиентов, указав значение **Да**.
11. Нажмите кнопку **Применить**.
12. На странице **Обновить определение приложения** появится подтверждающее сообщение DCM. После завершения обновления определения приложения сервера FTP нажмите кнопку **Готово**.

Дальнейшие действия:

Разрешите применение SSL на сервере FTP



Настройка SSL на сервере FTP

» Для того чтобы разрешить применения SSL на сервере FTP, выполните следующие действия:

1. В окне программы Навигатор разверните **значок сервера iSeries** → **Сеть** → **Серверы** → **TCP/IP**.
2. Щелкните правой кнопкой мыши на значке **FTP**.
3. Выберите пункт **Свойства**.
4. Перейдите на страницу **Общие**.
5. Выберите одну из следующих опций поддержки SSL:
 - **Только защищенные соединения**
Выберите эту опцию, чтобы на сервере FTP можно было устанавливать только сеансы SSL. Соединения могут устанавливаться и с незащищенным портом FTP, однако пользователю будет разрешено войти в систему только после того, как клиент FTP выполнит процедуру согласования SSL.
 - **Только незащищенные соединения**
Эта опция позволяет запретить открывать защищенные сеансы на сервере FTP. Все запросы на подключение к серверу через порт SSL будут отклоняться.
 - **Любые соединения**
Разрешает открывать как защищенные, так и незащищенные сеансы работы с сервером FTP.

Примечание:

Сервер FTP перезапускать не нужно. Он динамически определит, что ему был назначен сертификат. Если этого не произойдет, убедитесь, что на сервере iSeries установлена последняя версия PTF.



Защита клиента FTP с помощью TLS или SSL

» Для того чтобы данные передавались в зашифрованном виде по информационным и управляющим соединениям FTP, настройте соединения TLS или SSL. Шифрование данных, передаваемых по управляющим соединениям, позволяет скрыть пароль, применяемый для входа в систему сервера FTP.

Для того чтобы клиент смог устанавливать защищенные соединения с сервером FTP, настройте на этом клиенте уполномоченные сертификатные компании с помощью DCM. В их число должны входить все сертификатные компании, которыми были выданы сертификаты серверам. В

зависимости от типа применяемой сертификатной компании (CA), вам может потребоваться экспортировать и импортировать сертификаты CA. Дополнительная информация об уполномоченных сертификатных компаниях приведена в разделе Выбор уполномоченных CA для приложения в главе, посвященной DCM.

Если при работе с управляющим соединением применяются средства шифрования TLS или SSL, то клиент FTP по умолчанию будет шифровать данные, передаваемые по информационным соединениям. Протокол FTP не позволяет настроить защиту только для информационных соединений, оставив управляющие соединения незащищенными.

Применение средств шифрования значительно снижает производительность, поэтому их можно не настраивать для информационных соединений. Такой вариант защиты системы подходит для передачи файлов, не содержащих конфиденциальной информации. Защита будет обеспечиваться путем шифрования паролей.

Клиент FTP позволяет задавать параметры команды CL STRTCPFTP и команд, применяемых для поддержки TLS и SSL (SECOpen и SECData).

Применение средств защиты TLS и SSL на клиенте FTP iSeries

Управляющее соединение

Опцию применения средств защиты TLS и SSL можно указать в командах STRTCPFTP и SECOPEN.

Для того чтобы установить защищенное управляющее соединение, укажите в параметре SECCNN команды STRTCPFTP (FTP) значение *SSL. Кроме того, можно указать значение *IMPLICIT, для того чтобы защищенное соединение было установлено через предопределенный порт сервера. (Более подробная информация приведена в описании неявного соединения SSL.)

В сеансе клиента FTP защищенное управляющее соединение можно установить с помощью команды SECOPEN.

Информационное соединение

Для того чтобы установить защищенное информационное соединение, укажите в параметре DTAPROT команды STRTCPFTP (FTP) значение *PRIVATE. Для того чтобы данные передавались по информационному соединению в незашифрованном виде, укажите в параметре DTAPROT значение *CLEAR.

Если было установлено защищенное управляющее соединение, то вы можете изменить уровень защиты информационного соединения с помощью команды SECData.

Неявное соединение SSL

Некоторые серверы FTP поддерживают неявные соединения SSL. Такие соединения предоставляют те же средства шифрования, что и опция *SSL, однако они устанавливаются через определенный порт сервера (обычно это порт 990), настроенный для приема запросов на установление соединений TLS или SSL.

Такой способ позволяет устанавливать защищенные соединения в тех реализациях FTP, которые не поддерживают стандартный протокол работы с соединениями TLS и SSL.

Неявные соединения поддерживаются в большинстве старых реализаций SSL, однако в настоящее время их применять не рекомендуется.

Примечание:

В стандартном протоколе настройки соединения TLS или SSL при подключении к серверу требуется указать команду сервера AUTH (Идентификация). Уровень защиты данных задается командами сервера PBSZ и PROT.

Однако при установлении неявного соединения SSL команды AUTH, PBSZ и PROT **не** передаются на сервер. При этом сервер будет работать так, как будто клиент передал ему эти команды со следующими параметрами:

- AUTH SSL
- PBSZ 0
- PROT P



Управление доступом с помощью программ выхода FTP

Защита сервера FTP основана на защите объектов OS/400. Это означает, что для подключения к серверу FTP системы iSeries удаленный пользователь должен обязательно указать имя своего пользовательского профайла и пароль.

Вы можете усилить защиту, создав программы выхода FTP для точек выхода клиента и сервера FTP, которые будут дополнительно ограничивать доступ к серверу FTP. Например, вы можете ограничить как вход в систему сервера FTP, так и доступ к библиотекам, объектам и командам.

Для того чтобы ограничить набор команд CL и FTP, которые будет разрешено выполнять пользователям, создайте программу выхода для процедуры проверки запросов сервера FTP. Инструкции и примеры приведены в разделе Точка выхода проверки запроса:клиент и сервер.

Программа выхода для точки выхода из процедуры входа в систему может применяться для идентификации пользователей на сервере приложений TCP/IP.

Вы можете создать программу выхода для проверки запросов к клиенту FTP, связав ее с точкой выхода, рассмотренной в разделе Точка выхода клиента:проверка запроса. Такая программа может ограничивать набор функций клиента FTP, с которыми разрешено работать пользователю.

Вместо применения программ выхода для процедур проверки запросов сервера и клиента FTP вы можете ограничить доступ к командам FTP с помощью Средств ограничения доступа Администрирования приложений.

Для обеспечения правильной работы программ выхода необходимо Установить и зарегистрировать программы точек выхода. Если программы стали не нужны, то вы всегда можете Удалить программы точек выхода.

Управление доступом с помощью Навигатора

Доступ пользователей к функциям сервера и клиента FTP можно ограничить с помощью Навигатора. Функция Администрирование приложений позволяет настроить права доступа к функциям для отдельных пользователей или групп пользователей. Кроме того, для управления доступом к функциям FTP могут применяться программы выхода FTP, связанные с точками выхода процедур проверки запросов FTP.

Для того чтобы настроить права доступа пользователей к функциям FTP с помощью Навигатора, выполните следующие действия:

1. В окне Навигатора щелкните правой кнопкой мыши на значке системы **сервер iSeries** и выберите пункт **Администрирование приложений**.
2. Выберите вкладку **Приложения хоста**.

3. Разверните значок **Утилиты TCP/IP iSeries**.
4. Разверните значок **Протокол передачи файлов (FTP)**.
5. Разверните значок **Клиент FTP** или **Сервер FTP**.
6. Выберите функцию, доступ к которой вы хотите ограничить.
7. Выберите **Настроить**.
8. В окне диалога **Настроить права доступа** измените список пользователей и групп, которым разрешен или запрещен доступ к функции.
9. Нажмите **ОК** для сохранения изменений, внесенных на странице **Настроить доступ**.
10. Нажмите **ОК** для завершения работы со страницей **Администрирование приложений**.

Для настройки прав доступа отдельного пользователя или группы к зарегистрированным функциям FTP можно воспользоваться функцией Пользователи и группы Навигатора. Для этого выполните следующие действия:

1. В Навигаторе откройте **сервер iSeries** → **Пользователи и группы**.
2. Выберите **Все пользователи** или **Группы**.
3. Щелкните правой кнопкой мыши на имени пользователя или группы и выберите пункт **Свойства**.
4. Нажмите кнопку **Возможности**.
5. Выберите **Приложения**.

В появившемся окне диалога можно изменить права доступа пользователя или группы к указанным функциям. Для изменения прав доступа к группе зависимых функций достаточно изменить права доступа к "родительской" (т.е. расположенной выше в иерархическом списке) функции.

Дополнительная информация о защите сервера FTP iSeries приведена в разделе Защита FTP.

Отслеживание пользователей FTP

Ведение и просмотр протокола FTP позволяют контролировать работу системы и обнаруживать атаки извне. Для отслеживания пользователей, подключающихся к серверу FTP, выполните следующие действия:

1. В Навигаторе разверните **значок сервера** → **Сеть** → **Серверы** → **TCP/IP**.
2. В правой панели щелкните правой кнопкой мыши на **FTP** и выберите пункт **Задания сервера**.
3. Будет показана панель задания сервера FTP. В столбце Текущий пользователь будет указан пользователь, подключившийся к этому заданию сервера. Если подключившихся пользователей нет, то будет указано значение Qtcpr. Нажмите клавишу **F5** или выберите **Вид** → **Обновить**, чтобы обновить информацию в этом меню.

Имена этих заданий указываются в формате *QTFTPnnnnn*, где *nnnnn* - произвольно выбранное число.

Для запуска заданий сервера FTP выполните действия, описанные в разделе Запуск сервера FTP.

Управление сервером FTP

Протокол передачи файлов (FTP) позволяет пользователям системы iSeries обмениваться файлами по сети и работать с общими файлами. FTP состоит из двух компонентов - клиента и сервера. Вы работаете с клиентом FTP. Клиент FTP взаимодействует с сервером FTP. Как правило, пользователь не работает напрямую с сервером FTP. Информация об управлении сервером FTP приведена в следующих разделах:

- Запуск и завершение работы сервера FTP
- Выбор числа доступных серверов FTP
- Повышение производительности сервера FTP за счет настройки подсистемы

Запуск и остановка сервера FTP

Сервер FTP можно запустить и остановить с помощью Навигатора. Инструкции по работе с FTP приведены в разделе Работа с FTP с помощью Навигатора.

Для запуска сервера FTP выполните следующие действия:

1. В Навигаторе разверните **значок сервера iSeries** → **Сеть** → **Серверы** → **TCP/IP**.
2. В правой панели щелкните правой кнопкой мыши на пункте **FTP** и выберите **Запустить**.

Для того чтобы завершить работу сервера FTP, выполните следующие действия:

1. В Навигаторе разверните **значок сервера iSeries** → **Сеть** → **Серверы** → **TCP/IP**.
2. В правой панели щелкните правой кнопкой мыши на пункте **FTP** и выберите **Остановить**.

Настройка числа доступных серверов FTP

» Вы можете задать минимальное число серверов, доступных для подключающихся клиентов. Для этого перейдите на страницу **Свойства FTP** и укажите число от 1 до 20 в поле **Начальное число запускаемых серверов**. Значение 1 замедляет подключение к серверу FTP. Рекомендуемое значение - 3.

При подключении клиента к серверу FTP iSeries сервер сравнивает число работающих серверов, не подключенных к клиентам, со значением параметра Начальное число запускаемых серверов. В случае, если значение параметра превышает число доступных серверов, запускаются дополнительные серверы. В противном случае, никакие действия не выполняются. Измененное начальное значение сервера вступает в силу при следующем подключении клиента, когда происходит описанная выше проверка.

Например, если установлено пять сеансов клиентов FTP и задано начальное значение серверов, равное 10, то всего будет запущено 15 серверов FTP. Из них 5 серверов будут выделены активным сеансам клиентов, а 10 останутся незанятыми. Число доступных серверов может превышать значение параметра начальных серверов. В этом же примере, в случае, если пять сеансов клиентов будут закрыты и не будут запущены другие сеансы, то останется 15 доступных серверов. «

Повышение производительности сервера FTP за счет настройки подсистемы

Подсистема QSYS/QSYSWRK по умолчанию применяется для выполнения многих заданий серверов, поставляемых фирмой IBM. Если вы выберете другую подсистему для сервера FTP, ему не нужно будет разделять ресурсы с другими заданиями, поэтому его производительность может повыситься.

Для выбора подсистемы сервера FTP выполните следующие действия:

1. В окне программы Навигатор разверните **значок сервера iSeries** → **Сеть** → **Серверы** → **TCP/IP**.
2. Щелкните правой кнопкой мыши на пункте **FTP** и выберите **Свойства**.
3. На странице **Свойства FTP** выберите опцию **Описание подсистемы**.
4. Укажите описание подсистемы и имя библиотеки.

Если указанная в команде подсистема не существует, то сервер FTP создает эту подсистему, необходимые записи таблицы маршрутизации и описания заданий. С помощью предварительного задания можно настроить параметры новой подсистемы, а затем запустить пакетные задания сервера в этой подсистеме.

Работа с клиентом FTP в системе iSeries

Клиент FTP позволяет передавать файлы из файловых систем root, QSYS.Lib, QOpenSys, QOPT и QFileSvr.400 сервера iSeries в другие системы. Кроме того, он позволяет передавать папки и документы из файловой системы библиотек документов (QDLS). С клиентом FTP можно работать в интерактивном или пакетном режиме. В последнем случае команды считываются из файла, а ответы сервера записываются в другой файл. С его помощью можно выполнять и другие операции над файлами системы.

Клиент предоставляет пользовательский интерфейс для ввода команд, передаваемых на сервер FTP. Результаты выполнения этих команд выдаются пользователю.

Для передачи файлов между клиентом и сервером устанавливаются два соединения. Управляющее соединение применяется для передачи команд на сервер FTP. Сервер отправляет по этому соединению ответы с результатом выполнения команды. По информационному соединению передаются списки и содержимое файлов.

В клиенте и сервере предусмотрены функции передачи данных, взаимодействующие с операционными системами. Эти функции позволяют считывать и записывать данные в локальных файловых системах и передавать их по информационному соединению.

Открытие и закрытие сеанса клиента

Информация о том, каким образом можно открыть и закрыть сеанс сервера.

Тайм-аут сервера

Указания о том, каким образом можно избежать завершения соединения по тайм-ауту.

Передача файлов с помощью FTP

Описание процедур отправки и приема файлов с помощью FTP.

Запуск задания FTP в пакетном режиме

Примеры запуска FTP в неконтролируемом режиме.

Открытие и закрытие сеанса клиента

В этом разделе подробно описана работа с клиентом FTP на сервере iSeries.

Открытие сеанса клиента FTP

Закрытие сеанса клиента FTP

Открытие сеанса клиента FTP

Для запуска функции клиента FTP необходимо знать следующее:

- Имя или IP-адрес системы, с которой устанавливается соединение для отправки или приема файлов.
- ИД и пароль (если они необходимы) для входа в эту удаленную систему.
- Имена файлов, которые вы собираетесь отправлять или принимать.

Для запуска клиента на локальном сервере iSeries служит команда Запустить протокол передачи файлов TCP/IP (STRTCPFTP “удаленная-система”), которая устанавливает соединение с сервером FTP указанной удаленной системы. Например, команда FTP myserver.com запускает сеанс клиента на локальном сервере iSeries и устанавливает соединение с сервером FTP удаленной системы myserver.com. Для ввода дополнительных параметров необходимо указать их в командной строке или ввести команду STRTCPFTP без указания удаленной системы.

```

                                Запустить протокол передачи файлов (FTP) TCP/IP
Введите варианты, нажмите Enter.
Удаленная система . . . . . > MYSERVER.COM

Идентификатор набора символов      *DFT          1-65533, *DFT
Порт . . . . . > *SECURE          1-65535, *DFT, *SECURE
Защищенное соединение. . . . . *DFT          *DFT, *NONE, *SSL, *IMPLICIT
Защита данных . . . . . *DFT          *DFT, *CLEAR, *PRIVATE

```

После того, как вы укажете имя удаленной системы, появится приглашение ввести дополнительную информацию. Ниже перечислены все возможные варианты (подробности см. в справке по соответствующему полю):

Удаленная система (RMTSYS)

Имя удаленной системы, с которой устанавливается соединение. Возможны следующие значения:

***INTNETADR**

Параметр IP-адрес (INTNETADR). IP-адрес указывается в формате nnn.nnn.nnn.nnn, где nnn - десятичное число от 0 до 255.

удаленная-система

Укажите имя удаленной системы, с которой устанавливается соединение.

Идентификатор набора символов (CCSID)

Укажите идентификатор набора символов ASCII (CCSID), применяемый при передаче файлов ASCII SBCS (набор однобайтовых символов). Этот параметр используется только в том случае, когда в поле FTP TYPE установлено значение ASCII. Возможны следующие значения:

***DFT**

Применяется значение CCSID 00819 (ISO 8859-1 8-битовый ASCII).

значение-CCSID

Применяется запрошенное значение CCSID. Предварительно проверяется его допустимость.

Порт (PORT)

Номер порта для соединения с сервером FTP. Обычно для соединения с сервером FTP используется “стандартный” порт 21. Однако в некоторых случаях соединение устанавливается через другой порт. В таком случае в этом поле нужно указать номер используемого порта. Возможны следующие значения:

***DFT**

Применяется значение 00021.

***SECURE**

Применяется значение 00990. Порт 990 применяется для защищенных серверов FTP, которые шифруют данные с помощью протокола Transport Layer Security (TLS) или Secure Sockets Layer (SSL).

порт

Применяется указанный порт. Предварительно проверяется его допустимость.

Примечание: Для клиента FTP значения порта 990 и *SECURE эквивалентны.

» Защищенное соединение (SECCNN)

В этом поле указывается тип применяемого механизма защиты информации, передаваемой по соединению FTP (к таким механизмам относится также пароль, по которому сеанс идентифицируется на сервере FTP). Совместимые протоколы TLS и SSL шифруют передаваемые данные, защищая их от просмотра и обеспечивая их целостность.

Примечание: Для открытия защищенного соединения FTP в сеансе клиента FTP служит команда FTP SECOPEN.

Возможны следующие значения:

***DFT**

Если в поле PORT указано значение *SECURE или 990, то применяется значение *IMPLICIT; в противном случае - *NONE.

***IMPLICIT**

Клиент FTP автоматически пытается применить TLS/SSL при подключении к указанному серверу FTP (не отправляя на сервер команду AUTH). Соединение закрывается, если сервер не поддерживает неявное использование TLS/SSL на указанном порте или по каким-либо причинам возникает сбой согласования TLS/SSL.

***SSL**

После подключения к указанному серверу FTP клиент FTP пытается установить защищенное соединение TLS/SSL, отправляя команду AUTH. Если сервер поддерживает TLS/SSL, выполняется процедура согласования TLS/SSL. Соединение закрывается, если сервер не поддерживает использование TLS/SSL на указанном порте или возникает сбой согласования TLS/SSL.

***NONE**

Клиент FTP не применяет шифрование в управляющем соединении с сервером FTP. «

» Защита данных (DTAPROT)

Тип защиты информации, передаваемой по соединению обмена данными FTP. По этому соединению передаются файлы и списки каталогов. Защита соединения обмена данными FTP невозможна, если управляющее соединение не защищено.

Примечание: Для последовательного изменения уровня защиты данных служит команда FTP SECData. С помощью команды сервера FTP PROT клиент запрашивает указанный уровень защиты данных после установки защищенного управляющего соединения.

Возможны следующие значения:

***DFT**

Если в параметре SECCNN указано защищенное управляющее соединение, используется значение *PRIVATE; в противном случае - *CLEAR.

***PRIVATE**

Информация, передаваемая по соединениям обмена данными FTP, шифруется. Если в параметре SECCNN указано, что управляющее соединение FTP не защищено, то значение *PRIVATE недопустимо.

***CLEAR** Информация, передаваемая по соединениям обмена данными FTP, не шифруется. <<

Таблица преобразования EBCDIC/ASCII (TBLFTPOUT)

Таблица, используемая для преобразования всех исходящих данных клиента FTP. Исходящие данные преобразуются из формата EBCDIC в ASCII. Если значение в этом поле не указано, то тип преобразования определяется по параметру CCSID. Возможны следующие значения:

***CCSID**

Тип преобразования исходящих данных определяется по параметру CCSID.

***DFT**

Тип преобразования исходящих данных определяется по параметру CCSID.

Имя таблицы преобразования исходящих данных может быть уточнено одним из следующих значений библиотеки:

***LIBL**

Поиск будет выполняться во всех библиотеках из пользовательской и системной частей списка библиотек до обнаружения первого совпадения.

***CURLIB**

Поиск будет выполняться в текущей библиотеке задания, а если она не указана, то в библиотеке QGPL.

имя-библиотеки

Укажите имя библиотеки, в которой следует выполнять поиск.

таблица-преобразования-исходящих-данных

Укажите таблицу для преобразования исходящих данных клиента FTP.

Таблица преобразования ASCII/EBCDIC (TBLFTPIN)

Таблица, используемая для преобразования всех входящих данных клиента FTP. Входящие данные преобразуются из формата ASCII в EBCDIC. Если значение в этом поле не указано, то тип преобразования определяется по параметру CCSID. Возможны следующие значения:

***CCSID**

Тип преобразования входящих данных определяется по параметру CCSID.

***DFT**

Тип преобразования входящих данных определяется по параметру CCSID.

Имя таблицы преобразования входящих данных может быть уточнено одним из следующих значений библиотеки:

***LIBL**

Поиск будет выполняться во всех библиотеках из пользовательской и системной частей списка библиотек до обнаружения первого совпадения.

***CURLIB**

Поиск будет выполняться в текущей библиотеке задания, а если она не указана, то в библиотеке QGPL.

имя-библиотеки

Укажите имя библиотеки, в которой следует выполнять поиск.

таблица-преобразования-входящих-данных

Укажите таблицу для преобразования входящих данных клиента FTP.

Процедура передачи файлов между системами описана в разделе Передача файлов по FTP.

Закрытие сеанса клиента FTP

Для закрытия сеанса FTP служит команда QUIT. Эта команда прерывает соединение с удаленным хостом и завершает сеанс FTP в локальной системе iSeries. Кроме того, для завершения сеанса вы можете нажать клавишу F3 (Выход) и подтвердить завершение сеанса.

Информация о тайм-ауте сервера

» Следует уделить особое внимание значению тайм-аута бездействия. Это время бездействия сервера FTP в секундах, по истечении которого сеанс закрывается. Некоторые удаленные серверы позволяют клиенту изменять это значение. Например, система iSeries поддерживает команду сервера FTP TIME, которую можно отправить на сервер с помощью команды клиента FTP QUOTE. См. раздел QUOTE (Отправить команду на сервер FTP). Серверы UNIX, как правило, поддерживают команду SITE IDLE.

При выполнении локальных команд iSeries с помощью команды SYSCMD или F21 обмен данными между клиентом и сервером отсутствует. По этой причине, если при выполнении этих локальных команд iSeries будет превышено значение тайм-аута бездействия сервера, то сервер закроет соединение. В случае прерывания соединения необходимо повторно войти в систему сервера с помощью команды OPEN (OPEN <имя-удаленной-системы>) и вызвать команду USER, как это описано в примечании к разделу Вход в удаленную систему (Сервер).



Передача файлов с помощью FTP

Для передачи файлов по соединению FTP выполните следующие действия.

1. Вам потребуется следующая информация:
 - Имя хоста или IP-адрес удаленного компьютера.
 - Имя и пароль для входа в удаленную систему (кроме тех случаев, когда удаленный компьютер поддерживает анонимный доступ к FTP)
 - Имя и расположение файла, который вы планируете получить
 - Каталог, в который нужно поместить файл
 - Тип передачи, который будет применяться: ASCII, EBCDIC или BINARY
 - Способ защиты соединения: TSL или SSL.
2. Введите в командной строке **FTP** и нажмите **Enter**.
3. В приглашении укажите имя хоста или IP-адрес удаленной системы и нажмите **Enter**. Можно указать либо имя, либо IP-адрес, например:
remote.systemname.com
или
110.25.9.13
4. Введите идентификатор кодированного набора символов (CCSID). Если вам не требуется какой-то конкретный CCSID, то укажите значение по умолчанию (*DFT).
5. » Если для передачи пароля и данных должно применяться защищенное соединение, укажите в параметре Порт значение *SECURE. «
6. Нажмите **Enter**. Клиент FTP сообщит об успешном подключении к удаленной системе.



Примечание:


Если был задан порт *SECURE, и сервер не поддерживает установление неявных соединений TLS и SSL через этот порт, либо во время согласования параметров TLS или SSL возникла ошибка, соединение будет закрыто.



7. Для того чтобы изменить тип передачи файлов, выполните следующие действия:
 - a. Для настройки режима EBCDIC введите **EBCDIC** и нажмите **Enter** до начала передачи файла.
 - b. Для настройки режиме BINARY введите **BINARY** и нажмите **Enter** до начала передачи файла.
 - c. Для возврата в режим ASCII введите **ASCII** и нажмите **Enter** до начала передачи файла.
8. Теперь вы можете передать файлы:
 - a. Введите команду **CD** и имя каталога. Нажмите **Enter**.
 - b. Выполните одно из следующих действий:
 - Для передачи файла из системы сервера в систему клиента введите команду **GET** и имя файла:
GET myfile.txt
 - Для передачи файла из системы клиента в систему сервера введите команду **PUT** и имя файла:
PUT myfile.txt
9. Для того чтобы завершить сеанс клиента FTP и вернуться к командной строке iSeries, введите команду FTP **QUIT**.

Запуск FTP в пакетном режиме

Клиент FTP можно запускать не только в интерактивном, но и в пакетном режиме. В этом разделе приведено два примера запуска FTP в пакетном режиме: простой (См. стр. 31) и сложный (См. стр. 32). Другой пример приведен в разделе 6.7 "Batch FTP" книги V4 TCP/IP for AS/400: More Cool Things

Than Ever  (около 744 страниц).

Запуск FTP в пакетном режиме: простой пример

Ниже приведен простой пример передачи файлов по FTP в пакетном режиме. Один файл передается из удаленной системы в локальную.

В этом примере задействованы следующие объекты:

- Программа на CL
- Входной файл с командами FTP
- Выходной файл с сообщениями FTP

Программа на CL

```
*****
ITSOLIB1/QCLSRC BATCHFTP:
-----
PGM
OVRDBF FILE(INPUT) TOFILE(ITSOLIB1/QCLSRC) MBR(FTPCMDS)
OVRDBF FILE(OUTPUT) TOFILE(ITSOLIB1/QCLSRC) MBR(OUT)
FTP RMTSYS(SYSxxx)
ENDPGM
*****
```

Примечание:

Для того чтобы этот пример работал в программах на языке ILE CL, необходимо добавить опцию OVRSCOPE(*CALLLVL) в команды OVRDBF.

Программа BATCHFTP присваивает параметру INPUT имя исходного физического файла ITSOLIB1/QCLSRC MBR(FTPCMDS). Вывод записывается в элемент MBR(OUT).

Входной командный файл

```
*****
ITSOLIB1/QCLSRC FTPCMDS:
-----
ITSO ITSO
CD ITSOLIB1
SYSCMD CHGCURLIB ITSOLIB2
GET QCLSRC.BATCHFTP QCLSRC.BATCHFTP (REPLACE
QUIT
*****
```

Необходимые команды FTP указаны в файле FTPCMDS.

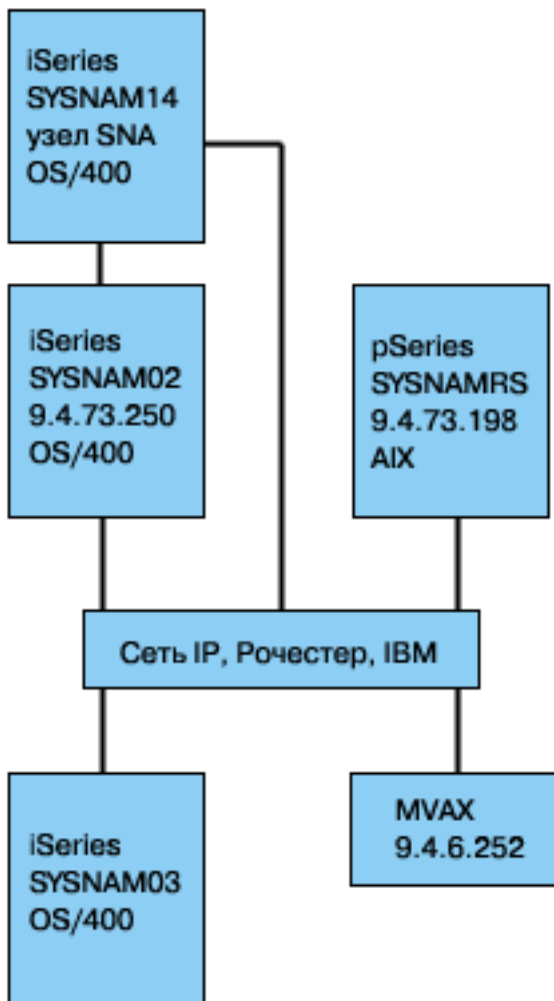
Выходной файл сообщений

```
*****
Вывод FTP перенаправлен в файл
Чтение команд из входного файла
Подключение к хосту SYSxxx
с адресом х.xxx.xx.xxx через порт 21.
220-QTCP на SYSxxx.sysnam123.ibm.com.
220 Соединение будет прервано при простое больше 5 минут.
Введите ИД пользователя (itso):
> ITSO ITSO
331 Введите пароль.
230 Пользователь ITSO вошел в систему.
Удаленная операционная система - OS/400. Версия TCP/IP - "V3R1M0".
250 Применяется формат имен "0".
257 Текущая библиотека "QGPL".
Введите команду FTP.
> CD ITSOLIB1
Введите команду FTP.
250 Текущая библиотека изменена на ITSOLIB1.
> SYSCMD CHGCURLIB ITSOLIB2
Введите команду FTP.
> GET QCLSRC.BATCHFTP QCLSRC.BATCHFTP (REPLACE
200 Команда PORT выполнена.
150 Получение элемента BATCHFTP из файла QCLSRC в библиотеке ITSOLIB1.
250 Передача файлов выполнена.
147 байт передано за 0,487 с. Скорость передачи - 0,302 Кб/с.
Введите команду FTP.
> QUIT
221 Получена команда QUIT.
*****
```

Это содержимое выходного файла. Самый простой подход - написать программу обработки этого файла, которая будет проверять наличие ошибок и при необходимости отправлять сообщение об ошибке в очередь QSYSOPR. Коды ошибок FTP начинаются с 4 и 5.

Запуск FTP в пакетном режиме: сложный пример

В следующем примере несколько файлов копируются с различных удаленных хостов в центральную систему iSeries в пакетном режиме:



*

Пользователю GWIL системы iSeries SYSNAM03 требуется:

1. Получить файлы с хостов SYSNAMRS (RS/6000^R) и MVAX (VAX).
2. Передать файл, полученный из системы SYSNAMRS, в систему SYSNAM02 (другая система iSeries) по FTP.
3. Передать этот файл в систему iSeries с именем SYSNAM14 по протоколу TCP/IP.

Создайте программу на CL для запуска FTP

1. Как было показано в предыдущем примере, в FTP по умолчанию для ввода команд и вывода сообщений применяется дисплейная станция. Для запуска FTP в пакетном режиме нужно создать для этих целей специальные файлы. С помощью команды OVRDBF можно переопределить файлы ввода и вывода для пакетного режима:

```

OVRDBF FILE(INPUT) TOFILE(GERRYLIB/QCLSRC) MBR(FTPCMDS)
OVRDBF FILE(OUTPUT) TOFILE(GERRYLIB/QCLSRC) MBR(FTPLOG)

```

2. В команде STRTCPFTP, указанной в программе на CL, должно быть задано имя или IP-адрес удаленной системы. Если вы хотите указать имена удаленных систем в командах входного файла, а не в программе на CL, то в команде STRTCPFTP нужно указать фиктивное имя хоста. Это может быть имя несуществующего или существующего хоста. Если это имя существующего хоста, то первыми командами во входном файле должны быть ИД пользователя и пароль, а затем

должна быть указана команда CLOSE. Если это имя несуществующего хоста, то первой во входном файле должна быть команда OPEN для подключения к нужному серверу.

```
FTP RMTSYS(LOOPBACK)
```

FTP выполняет команды, указанные во входном файле, и записывает сообщения в выходной файл (FTPLOG).

3. После завершения работы FTP отмените использование файлов в качестве потоков ввода и вывода:

```
DLTOVR FILE(INPUT OUTPUT)
```

Программа на CL должна выглядеть примерно следующим образом (ниже приведен пример программы для системы SYSNAM01):

```
Колонки . . . : 1 71          Выбрать          GERRYLIB/QCLSRC
SEU==>
FMT **  ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
***** Начало данных *****
0001.00 PGM
0002.00          OVRDBF  FILE(INPUT) TOFILE(GERRYLIB/QCLSRC) +
0003.00                               MBR(FTPCMDS)
0004.00          OVRDBF  FILE(OUTPUT) TOFILE(GERRYLIB/QCLSRC) +
0005.00                               MBR(FTPLOG)
0006.00          FTP    RMTSYS(LOOPBACK) /* (FTP CL Program) */
0007.00          DLTOVR  FILE(INPUT OUTPUT)
0008.00 ENDPGM
***** Конец данных *****

F3=Выход  F5=Обновить  F9=Восстановить F10=Курсор F12=Отмена
F16=Повторить поиск  F24=Доп. клавиши
(C) COPYRIGHT IBM CORP. 1981, 1994.
```

Рисунок 1. Программа на CL FTPBATSH для запуска FTP в пакетном режиме.

Создайте входной командный файл FTP (FTCPDMS)

В этом файле должны содержаться все команды клиента FTP, необходимые для подключения к серверу, передачи файлов, закрытия соединения и завершения сеанса клиента. В приведенном ниже примере указаны команды, которые необходимо выполнить для передачи файлов в две удаленные системы.

```

Колонки . . . :   1 71          Выбрать          GERRYLIB/QCLSRC
SEU==>                                     FTPCMDS
FMT **  ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
***** Начало данных *****
0001.00 gwil ****
0002.00 close
0003.00 open sysnamrs
0004.00 user root root
0005.00 ascii
0006.00 syscmd dlrf file(gerrylib/rs6)
0007.00 get /Itsotest gerrylib/rs6.rs6
0008.00 close
0009.00 open mvax
0010.00 user tester tester
0011.00 get screen1.file gerrylib/vax.vax (replace
0012.00 close
0013.00 open sysnam02
0014.00 user gwil ****
0015.00 ebcdic
0016.00 put gerrylib/rs6.rs6 gerrylib/rs6.rs6
0017.00 quote rcmd sndnetf file(gerrylib/rs6) tousrid((gwil sysnam14))
0018.00 close
0019.00 quit
***** Конец данных *****
F3=Выход F5=Обновить F9=Восстановить F10=Курсор F12=Отмена
F16=Повторить поиск F24=Доп. клавиши

```

Рисунок 2. Передача файлов в две удаленные системы.

Ниже приведено описание команд клиента FTP, показанных на рисунке 2. Указанные номера задают номера строк в меню.

0001

ИД пользователя и пароль для фиктивного подключения к клиенту iSeries SYSNAM03.

0002

Закрыть фиктивное соединение в системе iSeries SYSNAM03.

0003

Открыть управляющее соединение с системой RISC System/6000^R SYSNAMRS.

0004

Команда USER с ИД пользователя и паролем для SYSNAMRS.

Примечание:

При работе FTP в пакетном режиме после команды OPEN должна быть вызвана команда OPEN. В команде USER должны быть заданы ИД пользователя и пароль. В интерактивном режиме это не требуется, так как клиент FTP автоматически выполняет команду USER и запрашивает ИД пользователя для входа в систему. В пакетном режиме автоматическое выполнение команды USER не предусмотрено.

0005

Передать данные ASCII (в системе iSeries они будут преобразованы из/в EBCDIC).

0006

Команда CL, удаляющая файл в клиентской системе iSeries. Вместо нее можно указать опцию (REPLACE в следующей команде).

0007

Получить файл из системы RISC System/6000.

0008

Закрыть управляющее соединение с системой RISC System/6000 SYSNAMRS.

0009

Установить соединение с системой VAX MVAX.

0010

Команда USER с ИД пользователя и паролем для MVAX.

0011

Получить файл из VAX с заменой существующего файла в iSeries.

0012

Закрыть управляющее соединение с системой VAX MVAX.

0013

Открыть управляющее соединение с удаленной системой iSeries SYSNAM02.

0014

Команда USER с ИД пользователя и паролем для SYSNAM02.

0015

Передать данные EBCDIC (точно так же, как и при передаче из одной системы iSeries в другую).

0016

Отправить файл из iSeries в систему iSeries SYSNAM02 по соединению TCP/IP.

0017

Отправить этот файл с сервера iSeries SYSNAM03 в удаленную систему iSeries SYSNAM14 по сети TCP/IP.

0018

Закрыть управляющее соединение с системой iSeries SYSNAM02.

0019

Завершить работу FTP.

Создайте программу на CL для передачи на выполнение задания FTPBATCN

Для того чтобы запланировать передачу файлов в автономном режиме, создайте программу на CL, которая будет запускать задание FTPBATCN. В приведенном ниже примере автономная передача файлов должна быть выполнена в пятницу, в 17:00.

```

Колонки . . . :   1  71          Выбрать          GERRYLIB/QCLSRC
SEU==>                                     FTPSUBMIT
FMT **   ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
          ***** Начало данных *****
0001.00 PGM
0002.00          SBMJOB      CMD(CALL PGM(GERRYLIB/FTPBATC)) +
0003.00                                     JOB(FTPFRIDAY) OUTQ(QUSRSYS/GERRYQ)  +
0004.00                                     SCDDATE(*FRI) SCDTIME(170000) /* FTP for +
0005.00                                     Friday, 5:00 in the afternoon */
0006.00 ENDPGM
          ***** Конец данных *****

F3=Выход  F5=Обновить  F9=Восстановить F10=Курсор F12=Отмена
F16=Повторить поиск  F24=Доп. клавиши
                                (C) COPYRIGHT IBM CORP. 1981, 1994.

```

Рисунок 3. Программа на CL, запускающая пакетное задание FTP.

Проверьте наличие сообщений об ошибках в выходном файле FTP

При работе в пакетном режиме FTP записывает данные в элемент файла FTPLOG. Содержимое элемента файла FTPLOG соответствует исходным операторам, указанным в обоих примерах.

```

Подключение к хосту LOOPBACK с адресом 127.0.0.1 через порт 21.
220-QTSP на локальном хосте.
220 Соединение будет прервано, если время простоя составит больше 5 минут.
Введите ИД пользователя (gwil):

>>>GWIL ****
331 Введите пароль.
230 Пользователь GWIL вошел в систему.
    Удаленная операционная система - OS/400. Версия TCP/IP - "V4R2M0".
250 Применяется формат "0".
257 Текущая библиотека - "QGPL".
Введите команду FTP.

> CLOSE
221 Получена команда QUIT.
Введите команду FTP.

> OPEN SYSNAMRS
Подключение к хосту SYSNAMRS с адресом 9.4.73.198 через порт 21.
220 Сервер FTP sysnamrs.sysnam123.ibm.com (версия 4.9, Вт, 2 сб, 20:35:07 CDT
    1993) готов к работе.
Введите команду FTP.

```

Рисунок 4. Вывод FTP (FTPLOG) после выполнения программы FTPBATC (часть 1)

```
> USER root ****
331 Введите пароль root.
230 Пользователь root вошел в систему.
UNIX Type: L8 Version: BSD-44
Введите команду FTP.

> ASCII
200 Type set to A; form set to N.
Введите команду FTP.

> SYSCMD DLTf FILE(GERRYLIB/RS6)
Введите команду FTP.

> GET /Itsotest GERRYLIB/RS6/RS7
200 Команда PORT выполнена.
150 Установлено соединение для передачи данных /Itsotest (467 байт).
226 Передача данных выполнена.
467 байт передано за 2.845 секунды. Скорость передачи - 0.167 Кб/с.
Введите команду FTP.
```

Рисунок 5. Вывод FTP (FTPLOG) после выполнения программы FTPBATCH (часть 2)

```
> CLOSE
221 Goodbye.
Введите команду FTP.

> OPEN MVAX
Подключение к хосту mvax с адресом 9.4.6.252 через порт 21.
220 Служба FTP готова к работе
Введите команду FTP.

> USER TESTER *****
331 Введите пароль для пользователя TESTER
230 Пользователь TESTER вошел в систему, каталог - $DISK1:[TESTER]
Введите команду FTP.

GET SCREEN1.FILE GERRYLIB/VAX.VAX (REPLACE
200 Команда PORT выполнена.
125 Передача файла $DISK1:[TESTER SCREEN1.FILE;1(266586 байт) в режиме ASCII
226 Передача файла выполнена.
265037 байт передано за 8.635 секунд. Скорость передачи данных - 30.694 Кб/с.
Введите команду FTP.

> CLOSE
221 Goodbye.
Введите команду FTP.

OPEN SYSNAM02
Подключение к хосту SYSNAM02 с адресом 9.4.73.250 через порт 21.
220-QTCP в системе SYSNAM02.sysnam123.ibm.com.
220 Соединение будет прервано, если время простоя составит больше 5 минут.
Введите команду FTP.
```

Рисунок 6. Вывод FTP (FTPLOG) после выполнения программы FTPBATCH (часть 3)

```

> USER GWIL ****
331 Введите пароль.
230 Пользователь GWIL вошел в систему.
    Удаленная операционная система - OS/400. Версия TCP/IP - "V4R2M0".
250 Применяется формат "0".
257 Текущая библиотека - "QGPL".
Введите команду FTP.

> EBCDIC
200 Тип представления данных - EBCDIC nonprint.
Введите команду FTP.

> PUT GERRYLIB/RS6.RS6 GERRYLIB/RS6.RS6
200 Команда PORT выполнена.
150 Отправка файла в элемент RS6 файла RS6 из библиотеки GERRYLIB.
250 Передача файла выполнена.
467 байт передано за 0.148 секунд. Скорость передачи данных - 3.146 Кб/с.
Введите команду FTP.

> RCMD SNDNETF FILE(GERRYLIB/RS6) TOUSRID((GERRYLIB SYSNAM14))
250 Команда SNDNETF FILE(GERRYLIB/RS6) TOUSRID((GWIL SYSNAM14))
    выполнена.
Введите команду FTP.

```

Рисунок 7. Вывод FTP (FTPL0G) после выполнения программы FTPBATCН (часть 4)

```

> CLOSE
221 Получена команда QUIT.
Введите команду FTP.
> QUIT
(Завершает работу FTP)

```

Рисунок 8. Вывод FTP (FTPL0G) после выполнения программы FTPBATCН (часть 5)

Проверьте, не содержит ли этот файл вывода сообщения об ошибках, которые могли произойти при работе с FTP. Вы можете либо просмотреть вывод вручную, либо воспользоваться специальной программой для поиска кодов ошибок. Трехзначные коды ошибок FTP начинаются с цифр 4 и 5. Учтите, что с этих цифр могут начинаться и обычные сообщения вида '467 байт передано...'

Пример процедуры: Вместе с продуктом TCP/IP поставляются пример процедуры на языке REXX и пример элемента физического файла. Файл QATMPINC из библиотеки QTCP состоит из следующих элементов:

- ВАТСНFTP - содержит исходный код процедуры на REXX, которая переопределяет входной и выходной файлы и запускает FTP.
- ВFTPFILE - содержит команды и данные, необходимые для входа в систему и работы с FTP.

Справочная информация о FTP

➤ Следующие разделы содержат информацию, которая может оказаться полезной при работе с серверами и клиентами FTP:

Команды сервера FTP

Это команды обмена данными между сервером и клиентом. Раздел также содержит описания эквивалентных команд CL iSeries, поддерживаемых только сервером FTP iSeries (См. стр. 41)

Команды клиента FTP

Команды клиента FTP позволяют подключиться к удаленному серверу FTP, просмотреть его библиотеки и каталоги, а также создать, удалить или передать файл.

Программы выхода FTP

Программы выхода позволяют обеспечить защиту FTP. Сервер FTP взаимодействует с каждой программой выхода через соответствующую точку выхода. Этот раздел содержит описания параметров и примеры программ.

Дополнительная справочная информация

- Способы передачи данных
- Файловые системы и соглашения о присвоении имен
- Коды ответа сервера FTP
- Правила описания формата команд сервера FTP
- Правила описания формата команд клиента FTP



Команды сервера FTP

В этом разделе приведено описание команд сервера FTP. Клиент FTP взаимодействует с сервером посредством набора команд сервера. Обычно пользователю не приходится работать с сервером FTP напрямую, поэтому здесь приведены только общие сведения: имена команд, их действие, формат вызова и ответные сообщения о состоянии FTP.

Сервер FTP системы iSeries поддерживает следующие команды:

Команда	Описание
ABOR	Отменяет выполнение предыдущей команды
ADDM	Добавляет элемент в физический файл
ADDV	Добавляет элемент в элемент переменной длины физического файла
APPE	Добавляет данные в указанный файл
AUTH	Задаёт способ идентификации для текущего сеанса FTP.
CDUP	Переход в родительский каталог
CRTL	Создаёт библиотеку
CRTP	Создаёт физический файл
CRTS	Создаёт исходный физический файл
CWD	Изменяет рабочий каталог или библиотеку
DBUG	Включает или выключает трассировку сервера
DELE	Удаляет файл, элемент или документ
DLTF	Удаляет файл
DLTL	Удаляет библиотеку
HELP	Выдаёт информацию о командах сервера FTP
LIST	Выдаёт список файлов или каталогов
MKD	Создаёт каталог
MODE	Задаёт формат передачи данных
NLST	Выдаёт список имен файлов или каталогов

Команда	Описание
NOOP	Проверяет, отвечает ли сервер
PASS	Передает пароль на сервер
PASV	Передает серверу запрос на установление следующего пассивного информационного соединения
PBSZ	Задаёт максимальный размер буфера, применяемого для приема или передачи данных, зашифрованных на уровне приложения.
PORT	Задаёт порт данных, через который устанавливается соединение с клиентом
PROT	Задаёт уровень защиты информационных соединений FTP
PWD	Показывает имя текущего рабочего каталога
QUIT	Выход пользователя из системы, завершение сеанса
RCMD	Передает команду CL на сервер FTP
REIN	Заново запускает сеанс на сервере
RETR	Получает данные с сервера
RMD	Удаляет каталог
RNFR	Задаёт файл, который должен быть переименован
RNTO	Задаёт новое имя файла
SITE	Отправляет на сервер необходимую информацию
STAT	Получает информацию о состоянии с сервера
STOR	Сохраняет данные на сервере с заменой существующего файла
STOU	Сохраняет данные на сервере без замены существующего файла
STRU	Задаёт структуру файла
SYST	Выдает имя операционной системы сервера
TIME	Задаёт значение тайм-аута для сервера FTP
TYPE	Задаёт тип передачи файлов
USER	Передаёт ИД пользователя на сервер
XCUP	Переход в родительский каталог
XCWD	Переход в рабочий каталог
XMKD	Создаёт каталог
XPWD	Показывает текущий каталог или библиотеку
XRMD	Удаляет каталог

Особые команды сервера FTP системы iSeries

В набор команд сервера FTP системы iSeries входят особые команды, которые представляют собой сокращенные названия более длинных команд CL. В соответствии с правилами, установленными для протокола FTP, длина имен этих команд не превышает 4 символа. Сервер iSeries интерпретирует эти команды следующим образом:

- ADDM = ADDPFM (Добавить элемент физического файла)
- ADDV = ADDPVLML (Добавить элемент переменной длины в физический файл)

- CRTL = CRTLIB (Создать библиотеку)
- CRTP = CRTPF (Создать физический файл)
- CRTS = CRTSRCPF (Создать исходный физический файл)
- DLTF = DLTF (Удалить файл)
- DLTL = DLTLIB (Удалить библиотеку)

Любую другую команду CL можно отправить на сервер с помощью команды сервера FTP RCMD.

Связанные разделы:

- Правила описания формата команд сервера FTP
- Коды ответов сервера FTP
- Команды клиента FTP: С помощью этих команд можно подключиться к удаленному серверу FTP, просмотреть его библиотеки и каталоги, создать, удалить или передать файл.

ADDM (Добавить элемент физического файла) **Команда сервера FTP**

ADDM параметры

параметры

Параметры этой команды совпадают с параметрами команды CL ADDPFM.

Например, для добавления элемента BANANA в физический файл GEORGE в библиотеке RLKAYS сервера iSeries введите:

```
ADDM FILE(RLKAYS/GEORGE) MBR(BANANA)
```

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

ADDV (Добавить элемент переменной длины в физический файл) **Команда сервера FTP**

ADDV параметры

параметры

Параметры этой команды совпадают с параметрами команды CL ADDPVLM.

Например, для добавления элемента POLEBEAN в физический файл GEORGE в библиотеке RLKAYS сервера iSeries введите:

```
ADDV FILE(RLKAYS/GEORGE) MBR(POLEBEAN)
```

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

APPE (Добавить к существующему файлу) **Команда сервера FTP**

Команда сервера FTP APPE принимает передаваемые данные и сохраняет их в файле на сервере. Если указанный файл существует, то данные добавляются к его текущему содержимому; в противном случае создается новый файл.

APPE имя-файла

имя-файла

Имя файла, в который будут записаны передаваемые данные на сервере.

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

AUTH (Способ идентификации)

Команда сервера FTP

Команда сервера FTP AUTH задает способ идентификации (защиты), который будет применяться в текущем сеансе FTP. Ниже описан формат этой команды:

AUTH [TLS-C | TLS-P | TLS | SSL]

Значения параметров:

TLS-C	Для защиты данных будет применяться протокол защиты транспортного уровня (TLS). Параметрам защиты соединения будут присвоены значения по умолчанию, определенные в RFC2228. Другими словами, для защиты соединения для передачи данных не будут применяться неявные средства.
TLS-P	Для защиты данных будет применяться протокол TLS. Дополнительно будут применяться неявные средства защиты соединения для передачи данных (что эквивалентно вызову последовательности команд AUTH TLC-C, PBSZ 0, PROT P)
TLS	Аналог TLS-C.
SSL	Аналог TLS-P.

Примечание:

Протокол TLS совместим с протоколом SSL.

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

CRTL (Создать библиотеку)

Команда сервера FTP

CRTL параметры

параметры

Параметры этой команды совпадают с параметрами команды CL CRTLIB.

Например, для создания библиотеки TESTTCP на сервере iSeries введите следующую команду:
CRTL TESTTCP

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

**C RTP (Создать физический файл)
Команда сервера FTP**

C RTP параметры

параметры

Параметры этой команды совпадают с параметрами команды CL CRTPF.

Например, для того чтобы создать физический файл MYFILE с длиной записи 80 байт и без ограничений на количество элементов, введите:

```
C RTP FILE(RLKAYS/MYFILE) RCDLEN(80) MAXMBRS(*NOMAX)
```

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

**C RTS (Создать исходный физический файл)
Команда сервера FTP**

C RTS параметры

параметры

Параметры этой команды совпадают с параметрами команды CL CRTSRCPF.

Например, для создания исходного физического файла с именем GEORGE в библиотеке RLKAYS, введите:

```
C RTS FILE(RLKAYS/GEORGE)
```

Перейдите к следующим разделам:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

**C WD (Изменить рабочий каталог или библиотеку)
Команда сервера FTP**

Для изменения текущего рабочего каталога, библиотеки или группы файлов служит команда сервера FTP CWD:

C WD каталог

Перейти:

- Команды сервера FTP

- Правила описания формата команд сервера FTP

DBUG (Включить трассировку сервера FTP)

Примечание: Трассировку сервера FTP следует выполнять только для того, чтобы сообщить о неполадках в фирму IBM. Включение этой функции может привести к снижению производительности сервера.

Команда сервера FTP

```
DBUG
```

Если трассировка сервера FTP не была запущена, она будет включена. Трассировка будет выполняться до повторного вызова команды DBUG или выполнения команды QUIT. После завершения трассировки может потребоваться значительное время на форматирование полученных данных.

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

DELE (Удалить файл или документ)

Команда сервера FTP

Для удаления файла, элемента или документа служит команда сервера FTP DELE:

```
DELE удаленный-файл
```

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

DLTF (Удалить файл)

Команда сервера FTP

```
DLTF параметры
```

параметры

Параметры этой команды совпадают с параметрами команды CL DLTF.

Например, для удаления файла MYFILE из библиотеки RLKAYS введите:

```
DLTF FILE(RLKAYS/MYFILE)
```

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

DLTL (Удалить библиотеку)

Команда сервера FTP

DLTL параметры

параметры

Параметры этой команды совпадают с параметрами команды CL DLTLIB.

Например, для удаления библиотеки введите следующую команду:

DLTL имя-библиотеки

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

HELP (Получение справки с удаленного сервера iSeries)

Для получения информации о командах сервера FTP введите команду HELP в следующем формате:

HELP [команда]

команда

Команда сервера, информацию о которой вы хотите получить. Например, команда HELP ADDM предоставит информацию о добавлении элемента к физическому файлу сервера iSeries.

Для того чтобы узнать формат команды ADDV сервера iSeries, введите следующую команду сервера:

HELP ADDV

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

LIST (Список файлов)

Команда сервера FTP

Для того чтобы просмотреть содержимое каталога, библиотеки или группы файлов, вызовите команду сервера FTP LIST:

LIST [каталог | имя]

В список будут включены только те файлы, передачу которых поддерживает FTP

Перейдите к следующим разделам:

- Команды сервера FTP
- Правила описания формата команд сервера FTP
- SITE (Отправить серверу информацию о клиенте): Эта команда служит для изменения формата вывода команды LIST.
- Вывод команды LIST в формате UNIX: Эта команда позволяет просмотреть формат вывода команды LIST.
- Вывод команды LIST в формате iSeries: Эта команда позволяет просмотреть формат вывода команды LIST.

MKD (Создать каталог) Команда сервера FTP

Для создания каталога предназначена команда сервера FTP MKD:

```
MKD имя-каталога
```

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

MODE (Задать режим передачи) Команда сервера FTP

Команда сервера FTP MODE предназначена для задания режима (формата) передачи данных:

```
MODE [B | S]
```

B

Блочный режим. В этом режиме данные передаются как последовательность блоков, к каждому из которых добавляется заголовок из одного или нескольких байт.

S

Потоковый режим. В этом режиме данные передаются в виде потока байт. В потоковом режиме вы можете выбрать любой тип представления данных. Этот режим более эффективен, так как не передается информация о блоках данных.

Примечания:

1. Потоковый режим применяется на сервере iSeries по умолчанию. По возможности всегда рекомендуется выбирать этот режим передачи.
2. Если вы укажете эту команду без параметров, сервер выдаст информацию о текущем режиме.

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

NLST (Список имен) Команда сервера FTP

Команда сервера FTP NLST предназначена для просмотра списка имен файлов в каталоге, библиотеке или группе файлов:

```
NLST [каталог | имя]
```

В список будут включены только те файлы, передачу которых поддерживает FTP.

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

NOOP (Получить ответ сервера)

Команда сервера FTP

Команда сервера FTP NOOP передает клиенту ответ "OK". Она не выполняет никаких действий на сервере. Эта команда применяется клиентами для проверки соединения с сервером и работоспособности сервера. Формат команды NOOP:

```
NOOP
```

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

PASS (Пароль)

Команда сервера FTP

```
PASS пароль
```

пароль

Строка, задающая ваш пароль на сервере.

Примечание: Перед командой сервера PASS должна быть вызвана команда сервера USER.

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

PASV (Установить пассивное информационное соединение)

Команда сервера FTP

Команда PASV указывает, что следующее информационное соединение должно быть установлено сервером в пассивном режиме. Формат команды:

```
PASV
```

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

PBSZ (Размер буфера защиты)

Команда сервера FTP

Команда PBSZ устанавливает максимальный размер буфера, применяемого для приема или передачи данных, зашифрованных на уровне приложения. Ниже описан формат этой команды:

PBSZ значение

где *значение* - символьная строка ASCII, представляющая десятичное целое число.

Примечание: В соответствии со стандартом RFC2228 команда PBSZ должна вызываться перед командой PROT. Однако при работе с TLS/SSL данные обрабатываются поблочно, поэтому допустимо только нулевое значение.

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

PORT (Порт данных) **Команда сервера FTP**

Команда сервера FTP PORT позволяет задать порт, через который клиент будет устанавливать информационное соединение:

```
PORT h1,h2,h3,h4,p1,p2
```

h1

Десятичное значение от 0 до 255, задающее IP-адрес системы.

p1

Десятичное значение от 0 до , задающее номер порта TCP.

Значения p1 и p2 преобразуются в номер порта TCP по формуле:

$$\text{номер-порта} = (p1 * 256) + p2$$

Например, команда PORT

```
PORT 9,180,128,180,4,8
```

задает номер порта 1032 для IP-адреса 9.180.128.180.

Примечание: Согласно RFC 1122 по TCP/IP, после разрыва соединения сервер не может подключаться к клиенту с тем же IP-адресом через тот же порт в течение 2 минут. Однако вы можете устанавливать соединения с тем же IP-адресом, но через другие порты.

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

PROT (Уровень защиты канала для передачи данных) **Команда сервера FTP**

Команда PROT определяет параметры защиты информационных соединений FTP (то есть соединений, по которым передаются файлы и информация о содержимом каталогов). Ниже описан формат этой команды:

```
PROT [ C | P ]
```

Значения параметров:

C	Обычное соединение. Данные передаются по информационному соединению "в незащищенном виде".
P	Частное соединение. Для защиты данных, передаваемых по информационному соединению, применяется протокол TLS/SSL, который обеспечивает целостность и секретность данных.

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

**PWD (Показать рабочий каталог или библиотеку)
Команда сервера FTP**

При вызове команды сервера FTP PWD сервер возвращает клиенту имя текущего каталога или библиотеки:

PWD

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

**QUIT (Завершить сеанс сервера FTP)
Команда сервера FTP**

Команда сервера FTP QUIT отключает программу-клиент от сервера и закрывает управляющее соединение. Если в момент ее вызова выполняется передача файла, то соединение будет открытым до завершения операции, а затем сервер закроет его.

QUIT

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

**RCMD (Отправить команду CL в систему сервера FTP)
Команда сервера FTP**

Команда сервера RCMD предназначена для запуска команд CL в системе сервера FTP. Длина строки команды RCMD не должна превышать 1000 символов. В команде RCMD не предусмотрена возможность вызова приглашения команды CL, поэтому в ней должны быть указаны все необходимые параметры.

Если команда CL выполнена без ошибок, то будет показано сообщение об успешном выполнении команды. В противном случае появится сообщение об ошибке. Это сообщение не содержит информацию о характере ошибки, кроме случаев, когда в команде неверно указано имя библиотеки, файла или элемента.

Ниже приведен пример использования команды RCMD для выполнения команды Удалить файл (DLTF):

```
QUOTE RCMD DLTF FILE(mylib/myfile)
```

mylib - это имя библиотеки, из которой должен быть удален файл. myfile - это имя удаляемого файла.

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

Ознакомьтесь с дополнительной информацией о сервере REXEC, который предоставляет альтернативный способ для запуска команд CL в удаленной системе.

**REIN (Повторно инициализировать сеанс)
Команда сервера FTP**

```
REIN
```

Команда REINITIALIZE:

1. Завершает текущую передачу данных
2. Завершает сеанс USER и удаляет всю информацию ввода/вывода и сведения о счете
3. Присваивает всем параметрам сервера значения по умолчанию
4. Оставляет управляющее соединение открытым

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

**RETR (Получить файл)
Команда сервера FTP**

Для получения данных с сервера предназначена команда сервера FTP RETR.

```
RETR удаленный-файл
```

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

**RMD (Удалить каталог)
Команда сервера FTP**

Для удаления каталога служит команда сервера FTP RMD:

```
RMD имя-каталога
```

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

RNFR (Переименовать из) Команда сервера FTP

Для переименования файлов применяется команда сервера FTP RNFR. После этой команды должна быть вызвана команда RNTO (Переименовать в).

```
RNFR имя-файла
```

имя-файла

Имя файла, который нужно переименовать.

Примечание: При переименовании файла системы iSeries нельзя изменять его файловую систему.

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

RNTO (Переименовать в) Команда сервера FTP

При переименовании файла его новое имя задается с помощью команды FTP RNTO. Она должна выполняться после команды RNFR, в которой было указано исходное имя файла.

```
RNTO имя-файла
```

имя-файла

Новое имя файла.

Примечание: При переименовании файла системы iSeries нельзя изменять его файловую систему.

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

SITE (Отправить информацию о клиенте) Команда сервера FTP

Для передачи на сервер информации, которая ему требуется для выполнения определенных операций, служит команда сервера FTP SITE.

```
SITE [параметры]
```

Сервер FTP системы iSeries поддерживает следующие параметры команды SITE:

LISTFMT 0

Вывод команды LIST будет возвращен сервером в формате iSeries, впервые представленном в выпуске V3R1M0. Клиент iSeries поддерживает два формата вывода информации: формат iSeries и формат UNIX

LISTFMT 1

Вывод команды LIST будет возвращен сервером в формате UNIX. Имя файла находится в последнем поле каждой строки списка. Клиент iSeries поддерживает два формата вывода информации: формат iSeries и формат UNIX

LISTFMT

Возвращает текущее значение параметра LISTFMT сервера FTP.

Примечания:

Для того чтобы изменить на сервере значение LISTFMT по умолчанию, вызовите команду CHGFTPА с параметром LISTFMT. Это свойство сервера FTP можно настроить и с помощью Навигатора iSeries:

1. В окне программы Навигатор разверните **значок сервера iSeries** → **Сеть** → **Серверы** → **TCP/IP**.
2. На правой панели щелкните правой кнопкой мыши на значке **FTP** и выберите пункт **Свойства**.
3. Перейдите на страницу **Начальные форматы**.
4. Под заголовком **Список файлов** выберите iSeries или UNIX в качестве значения по умолчанию для параметра LISTFMT.
5. Для сохранения изменений нажмите кнопку **OK**.

NAMEFMT 0

Применять формат имен Библиотека/Файл.Элемент. Этот формат применяется только для файлов базы данных библиотечной файловой системы.

NAMEFMT 1

Использовать формат имен с указанием пути. Этот формат имен применяется для всех файловых систем, поддерживаемых FTP (включая библиотечную файловую систему). Для работы со всеми файловыми системами iSeries, кроме библиотечной, должен быть установлен формат имен 1.

NAMEFMT

Показывает сообщение с текущим значением формата имен файлов сервера.

Примечание:

Для того чтобы задать значение по умолчанию для параметра NAMEFMT сервера FTP системы iSeries, вызовите команду CHGFTPА с параметром NAMEFMT.

CRTCCSID *CALC

В новых файлах базы данных, создаваемых при передаче файлов в формате ASCII, применяется CCSID EBCDIC по умолчанию, соответствующий CCSID передаваемого файла ASCII.

CRTCCSID *USER

Для новых файлов базы данных, создаваемых при передаче файлов ASCII, применяется CCSID текущего задания. Если этот CCSID равен 65535, то применяется CCSID по умолчанию, определяемый идентификатором языка, указанным для текущего задания.

CRTCCSID *SYSVAL

Для новых файлов базы данных, создаваемых при передаче файлов ASCII, применяется CCSID, указанный в системном значении QCCSID.

CRTCCSID [номер-CCSID]

Укажите CCSID, который будет применяться для файлов базы данных клиента, создаваемых при передаче файлов ASCII. Это значение будет подтверждено сервером.

CRTCCSID

Показывает сообщение с текущим значением CRTCCSID для клиента FTP.

NULLFLDS 0

Сервер не разрешает передавать файлы базы данных с пустыми полями. Это значение применяется по умолчанию.

NULLFLDS 1

Сервер разрешает передавать файлы базы данных с пустыми полями.

Примечание: Передача файлов с пустыми полями возможна только тогда, когда она разрешена одновременно и для клиента, и для сервера. Если сервер передает файл, содержащий пустые поля, в систему, отличную от iSeries, либо при передаче файла происходит преобразование кодовой страницы данных, результат будет непредсказуемым.

NULLFLDS

Показывает текущее значение параметра NULLFLDS сервера FTP.

TRIM 0

Выключает опцию Trim. Сервер передает записи базы данных вместе с конечными пробелами.

TRIM 1

Включает опцию Trim. При передаче файлов базы данных с сохранением файловой структуры в потоковом режиме конечные пробелы в записях базы данных не передаются. Это значение применяется по умолчанию.

TRIM 2

Конечные пробелы записей базы данных не передаются в любом случае, даже при передаче структуры записей и в блочном режиме.

TRIM

Показывает текущее значение параметра TRIM сервера FTP.

Примечания:

1. В предыдущих версиях, не поддерживающих эту команду, конечные пробелы в записях базы данных файловой системы QSYS.LIB всегда удалялись перед передачей файла на сервер.
2. Опция TRIM не поддерживается для файлов в двоичном формате (TYPE I). В этом случае пробелы никогда не передаются, независимо от значения параметра TRIM.

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

STOR (Записать файл)

Команда сервера FTP

Команда сервера FTP STOR предназначена для записи файлов на сервер с заменой существующих файлов. Формат команды:

STOR удаленный-файл

Перейти:

- Команды сервера FTP

- Правила описания формата команд сервера FTP

STOU (Записать файл без замены существующего)

Команда сервера FTP

Команда сервера FTP STOU предназначена для записи файлов на сервер без замены существующих файлов. Формат команды:

```
STOU удаленный-файл
```

Сервер создает уникальное имя файла. Имя нового файла передается клиенту в ответном сообщении.

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

STRU (Задать структуру файла)

Команда сервера FTP

Команда сервера FTP STRU позволяет передавать файл, как непрерывную последовательность байтов. Формат команды:

```
STRU [F | R]
```

F

Структура "файл". Файл представляет собой непрерывную последовательность байт данных.

R

Структура "запись". Файл представляет собой последовательность записей.

Примечания:

1. От структуры файла зависит режим передачи, а также интерпретация и способ хранения данных.
2. Если вы укажете эту команду без параметров, сервер выдаст информацию о текущей структуре файлов.

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

SYST (Определить операционную систему)

Команда сервера FTP

Команда сервера FTP SYST позволяет узнать, какая операционная система установлена на сервере:

```
SYST
```

Формат вывода команды зависит от типа удаленной системы.

При работе с сервером iSeries в выводе команды указывается версия TCP/IP.

Ниже приведен пример ответа сервера:

Удаленная операционная система - OS/400. Версия TCP/IP - "V4R4M0".

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

TIME (Задать значения тайм-аутов для сервера FTP)

Команда сервера FTP

После настройки управляющего соединения между клиентом и сервером FTP начинает отслеживаться время простоя сервера. Максимально допустимое время простоя управляющего соединения называется тайм-аутом простоя.

Кроме того, предусмотрено максимально допустимое время простоя информационного соединения - тайм-аут передачи.

Формат команды сервера FTP TIME следующий:

```
TIME тайм-аут-простоя [тайм-аут-передачи]
```

тайм-аут-простоя

Максимально допустимое время простоя управляющего соединения с клиентом, по истечении которого оно будет автоматически закрыто. Допустимы значения от 1 до 9 999 999 секунд. По умолчанию тайм-аут простоя равен 300 секундам.

тайм-аут-передачи

Тайм-аут передачи файлов в секундах. Это необязательный параметр. Если вы не укажете этот параметр, то текущее значение изменено не будет. Допустимы значения от 1 до 9 999 999 секунд. Значение по умолчанию - 420 секунд.

Например, для того чтобы установить тайм-аут простоя сервера FTP, равный 1000 секундам, не изменяя тайм-аут передачи, нужно выполнить следующую команду:

```
QUOTE TIME 1000
```

Команда TIME не входит в стандартный набор команд. Она поддерживается только сервером FTP системы iSeries.

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

TYPE (Задать способ представления данных)

Команда сервера FTP

Для задания типа передачи файла или способа представления данных при передаче предназначена команда сервера FTP TYPE:

```
TYPE [ A
      B [ 1 | 2 | 3 [A|R] | 4 [A|R ] | 5 | 6 | 7 ]
      C ccsid#
      E
      F [ 1 ]
      I ]
```


A

Задает тип передачи ASCII (значение по умолчанию). Форматирование файла по вертикали сервером не выполняется. В режиме ASCII сервер поддерживает только формат по умолчанию - NON PRINT. Этот тип передачи предназначен для обмена текстовыми файлами в тех случаях, когда хотя бы одна из систем не поддерживает тип EBCDIC.

Примечание: Для TYPE A применяется CCSID, указанный в атрибутах конфигурации сервера FTP. Эти атрибуты можно изменить с помощью команды CHGFTP.

B

Канжи JIS с открывающим/закрывающим символами (CCSID 932)

B 1

Канжи JIS с открывающим/закрывающим символами (CCSID 932)

B 2

Расширенный набор символов Канжи для UNIX (CCSID 5050)

B 3

JIS 1983 с открывающей Esc-последовательностью ASCII (CCSID 5054)

B 3 A

JIS 1983 с открывающей Esc-последовательностью ASCII (CCSID 5054)

B 3 R

JIS 1983 с открывающей Esc-последовательностью JISROMAN (CCSID 5052)

B 4

JIS 1978 с открывающей Esc-последовательностью ASCII (CCSID 5055)

B 4 A

JIS 1978 с открывающей Esc-последовательностью ASCII (CCSID 5055)

B 4 R

JIS 1978 с открывающей Esc-последовательностью JISROMAN (CCSID 5053)

B 5

Хангул (CCSID 934)

B 6

Корейский стандартный набор символов KSC-5601, версия 1989 (CCSID 949)

B 7

Традиционный китайский (5550) (CCSID 938)

C

Задает CCSID (идентификатор кодированного набора символов), поддерживаемый системой. Перед CCSID должен быть указан символ C.

E

Задает тип передачи EBCDIC. Форматирование файла по вертикали сервером не выполняется. В режиме EBCDIC поддерживается только формат по умолчанию - NON PRINT. Тип передачи EBCDIC предназначен для обмена файлами между системами с внутренним представлением символов EBCDIC.

F

Канжи IBM EBCDIC (CCSID 5035)

F 1

Канжи IBM EBCDIC (CCSID 5035)

I

Задаёт двоичный тип передачи. В двоичном режиме данные передаются как последовательность битов, сгруппированных по 8 в байты. Этот режим предназначен для эффективной передачи файлов и двоичных данных, например, исходного кода.

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

USER (Отправить ИД пользователя на сервер)

Команда сервера FTP

USER имя-пользователя

имя-пользователя

Имя пользовательского профайла на сервере iSeries.

Примечание:

Если команда USER будет выполнена, и для входа в систему iSeries необходимо ввести пароль, то клиенту будет отправлено приглашение на ввод пароля. Для передачи пароля нужно вызвать команду PASS. Если для системы установлен уровень защиты 10, то приглашение на ввод пароля не отправляется.

Перейти:

- Команды сервера FTP
- Правила описания формата команд сервера FTP

Команды клиента FTP

Команды Протокола передачи файлов (FTP) предназначены для передачи файлов от одного компьютера к другому. С помощью команд клиента FTP можно подключиться к удалённому серверу FTP, просмотреть его библиотеки и каталоги, создать, удалить или передать файл.

Список команд клиента и их описание приведены в указанных ниже разделах.

Ниже перечислены команды, поддерживаемые клиентом FTP сервера iSeries. В таблице приведены команды клиента, их общепринятые сокращения и описание назначения.

Команда	Назначение
?	Выдает справку по работе с FTP
ACCT	Передаёт учетную запись пользователя в удалённую систему
APPEND	Добавляет элемент локального файла к файлу удалённой системы
ASCII	Указывает, что файлы нужно передавать в формате ASCII
BINARY	Устанавливает двоичный тип передачи файлов

Команда	Назначение
CD	Изменяет рабочий каталог в удаленной системе
CDUP	Изменяет родительский каталог в удаленной системе
CLOSE	Завершает сеанс работы с удаленной системой
DEBUG	Включает или выключает режим отладки
DEBUG	Изменяет значения тайм-аутов клиента
DELETE	Удаляет файл из удаленной системы
DIR	Выдает список каталогов и файлов удаленной системы
EBCDIC	Указывает, что файлы нужно передавать в формате EBCDIC
GET	Копирует файл из удаленной системы в локальную
HELP	Выдает информацию о командах клиента FTP
LCD	Переходит в другой каталог локальной системы
LOCSITE	Задаёт информацию о локальном хосте
LOCSTAT	Выдает информацию о состоянии локального хоста
LPWD	Выдает имя рабочего каталога локальной системы
LS	Показывает список имен файлов из набора файлов удаленной системы
LTYPE	Задаёт тип передачи файлов в локальной системе
MDELETE	Удаляет несколько файлов в системе сервера
MGET	Копирует один или несколько файлов из удаленной системы
MKDIR	Создает каталог или подкаталог
MODE	Задаёт формат данных для функции передачи файлов
MPUT	Передаёт один или несколько локальных файлов в удаленную систему
NAMEFMT	Задаёт формат имен файлов
NOOP	Проверяет наличие ответа
NULLFLDS	Разрешает применение пустых полей
OPEN	Устанавливает соединение с сервером FTP
PASS	Передаёт пароль пользователя
PUT	Копирует элемент локального файла в удаленную систему
PWD	Выдает имя текущего каталога удаленной системы
QUIT	Завершает сеанс FTP
QUOTE	Отправляет команду на сервер FTP
REINITIALIZE	Повторно устанавливает сеанс с удаленной системой
RENAME	Переименовывает файл удаленной системы
RESET	Очищает очередь ответов сервера
RMDIR	Удаляет каталог удаленной системы
SECDATA	Указывает уровень защиты для соединения данных, если установлено защищенное соединение с сервером FTP.

Команда	Назначение
SECOPEN	Открывает защищенное управляющее соединение с сервером FTP с помощью указанного протокола защиты.
SENDPASV	Указывает, была ли отправлена команда PASV
SENDPORT	Указывает, была ли отправлена команда PORT
SENDSITE	Указывает, была ли отправлена команда SITE
SITE	Отправляет информацию о хосте удаленной системе
STATUS	Получает информацию о состоянии удаленной системы
STRUCT	Задаёт файловую структуру передаваемых данных
SUNIQUE	Управляет размещением файлов
SYSCMD	Запускает команду CL в локальной системе без завершения сеанса FTP
SYSTEM	Выдает имя операционной системы, установленной в удаленной системе
TYPE	Задаёт тип передачи файлов
USER	Отправляет ИД пользователя в удаленную систему
VERBOSE	Задаёт режим показа ответов сервера FTP

Связанные разделы:

- Правила описания формата команд клиента FTP
- Команды сервера FTP
- Сообщения с кодом ответа сервера FTP: Ознакомьтесь с общими кодами ответов и их значениями.

АССТ (Отправить учетную запись) Команда клиента FTP

Для выполнения некоторых системных функций в определенных системах необходимо предоставить учетную информацию. Эта информация будет запрошена удаленной системой. Для передачи учетной информации вызовите команду клиента FTP АССТ (Учетная запись):

АССТ <i>учетная-информация</i>

учетная-информация

Строка, задающая учетную запись пользователя. Например, это может быть пароль, по которому сервер предоставляет пользователю определенные права доступа. (Ваш пароль в удаленной системе.)

Например, функция TCP/IP операционной системы IBM Virtual Machine (VM) может потребовать пароль для чтения и записи информации с минидисков. С помощью команды АССТ нужно указать пароль для доступа к минидиску текущего каталога. Если в качестве удаленной системы выступает сервер iSeries, то команда АССТ не выполняет никаких действий.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

APPEND (Добавить элемент локального файла в удаленный файл)

Команда клиента FTP

Для того чтобы добавить к удаленному файлу элемент локального файла, документ или файл другой файловой системы, выполните команду клиента FTP APPEND в следующем формате:

```
APPend локальный-файл [удаленный-файл]
```

локальный-файл

Имя элемента локального файла, документа или другого файла iSeries. Указывает имя файла иерархической файловой системы (HFS), добавляемого в каталог удаленной системы.

Информация об именах файлов приведена в разделе NAMEFMT (Выбор формата имен файлов).

удаленный-файл

Файл удаленной системы. Если вы не укажете имя удаленного файла, то клиент FTP задаст имя по умолчанию. Информация о том, как в FTP создаются имена по умолчанию, приведена в разделе Имена файлов по умолчанию в командах передачи данных клиента.

Если указанный файл не существует на сервере FTP, то он будет создан.

Для добавления информации к файлу удаленной системы необходимы права на запись в этот файл. Для предоставления учетной информации служит команда ACCT. Информация о ней приведена в разделе ACCT (Отправить учетную запись).

По умолчанию файлы копируются в потоковом режиме. Вы можете выбрать другой режим с помощью команды MODE. Если в удаленном файле применяется фиксированный формат записей, то этот формат и длина записи будут сохранены. При необходимости записи из элемента локального файла будут усечены или дополнены пробелами.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

ASCII (Изменить тип передачи на ASCII)

Команда клиента FTP

```
AScii
```

Существует два основных типа файлов, применяемых при передаче файлов по протоколу FTP: ASCII и BINARY. Файлы ASCII содержат обычный текст. Эти файлы могут быть без расширения или с расширением .txt. Файлы BINARY - это программы и другие нетекстовые файлы, сохраненные в формате создавшего их приложения или в формате архивных или сжатых файлов.

Для обмена текстовыми файлами с системами, не поддерживающими EBCDIC, применяется тип передачи ASCII. Тип ASCII выбран по умолчанию. Форматирование файла по вертикали сервером не выполняется. ASCII поддерживает только формат по умолчанию NON PRINT.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

BINARY (Изменить тип передачи на двоичный)

Команда клиента FTP

Binary

Существует два основных типа файлов, применяемых при передаче файлов по протоколу FTP: ASCII и BINARY. Файлы ASCII содержат обычный текст. Эти файлы могут быть без расширения или с расширением .txt. Файлы BINARY - это программы и другие нетекстовые файлы, сохраненные в формате создавшего их приложения или в формате архивных или сжатых файлов.

Если вы добавляете двоичные данные в существующий файл iSeries, то длина записи будет равна длине записи в этом файле iSeries. Размер файла будет адаптирован под новые данные. Если указанный файл не существует на сервере iSeries, то FTP автоматически выберет длину.

Для передачи некоторых типов файлов, например, файлов сохранения, необходимо установить двоичный тип передачи. Если при передаче таких файлов параметру TYPE будет присвоено другое значение, появится сообщение о том, что необходимо установить двоичный тип передачи.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

CD (Изменить рабочий каталог или библиотеку)

Команда клиента FTP

Команда клиента FTP Изменить каталог (CD) предназначена для изменения рабочего каталога, библиотеки или группы файлов в удаленной системе:

CD каталог

каталог

Имя каталога, библиотеки или группы файлов в удаленной системе.

Если роль удаленной системы играет сервер iSeries, то эта команда изменяет текущий каталог или библиотеку. Список каталогов удаленной системы можно просмотреть с помощью команды Каталог (DIR).

Будьте внимательны при выполнении команды DIR. Советы и рекомендации по применению этой команды приведены в разделе DIR (Показать содержимое каталога, библиотеки или файлов).

Примечание:

При переходе в другую файловую систему iSeries с помощью команды CD (или LCD) нужно указать корневой каталог новой файловой системы.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

CLOSE (Завершить сеанс FTP с удаленной системой)

Команда клиента FTP

Для того чтобы завершить сеанс работы с удаленной системой без завершения работы клиента FTP в локальной системе iSeries, вызовите команду клиента FTP CLOSE:

```
CLOSE
```

После выполнения команды CLOSE среда FTP остается активной, и вы можете установить сеанс связи с другой системой. Для запуска сеанса связи с другой удаленной системой нужно вызвать команду OPEN. Для того чтобы завершить сеанс работы с сервером FTP и вернуться в среду той системы iSeries, в которой был запущен сеанс FTP, нужно вызвать команду QUIT.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

DEBUG (Трассировка клиента и вывод на экран команд сервера, отправляемых в удаленную систему)

Примечание: Трассировку клиента FTP следует выполнять только для создания отчета о программных неполадках для IBM. Трассировка может существенно снизить производительность системы.

Эта команда доступна в операционной системе OS/400, начиная с выпуска V4R4.

Команда клиента FTP

Для выполнения трассировки клиента FTP или просмотра переданных команд выполните команду клиента FTP DEBUG. Команда DEBUG включает и выключает режим отладки. Если задан необязательный параметр отладки, с его помощью будет установлен уровень отладки. В режиме отладки все команды, передаваемые на сервер, выводятся на экране с символами '>>>'. Для трассировки клиента FTP необходимо задать уровень отладки, равный 100.

```
DEBUG [уровень-отладки]
```

уровень-отладки

Если это значение равно 0, то режим отладки будет выключен. Если будет указано положительное целое число, то режим отладки будет включен. Если значение не указано, то режим отладки либо будет переключен на 1 (если он был равен 0), либо обнулен (если ранее был установлен ненулевой уровень отладки).

100

Включает режим трассировки клиента FTP. Трассировка будет выполняться до отключения режима отладки или завершения работы клиента FTP. После завершения трассировки может потребоваться значительное время на форматирование полученных данных.

Для включения трассировки при запуске клиента FTP создайте область данных QTMFTPD100 в библиотеке QTEMP с помощью следующей команды:

```
CRTDTAARA DTAARA(QTEMP/QTMFTPD100) TYPE(*LGL) AUT(*USE)
```

Если область данных QTMFTPD100 существует, то устанавливается уровень отладки, равный 100, и запускается трассировка клиента FTP. Это позволяет выполнять трассировку клиента в тех случаях, когда ее *нельзя* включить с помощью команды DEBUG 100.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

DEBUG (Изменить значения тайм-аутов клиента) **Команда клиента FTP**

Если значения тайм-аута, установленного по умолчанию, недостаточно для передачи данных, вы можете изменить его с помощью команды DEBUG. Значение тайм-аута следует изменять только в случаях, когда для передачи данных требуется очень много времени (из-за повышенной нагрузки на сеть или по другим причинам).

Для изменения значения тайм-аута клиента FTP выполните команду клиента FTP DEBUG:

```
DEBUg    T1 | T2    [ значение ]
```

T1

Изменить или показать тайм-аут чтения ответов сервера. Если сервер не вернет клиенту FTP ожидаемый ответ в течение указанного периода времени, то управляющее соединение с сервером будет закрыто. **T2**

Изменить или показать тайм-аут передачи данных. Если клиент FTP не получит в течение указанного периода времени ожидаемый ответ по информационному соединению, то информационное соединение будет закрыто.

значение

Значение тайм-аута в секундах. Допустимы положительные целые значения. Если значение тайм-аута не указано, команда выводит текущее значение тайм-аута.

Например:

```
DEBUG T1 900
```

Эта команда устанавливает для клиента тайм-аут ответа сервера, равный 900 секундам.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

DELETE (Удалить файл в удаленной системе) **Команда клиента FTP**

Команда DELETE позволяет удалить файл или элемент файла базы данных из удаленной системы. Ниже описан формат команды клиента FTP DELETE:

```
DELEte удаленный-файл
```

удаленный-файл

Файл, который нужно удалить из удаленной системы. Информация об указании имен файлов системы сервера iSeries приведена в разделе NAMEFMT (Выбор формата имен файлов).

Удаленная система может запросить информацию о правах на удаление файла. Для отправки такой информации вызовите команду ACCT (Отправить учетную запись).

Дополнительная информация приведена в следующих разделах:

- MDELETE (Удалить несколько файлов из удаленной системы)
- Команды клиента FTP
- Правила описания формата команд клиента FTP

DIR (Показать содержимое каталога, библиотеки или файлов) Команда клиента FTP

Команда клиента FTP DIR показывает список библиотек или каталогов удаленной системы и их содержимое. Ниже приведен формат команды Каталог (DIR):

```
Dir [ИМЯ] [(Disk)
```

ИМЯ

Имя каталога или библиотеки. По умолчанию команда выводит содержимое текущего каталога или библиотеки. Для изменения текущего каталога предназначена команда Сменить рабочий каталог (CD). Способ задания группы удаленных файлов зависит от конкретной системы. В большинстве систем для задания шаблона применяется символ звездочки *. Например, если роль удаленной системы играет сервер iSeries, то команда DIR MYLIB/MYFILE.* выведет список всех элементов файла MYFILE их библиотеки MYLIB.

Допустимы два формата имен файлов. В приведенном примере имя указано в формате NAMEFMT 0. Дополнительная информация об указании имен файлов при работе с FTP приведена в разделе NAMEFMT (Выбор формата имен файлов).

(Disk

Сохраняет результаты выполнения команды DIR в файле *CURLIB/DIROUTPUT.DIROUTPUT, не показывая их на экране.

Если в качестве удаленной системы выступает сервер iSeries, то в файл будет записана следующая информация:

- Для файлов баз данных - имена объектов *FILE и их элементов.
- Для файлов иерархической файловой системы (HFS):
 - Список папок и документов службы поиска документации (QDLS).
 - Список оптических томов (QOPT) и их содержимого (каталогов и файлов).

Будьте внимательны при выполнении команды DIR. Если вы укажете команду DIR без параметров, то будет показан список всех файлов текущего каталога. Его размер может оказаться значительно больше предполагаемого.

Для получения списка файлов, содержащихся в каталоге, вызовите команду Показать список (LS). Ее описание приведено в разделе LS (Показать список имен удаленных файлов)).

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

EBCDIC (Изменить тип передачи на EB CDIC) Команда клиента FTP

```
EBcdic
```

Тип передачи EBCDIC рекомендуется применять для обмена файлами между системами, поддерживающими EBCDIC. Тем самым можно будет избежать двойного преобразования из ASCII в EBCDIC и обратно.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

GET (Скопировать файл из удаленной системы в локальную)

Команда клиента FTP

```
Get удаленный-файл [локальный-файл]
[(Replace)]
```

удаленный-файл

Файл, который вы хотите получить из удаленной системы.

локальный-файл

Имя создаваемого элемента локального файла, документа или другого файла. Если имя локального файла не будет указано, то FTP задаст имя по умолчанию. Информация об именах по умолчанию приведена в разделе Имена файлов по умолчанию в командах передачи данных клиента.

(Replace

Указывает, что если локальный-файл существует, то его нужно заменить. По умолчанию сервер не заменяет локальный-файл.

Имя файла в команде GET должно быть задано в том формате, который поддерживается файловой системой, содержащей этот файл.

- Если файл находится не в библиотечной файловой системе (QSYS.LIB), то в команде GET должен применяться формат имен (NAMEFMT) 1:

```
GET /QDLS/QIWS0S2/PCSMENU.EXE
```

- Если файл находится в библиотечной файловой системе, то в команде GET должен применяться формат имен (NAMEFMT) 0:

```
GET YOURLIB/YOURFILE.YOURMBR (REPLACE
```

Если удаленный сервер - система iSeries, то эта команда скопирует элемент YOURMBR файла YOURFILE из библиотеки YOURLIB в элемент YOURMBR файла YOURFILE текущего каталога локальной системы. Информация о переходе в другой каталог приведена в разделе LCD (Изменить рабочую библиотеку или каталог локальной системы).

Примечание:

Если в имени удаленного файла содержатся апострофы, то имя файла должно быть также заключено в апострофы. Следующая команда копирует элемент 'MEMBER.ONE' с удаленного хоста. Дополнительная информация по этому вопросу приведена в разделе Ограничители параметров команды.

```
GET LIBRARY/FILE.MEMBER 'MEMBER.ONE'
```

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP
- MGET (Скопировать несколько файлов из удаленной системы в локальную)
- PUT (Отправить элемент локального файла в удаленную систему)

- MPUT (Отправить несколько элементов файлов из локальной системы в удаленную)

HELP (Справка по командам FTP)

Команда HELP выводит справочную информацию о командах FTP локальной и удаленной систем.

Справка по командам клиента FTP

Для получения информации о командах FTP, поддерживаемых локальной системой, введите команду HELP в следующем формате:

```
Help [* | ALL | команда ]
```

* или ALL

Выводит список команд клиента FTP.

команда

Выводит подробную информацию об указанной команде клиента. Например, команда HELP GET выдает справку о копировании файлов из удаленной системы в локальную. В команде HELP можно указывать сокращенные названия команд.

Если вы введете команду HELP без параметров, то будет показан список команд и перечень доступной справочной информации. Для того чтобы получить контекстную справку, поместите курсор на нужную команду в меню справки и нажмите клавишу **Enter**.

Для получения списка локальных команд сервера iSeries введите:

```
HELP
```

Справочную информацию можно получить и с помощью команды ?.

Справка по командам сервера FTP

Для того чтобы получить справку по командам FTP, которые поддерживаются удаленной системой, введите команду HELP в следующем формате:

```
Help SERVER [команда]
```

SERVER

Выдает справку по командам сервера FTP. Эта справка предоставляется удаленной системой. Эта команда равносильна команде QUOTE HELP. QUOTE HELP показывает список команд FTP, поддерживаемых удаленной системой.

команда

Имя команды сервера, информация о которой будет показана. Например, команда HELP SERVER STOR запросит у сервера справку по команде STOR.

Примечание: RHELP - синоним команды HELP SERVER. Например, команды HELP SERVER SITE и RHELP SITE эквивалентны.

Дополнительная информация по этому вопросу приведена в разделе QUOTE (Отправить команду на сервер FTP).

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

LCD (Изменить рабочую библиотеку или каталог в локальной системе)

Команда клиента FTP

LCd путь

каталог

Полное имя библиотеки, папки или каталога локальной системы.

Примечания:

1. Команда LCD не изменяет информацию о текущей библиотеке в списке библиотек.
2. При вызове команды CD (или LCD) для перехода в другую файловую систему нужно указать "корневой" каталог. Например, /QDLS или /QOPT.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

LOCSITE (Задать информацию о локальной системе)

Команда клиента FTP

Команда LOCSITE позволяет задать информацию, применяемую клиентом FTP для выполнения определенных операций в локальной системе:

LOCSITE [параметры]

Клиент FTP системы iSeries поддерживает следующие значения параметров команды LOCSITE:

CRTCCSID *CALC

В новых файлах базы данных, создаваемых при передаче файлов в формате ASCII, применяется CCSID EBCDIC по умолчанию, соответствующий CCSID передаваемого файла ASCII. Это значение по умолчанию.

CRTCCSID *USER

Для новых файлов базы данных, создаваемых при передаче файлов ASCII, применяется CCSID текущего задания. Но если этот CCSID равен 65535, то применяется CCSID по умолчанию, определяемый идентификатором языка, указанным в спецификации текущего задания.

CRTCCSID *SYSVAL

Для новых файлов базы данных, создаваемых при передаче файлов ASCII, применяется CCSID, определяемый системным значением QCCSID.

CRTCCSID [номер-CCSID]

Укажите CCSID, который вы хотите применять для файлов базы данных клиента, создаваемых при передаче файлов ASCII. Это значение будет подтверждено сервером.

CRTCCSID

Показывает сообщение с текущим значением CRTCCSID для клиента FTP.

TRIM 0

Выключает опцию Trim. Сервер передает записи базы данных вместе с конечными пробелами.

TRIM 1

Включает опцию Trim. При передаче файлов базы данных с сохранением файловой структуры в потоковом режиме конечные пробелы в записях базы данных не передаются. Это значение применяется по умолчанию.

TRIM 2

Устанавливает в опции Trim такое значение, при котором конечные пробелы записей базы данных не передаются в любом случае, в том числе при передаче структуры записей и в блочном режиме.

TRIM

Показывает сообщение с текущим значением параметра TRIM клиента FTP.

Примечания:

1. В предыдущих версиях, не поддерживающих эту команду, конечные пробелы в записях базы данных файловой системы QSYS.LIB всегда удалялись перед передачей файла на сервер.
2. Опция TRIM не поддерживается для файлов в двоичном формате (TYPE I). В этом случае пробелы никогда не передаются, независимо от значения параметра TRIM.

» DTAPROT C

Присвоить переменной защиты данных значение C (Clear). Эта переменная задает уровень защиты данных при открытии защищенного соединения для передачи управляющих данных. Дополнительная информация о защите данных приведена в описаниях следующих команд: SECDATA и SECOPEN.

DTAPROT P

Присвоить переменной защиты данных значение P (Private). Эта переменная задает уровень защиты данных при открытии защищенного соединения для передачи управляющих данных.

DTAPROT

Показывает сообщение с текущим значением переменной защиты данных. «

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

LOCSTAT (Показать информацию о состоянии локальной системы)

Команда клиента FTP

```
LOCStat
```

Выводит информацию о состоянии локальной системы, включающую следующие значения:

- Текущий режим SENDSITE
- Текущий режим SENDPORT
- Имя и номер порта удаленной системы, а также информация о входе в удаленную систему
- Тип данных и режим передачи
- Формат имен, применяемый в клиенте и на сервере
- Текущий режим VERBOSE
- Текущий режим DEBUG

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

LS (Показать список удаленных файлов) Команда клиента FTP

```
LS [ИМЯ] [(Disk]
```

ИМЯ

Удаленный каталог, файл или библиотека, содержимое которой нужно показать. Если роль удаленной системы играет сервер iSeries, то будет показан список имен файлов и их элементов. По умолчанию выдается полный список содержимого текущего каталога, библиотеки или папки. Для того чтобы перейти в другой каталог, библиотеку или папку, введите команду CD (дополнительная информация об этой команде приведена в разделе CD (Изменить рабочий каталог или библиотеку)). Формат описания удаленного файла зависит от типа системы.

(Disk

Сохраняет результаты выполнения команды LS в файле *CURLIB/LSOUTPUT.LSOUTPUT, не показывая их на экране. При каждом выполнении этой команды с параметром (Disk и одним и тем же значением *CURLIB сервер заменяет содержимое элемента LSOUTPUT.LSOUTPUT.

Примечание: Если сервер FTP вернет код отрицательного ответа (550), то элемент LSOUTPUT не будет создан. Если сервер FTP вернет код положительного ответа (150), но не передаст ни одного имени файла, то в элементе LSOUTPUT не будет записей.

Команда LS показывает только имена файлов. Информация о том, как просмотреть записи каталога с дополнительной информацией о файлах, приведена в разделе DIR (Показать содержимое каталога, библиотеки или файлов).

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

LTYPE (Локальный режим передачи) Команда клиента FTP

Для задания типа передачи файла (или способа представления данных при их копировании в локальную систему) предназначена команда клиента FTP LTYPE с форматом:

```
LType C ccsid#
```

C

Тип CCSID. Это значение обозначено символом C.

ccsid#

Номер CCSID. Допустимы значения от 1 до 65533.

Примечание: Команда LTYPE аналогична команде TYPE (дополнительная информация приведена в разделе TYPE (Задать тип передачи файлов)). Она изменяет представление данных только в системе клиента. Команда TYPE изменяет представление данных и в клиенте, и на сервере.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

**MDELETE (Удалить несколько файлов из удаленной системы)
Команда клиента FTP**

```
MDelete {удаленный-файл [удаленный-файл...]}
```

удаленный-файл

Один или несколько файлов сервера, которые необходимо удалить.

Примечание: При удалении файла системы iSeries из библиотеки QSYS.LIB будут удалены все элементы физического файла. Сам файл останется в библиотеке.

Ниже приведен пример удаления файлов из библиотечной файловой системы в случае, если NAMEFMT=0:

```
MDELETE MYLIB/FILE1.MBRA YOURLIB/FILE2.MBRB
```

Эта команда удаляет элемент MBRA из файла FILE1 библиотеки MYLIB и элемент MBRB из файла FILE2 библиотеки YOURLIB на удаленном сервере iSeries. Эта же команда для NAMEFMT 1:

```
MDELETE /QSYS.LIB/MYLIB.LIB/FILE1.FILE/MBRA.MBR  
/QSYS.LIB/YOURLIB.LIB/FILE2.FILE./MBRB.MBR
```

Ниже приведен пример удаления файлов из библиотеки документов, если NAMEFMT=1:

```
MDELETE /QDLS/QIWSOS2/PCSMENU.EXE /QDLS/PCSDIR/PCSFIL.EXE
```

Эта команда удаляет документ PCSMENU.EXE из папки QIWSOS2 в библиотеке средств работы с библиотеками документов и документ PCSFILE.EXE из папки PCSDIR в библиотеке QDLS системы iSeries.

С помощью символа звездочки (*) вы можете задать шаблон для удаления файлов. Например, для удаления файлов из системы iSeries, когда NAMEFMT=0, введите:

```
MDELETE MYLIB/MYFILE.*
```

Эта команда удалит все элементы файла MYFILE из библиотеки MYLIB. Звездочка должна быть последним символом в шаблоне имени.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

**MGET (Скопировать несколько файлов из удаленной системы в локальную)
Команда клиента FTP****Передача файлов командой MGET::**

Для копирования одного или нескольких удаленных файлов предназначена команда FTP MGET, формат которой указан ниже. Команда GET вызывается отдельно для передачи каждого удаленного файла. Имя соответствующего локального файла создается сервером автоматически согласно правилам присвоения имен по умолчанию.

Ниже описаны правила, по которым команда MGET выбирает целевой каталог для копирования файлов.

- Файлы всегда копируются в текущую библиотеку или каталог.
- Если ранее выполнялась команда LCD, то файлы копируются в библиотеку или каталог, указанные в этой команде.
- Если команда LCD предварительно не выполнялась, то текущий каталог определяется сервером следующим образом:
 - Если для пользовательского задания определена текущая библиотека, то текущим каталогом FTP считается она.
 - Если для пользовательского задания текущая библиотека не определена, то текущим каталогом считается библиотека QGPL.

```
MGet {удаленный-файл  
[удаленный-файл...]}{(Replace)}
```

удаленный-файл

Один или несколько файлов, которые нужно получить из удаленной системы.

(Replace

Указывает, что если некоторые из указанных файлов уже существуют в локальной системе, то их нужно заменить. Если опция Replace не указана, то существующие файлы не заменяются.

Имена локальных файлов задаются автоматически. Дополнительная информация приведена в описании команды GET GET (Скопировать файл из удаленной системы в локальную).

Для копирования всех элементов файла в текущую библиотеку или каталог укажите звездочку (*). Например, если роль удаленной системы играет сервер iSeries, то команда

- MGET MYLIB/MYFILE.* скопирует все элементы файла MYFILE из библиотеки MYLIB удаленной системы в текущую библиотеку локальной системы.
- Команда MGET /QSYS.LIB/MYLIB.LIB/MYFILE.FILE/*.MBR выполнит аналогичные действия, если NAMEFMT=1.
- Команда MGET /QOPT/PICTURES/IMAGES/*. * копирует все файлы из каталога IMAGES оптического тома PICTURES в текущую библиотеку (или каталог) локальной системы.
- Команда MGET TESTFILE.A* копирует все элементы файла TESTFILE, имя которых начинается с буквы A.
- Команда MGET /QDLS/QISSOS2/A* копирует все документы папки QISSOS2, имена которых начинаются с буквы A.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

MKDIR (Создать каталог)

Команда клиента FTP

```
MKdir каталог
```

каталог

Имя каталога, библиотеки или группы файлов в удаленной системе.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

MODE (Задать режим передачи данных) Команда клиента FTP

```
MODE [ B | S ]
```

B

Блочный режим. В этом режиме сервер передает данные как последовательность блоков, к каждому из которых добавляется заголовок из одного или нескольких байт. В блочном режиме можно передавать только данные в формате EBCDIC.

S

Потоковый режим. В этом режиме сервер передает данные в виде потока байт. В потоковом режиме вы можете выбрать любой тип представления данных.

Примечания:

1. По умолчанию FTP передает данные в потоковом режиме, так как некоторые системы не поддерживают блочный режим.
2. Если вы не укажете необязательный параметр, то клиент покажет текущее значение MODE.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

MPUT (Отправить несколько элементов файлов из локальной системы в удаленную) Команда клиента FTP

Команда клиента FTP MPUT служит для отправки одного или нескольких локальных файлов в удаленную систему. Клиент вызывает команду PUT для каждого передаваемого файла. Имя соответствующего удаленного файла создается согласно правилам присвоения имен по умолчанию.

```
MPut {локальный-файл [локальный-файл...]}
```

локальный-файл

Укажите один или несколько элементов файлов локальной библиотечной файловой системы или файлов других файловых систем, поддерживаемых FTP. Эти файлы будут переданы в удаленную систему. Имена удаленных файлов задаются автоматически.

Примечание:

Если удаленный файл уже существует, то его содержимое заменяется на содержимое *локального файла*, за исключением случая, когда включен режим SUNIQUE (дополнительная информация приведена в разделе SUNIQUE (Опция замены существующих файлов)).

Информация о том, как задавать имена файлов системы iSeries, приведена в разделе NAMEFMT (Выбор формата имен файлов). Следующий пример приведен для NAMEFMT 0:

```
MPUT MYLIB/FILE1.MBR1 MYLIB/FILE1.MBR2
```

Эта команда отправляет элементы MBR1 и MBR2 файла FILE1 в библиотеку MYLIB удаленной системы.

Следующий пример приведен для NAMEFMT 1:

```
MPUT /QDLS/QIWSOS2/PCSMENU.EXE /QDLS/QIWSOS2/PCSMENU2.EXE
```

Эта команда отправит в удаленную систему документы PCSMENU.EXE и PCSMENU2.EXE из папки QIWSOS2.

Для отправки всех элементов файла укажите звездочку (*). Например, команда MPUT MYLIB/MYFILE.* передает все элементы файла MYFILE из библиотеки MYLIB. Дополнительная информация приведена в разделе MGET (Скопировать несколько файлов из удаленной системы в локальную).

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP
- PUT (Отправить элемент локального файла в удаленную систему)

NAMEFMT (Выбрать формат имен файлов)

Команда клиента FTP

Команда клиента FTP NAMEFMT предназначена для выбора формата имен файлов в локальной системе, а также в удаленной системе (если это система iSeries).

```
NAMEfmt [ 0 | 1 ]
```

0

Этот формат имен допустим только для файлов базы данных библиотечной файловой системы. Общий вид имен следующий:

```
[имя-библиотеки/]имя-файла[.имя-элемента]
```

1

Этот формат имен допустим для всех файловых систем, поддерживаемых FTP (включая библиотечную файловую систему). Для работы со всеми файловыми системами iSeries должен применяться формат имен '1'.

Имена файлов библиотечной файловой системы в этом формате указываются в виде:

```
[/QSYS.LIB/][имя-библиотеки.LIB/]имя-файла.FILE[/имя-элемента.MBR]
```

Имена файлов сохранения задаются в виде:

```
/QSYS.LIB/библиотека.LIB/файл.SAVF
```

В файловой системе библиотек документов применяется следующий формат имен файлов:

```
[/QDLS/][{имя-папки[.расширение]/}]имя-файла[.расширение]
```

Имена файлов на оптических носителях задаются в следующем виде:

```
/QOPT/том/каталог/файл.расширение
```

Примечания:

1. Формат имен 0 допустим только при работе с библиотекой базы данных.
2. При вызове команды NAMEFMT без параметра будет показан текущий формат имен.

Дополнительная информация приведена в следующих разделах:

- Файловые системы и соглашения о присвоении имен, поддерживаемые FTP
- Команды клиента FTP
- Правила описания формата команд клиента FTP

NULLFLDS (Разрешить передачу файлов с пустыми полями) Команда клиента FTP

Эта команда позволяет указать, можно ли передавать из локальной системы в удаленную систему iSeries файлы базы данных с пустыми полями.

```
Nullfids [ 0 | 1 ]
```

Допустимы следующие значения параметра:

0

Запретить передачу файлов базы данных с пустыми полями. Это значение применяется по умолчанию.

1

Разрешить передачу файлов базы данных с пустыми полями.

Примечания:

1. Передача файлов с пустыми полями возможна только тогда, когда она разрешена одновременно и для клиента, и для сервера. Исходный файл должен копироваться в уже существующий целевой файл. Кроме того, определения целевого и исходного файла должны совпадать.
2. Если файл, содержащий пустые поля, копируется в систему, отличную от iSeries, либо если при передаче файла происходит преобразование кодовой страницы данных, то результат будет непредсказуемым.
3. При вызове команды NULLFLDS без параметра будет показано текущее значение параметра.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

OPEN (Подключиться к серверу FTP удаленной системы) Команда клиента FTP

```
Open имя-системы [номер-порта]
```

имя-системы

Имя или IP-адрес удаленной системы.

номер-порта

Номер порта, который будет применяться в данном сеансе до тех пор, пока соединение не будет закрыто. Это необязательный параметр. Если вы его не укажете, то порт будет выбран автоматически.

После подключения к одной удаленной системе для подключения к другой нужно сначала закрыть текущий сеанс.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

PASS (Отправить пароль)

Команда клиента FTP

```
PAss пароль
```

пароль

Строка, задающая ваш пароль.

Перед этой командой необходимо вызвать команды OPEN и USER. В некоторых системах этого достаточно для идентификации пользователя и настройки доступа. Если сервер автоматически запрашивает пароль при подключении, эту команду выполнять не обязательно.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

PUT (Скопировать элемент файла из локальной системы в удаленную)

Команда клиента FTP

```
PUt локальный-файл [удаленный-файл]
```

локальный-файл

Имя элемента файла библиотечной файловой системы, файла сохранения, документа или другого файла.

удаленный-файл

Имя файла в удаленной системе. Если вы не укажете имя локального файла, сервер FTP задал имя по умолчанию. Информация об именах по умолчанию приведена в разделе Имена файлов по умолчанию в командах передачи данных клиента. Если удаленный файл с таким именем уже существует, то сервер заменит его содержимое, за исключением случая, когда включен режим SUNIQUE (за дополнительной информацией обратитесь к разделу SUNIQUE (Опция замены существующих файлов)).

Для отправки файла в удаленную систему необходимы права на запись в текущий рабочий каталог удаленной системы.

Ниже приведен пример отправки элемента файла с помощью команды PUT:

```
PUT MYLIB/MYFILE.MYMBR (NAMEFMT = 0)
```

В удаленную систему будет отправлен элемент MYMBR файла MYFILE из библиотеки MYLIB.

Ниже приведен пример команды для отправки документа PCSMENU.EXE из папки QIWSOS2 файловой системы средств работы с библиотеками документов в удаленную систему.

```
PUT /QDLS/QIWSOS2/PCSMENU.EXE (NAMEFMT = 1)
```

Примечание:

Если в имени удаленного файла содержатся апострофы, то имя файла должно быть также заключено в апострофы. Следующая команда отправляет файл 'MEMBER.ONE' в удаленную систему. Дополнительная информация по этому вопросу приведена в разделе Ограничители параметров команды.

```
PUT LIBRARY/FILE.MEMBER 'MEMBER.ONE'
```

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP
- MPUT (Отправить несколько элементов файлов из локальной системы в удаленную)
- GET (Скопировать файл из удаленной системы в локальную)
- MGET (Скопировать несколько файлов из удаленной системы в локальную)

PWD (Показать текущий каталог, папку или библиотеку)**Команда клиента FTP**

Команда клиента FTP PWD показывает текущий каталог или библиотеку в удаленной системе:

```
PWd
```

Если роль удаленной системы играет сервер iSeries, то будет показана текущая библиотека или каталог файловой системы. Также будет показано имя рабочего каталога (в кавычках). Для того чтобы перейти в другой каталог или библиотеку, вызовите команду Изменить рабочий каталог (CD).

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

QUOTE (Отправить команду серверу FTP)**Команда клиента FTP**

```
QUOте строка
```

строка

Команда, которая будет отправлена на удаленный сервер FTP для выполнения. Сервер FTP передаст эту строку на удаленный сервер без изменений.

Примечания:

1. Специальную команду RCMD (Отправить команду CL в систему сервера FTP) сервера FTP системы iSeries можно выполнить только с помощью команды QUOTE. Например, чтобы записать протокол задания сервера в буферный файл, введите следующую команду:

```
QUOTE RCMD DSPJOBLOG
```

Команда WRKSPLF позволяет просмотреть протокол задания. Учтите, что если команда WRKSPLF запускается под управлением другого пользовательского профайла, то вы должны будете указать этот профайл.

2. Длина строки, передаваемой серверу FTP системы iSeries, не должна превышать 1000 символов.
3. Все, что указано в команде QUOTE, без изменений передается на сервер. Например, если ввести

```
QUOTE CWD 'SYS1'
```

Сервер получит

```
CWD 'SYS1'
```

Для получения справочной информации от сервера введите:

```
QUOTE HELP
```

На удаленный хост будет отправлена команда HELP, возвращающая список всех поддерживаемых команд. Формат списка зависит от типа удаленного хоста.

Учтите, что если вы выполните команду сервера с помощью команды QUOTE, то все действия по ее обработке будут выполнены только на сервере, тогда как при выполнении аналогичной команды клиента часть действий может быть выполнена на клиенте, а часть - на сервере. Например, команда REIN отправляет команду REIN на сервер и повторно инициализирует некоторые переменные состояния клиента. А команда QUOTE REIN отправляет на сервер команду REIN, но не изменяет переменные состояния клиента.

Примечание:

Будьте осторожны при вводе команд сервера в команде QUOTE. Как правило, она вызывается только в особых ситуациях, когда остальные команды клиента неприменимы. Примером может служить выполнение какой-либо особой команды сервера iSeries (CRTL и т.д.).

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

**REINITIALIZE (Повторно инициализировать сеанс)
Команда клиента FTP**

```
REInitialize
```

Если сервер поддерживает команду REINITIALIZE, то сеанс USER с сервером будет завершен. Сервер будет переведен в состояние, в котором он находился непосредственно после подключения, и вам потребуется повторно войти в систему.

Перед завершением сеанса USER будут полностью выполнены все текущие запросы на передачу файлов.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

RENAME (Переименовать файл удаленной системы)
Команда клиента FTP

```
REname исходное-имя новое-имя
```

исходное-имя

Имя файла удаленной системы.

новое-имя

Новое имя файла удаленной системы. Если файл с таким именем уже существует, то он будет заменен на указанный файл.

Следующая команда присваивает имя CAR.BMP файлу SPORTSCAR.BMP из каталога IMAGES оптического тома PICTURES:

```
REN /QOPT/PICTURES/IMAGES/SPORTSCAR.BMP  
/QOPT/PICTURES/IMAGES/CAR.BMP
```

Примечание: При переименовании файла системы iSeries нельзя изменять его файловую систему.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

RESET (Сброс)
Команда клиента FTP

Команда клиента FTP RESET очищает очередь ответов сервера:

```
REset
```

Эта команда применяется для синхронизации команд и ответов удаленного сервера FTP. Повторная синхронизация может потребоваться при нарушении удаленным сервером протокола FTP.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

RMDIR (Удалить каталог)
Команда клиента FTP

```
RMdir полное-имя
```

полное-имя

Имя каталога, библиотеки или группы файлов в удаленной системе. В иерархической файловой системе (HFS) можно удалять только пустые каталоги. Удаление библиотеки возможно в любом случае.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

SECData (Настройка защиты данных)

Команда клиента FTP

Для того чтобы задать уровень защиты для информационного соединения, когда с удаленной системой уже установлено защищенное управляющее соединение, вызовите команду SECData:

```
SECData [ C | P ]
```

Примечание:

SData является синонимом этой команды.

C

Будет установлен уровень защиты информационного соединения "прозрачное". Такое соединение **не** защищено. Такое соединение может применяться для передачи заранее зашифрованных данных или не конфиденциальных данных.

P

Будет установлен уровень защиты информационного соединения "частное". Такое соединение является защищенным. Перед передачей данных по соединению выполняется процедура согласования TLS.

1. Если параметр не задан, команда SECData выдает текущий уровень защиты данных.
2. Первоначально уровень защиты данных устанавливается равным значению параметра DTAPROT, заданного в команде CL STRTCPFTP при настройке защищенного управляющего соединения с сервером FTP.
3. Для применения команды SECData необходимо, чтобы было установлено защищенное управляющее соединение.
4. После того как уровень защиты данных изменяется с помощью команды SECData, вызывается команда сервера PROT.
5. Команда SECData отправляет команды PBSZ и PROT при настройке уровня защиты данных. Кроме того, команда SECData устанавливает переменную клиента после выполнения команды PROT. В этой переменной хранится уровень защиты данных (C или P), который был установлен на сервере последним. Эта переменная применяется для настройки уровня защиты данных во время открытия защищенного управляющего соединения с помощью команды SECOpen. Ее можно изменить с помощью опции LOCSITE DTAPROT (См. стр. 69).
6. Параметры 'C' и 'P' команды SECData совпадают с параметрами команды сервера PROT.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

SECOpen (Настройка уровня защиты данных)

Команда клиента FTP

Команда клиента FTP SECOpen открывает защищенное управляющее соединение с сервером FTP и устанавливает указанный уровень защиты. Ниже описан формат этой команды:

SECOpen имя-системы [номер-порта] [уровень-защиты]
--

Примечание: SOpen является синонимом SECOPEN.

имя-системы

Укажите имя или IP-адрес удаленной системы.

номер-порта

Введите номер порта, через который должно быть установлено соединение.

Примечания:

- Если этот параметр не указан, и задана опция (SSL, то используется порт 21.
- Если этот параметр не указан, и задана опция (IMPLICIT, то применяется порт 990.
- Если не указан ни номер порта, ни уровень защиты, то применяется порт 21 и уровень защиты (SSL.

уровень-защиты

Укажите способ защиты, который должен применяться.

(SSL

Для подключения к серверу FTP будет применяться защищенное соединение SSL. При установлении такого соединения вызывается команда сервера AUTH (Идентификация).

(IMPLICIT

Для подключения к серверу FTP будет установлено “неявное” защищенное соединение SSL/TLS. При настройке “неявного” соединения SSL на сервер не передаются команды AUTH, PBSZ и PROT. В этом случае сервер должен ожидать, что процедура согласования параметров соединения SSL или TLS будет выполняться через указанный номер порта.

В случае “неявного” соединения SSL сервер будет работать так, как будто клиент передал указанные команды со следующими параметрами:

- AUTH SSL
- PBSZ 0
- PROT P

Примечание:

Если уровень защиты не задан, применяется значение (SSL. Если номер порта равен 990, применяется значение (IMPLICIT.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

SENDPASV (Задать опцию передачи команды PASV)

Команда клиента FTP

Команда клиента FTP SENDPASV позволяет указать, нужно ли при передаче данных или выполнении команд DIR и LS отправлять на сервер FTP команду PASV.

SENDPASV [0 | 1]

Команда SENDPASV без параметра работает как переключатель. Значение SENDPASV, равное 1 (ВКЛ), изменяется на 0 (ВЫКЛ) и наоборот.

Допустимы следующие значения параметра команды:

0

Не отправлять команду PASV.

1

Отправлять команду PASV. Это значение применяется по умолчанию.

В системе iSeries по умолчанию опция передачи команды PASV включена. Если опция SENDPASV выключена, то сервер не будет передавать команду PASV.

Примечания:

1. Эта команда соответствует стандарту RFC 1579, "Firewall-Friendly FTP." Команду PASV рекомендуется применять для установления соединения через брандмауэр. В некоторых случаях передача данных без команды PASV невозможна.
2. Некоторые серверы FTP не поддерживают команду PASV. Если при работе с таким сервером будет включен режим SENDPASV, то клиент FTP выдаст сообщение о том, что сервер не поддерживает PASV. Система попытается установить информационное соединение без применения команды PASV.
3. Если режим SENDPASV выключен, то сервер передаст команду PORT в том случае, когда параметр SENDPORT равен ON. Дополнительная информация приведена в разделе SENDPORT (Задать опцию передачи команды PORT)
4. Серверы FTP, не поддерживающие PASV, не соответствуют требованиям RFC 1123.

Ограничения

При подключении к серверу FTP через сервер SOCKS команду SENDPASV можно выполнять только до вызова команд передачи данных и команд просмотра содержимого каталогов. В противном случае клиент не сможет установить информационное соединение с сервером FTP.

В этом случае для повторного вызова команды SENDPASV после выполнения команд передачи данных или просмотра каталога нужно прервать и вновь установить соединение с сервером FTP.

Команду SENDPASV можно вызывать после отключения клиента FTP от сервера FTP.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

SENDPORT (Задать опцию отправки команды PORT)

Команда клиента FTP

Эта команда клиента позволяет указать, нужно ли при передаче данных или выполнении команд DIR и LS отправлять на сервер FTP команду PORT. Формат вызова команды SENDPORT FTP:

SENDPOrt [0 | 1]

Команда SENDPORT без параметра работает как переключатель. Значение SENDPORT, равное 1 (ВКЛ), изменяется на 0 (ВЫКЛ) и наоборот.

Допустимы следующие значения параметра команды:

0

Не отправлять команду PORT.

1

Отправлять команду PORT. Это значение применяется по умолчанию.

Примечания:

1. Команду SENDPORT следует применять только в том случае, если без нее нельзя установить соединение с сервером. Необоснованное применение команды SENDPORT может привести к ошибкам.
2. Рекомендуется отключать отправку команды PORT при работе с системами, которые ее игнорируют, так как они ошибочно отправляют сообщение о том, что команда принята.
3. Если для SENDPASV указано ON (ВКЛ), то сервер не будет передавать команду PORT. Дополнительная информация приведена в разделе SENDPASV (Опция отправки команды PASV).

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

SENDSITE (Опция отправки команды SITE)

Команда клиента FTP

Команда клиента FTP SENDSITE позволяет указать, нужно ли при выполнении операций PUT и MPUT отправлять на сервер команду SITE с информацией о формате записей:

```
SENDSite [ 0 | 1 ]
```

Команда SENDSITE без параметра работает как переключатель. Значение SENDSITE, равное 0 (ВКЛ), изменяется на 1 (ВЫКЛ) и наоборот.

Допустимы следующие значения параметра команды:

0

Не отправлять команду SITE. Это значение применяется по умолчанию.

1

Отправлять команду SITE (содержащую информацию о формате записи) перед выполнением команд PUT и MPUT. Такое значение параметра команды SITE должно применяться при передаче файлов на сервер IBM Virtual Machine, который использует информацию о формате записи.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

SITE (Отправить информацию о локальном хосте)

Команда клиента FTP

Команда клиента FTP SITE предназначена для передачи информации, используемой удаленной системой для выполнения определенных операций:

Site [параметры]

параметры

Зависят от типа удаленной системы.

Информацию о допустимых параметрах и их формате можно получить с помощью команды HELP SERVER SITE. Некоторые серверы FTP не поддерживают команду SITE.

Примечание:

Команда SITE применяется командами PUT и MPUT для задания формата и длины записей. По умолчанию команда PUT автоматически выполняет команду SITE. Команда NAMEFMT выполняет команду SITE для определения формата имен, применяемого на сервере (NAMEFMT 0 или NAMEFMT 1).

За дополнительной информацией обратитесь к разделу SENDSITE (Задать опцию отправки команды SITE).

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

STATUS (Получить информацию о состоянии от удаленной системы)

Команда клиента FTP

STAtus [имя]

имя

Имя удаленного каталога или файла, информацию о состоянии которого нужно получить. Это необязательный параметр.

Примечание: Сервер FTP системы iSeries не поддерживает этот параметр.

Если параметр не указан, то будет предоставлена общая информация о состоянии сервера FTP. В том числе будут указаны значения всех параметров передачи данных и состояние соединений. Конкретная информация о состоянии зависит от версии сервера.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

STRUCT (Задать структуру файла)

Команда клиента FTP

Команда клиента FTP STRUCT позволяет задать структуру данных передаваемого файла. Формат вызова команды:

STRuct [F | R]

F Структура "файл". Файл представляет собой непрерывную последовательность байт данных.

R Структура "запись". Файл передается как последовательность записей.

От структуры файла зависит режим передачи и интерпретация данных.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

SUNIQUE (Опция замены файлов)

Команда клиента FTP

Команда клиента FTP SUNIQUE указывает, нужно ли заменять существующий целевой файл при выполнении команды PUT или MPUT:

SUnique [0 | 1]

Команда SUNIQUE без параметра работает как переключатель. Если значение SUNIQUE равно 0 (ВЫКЛ), то оно изменяется на 1 (ВКЛ), и наоборот.

Допустимы следующие значения параметра команды:

0

Указывает, что существующие файлы должны заменяться. Это значение применяется по умолчанию.

1

Не заменять файл в удаленной системе, а создавать новый с уникальным именем. При этом сервер FTP передает клиенту имя нового файла.

Примечание:

Если роль удаленной системы играет сервер iSeries, то имена создаваемых файлов в формате файл.элемент образуются путем добавления номера к имени *локального файла*, указанного в команде PUT или MPUT. Если файл с именем *файл.элемент* уже существует в удаленной системе, удаленный сервер iSeries создает файл с именем *файл.элемент1* и записывает в него данные.

Файлам других файловых систем, например, HFS, имена присваиваются по аналогичной схеме. Если файл уже существует, то создается новый файл, имя которого состоит из имени существующего файла и порядкового номера. Например, если в удаленной системе iSeries уже существует файл *xfsname*, то будет создан файл *xfsname1*.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

SYSCMD (Передать команду CL в локальную систему iSeries)

Команда клиента FTP

Для выполнения команды CL в локальной системе iSeries, не выходя из сеанса FTP, вызовите команду клиента FTP SYSCMD в следующем формате:

```
SYSCmd команда
```

команда

Команда CL системы iSeries. Если перед командой вы укажете вопросительный знак (?), то будет показано приглашение этой команды CL. Например, если ввести

```
SYSCMD ? SNDBRKMSG
```

то появится меню команды Отправить прерывающее сообщение (SNDBRKMSG).

Если вы хотите просмотреть низкоуровневые сообщения команды CL или выполнить несколько команд CL перед возвратом к сеансу FTP, воспользуйтесь командой CALL QCMD системы iSeries.

Например, для того чтобы перейти к меню Ввод команд iSeries, введите следующую команду:

```
SYSCMD CALL QCMD
```

После этого вы сможете запустить прикладную программу или выполнить команду CL из меню Ввод команды. После завершения прикладной программы или выполнения команды CL снова появится меню Ввод команды. Затем вы сможете просмотреть сообщения, выполнить другую команду или вернуться в сеанс FTP с помощью клавиши F3 (Выход) или F12 (Отмена).

Вы можете ввести команду CL, нажав F21 (Командная строка CL) в главном меню FTP. Клавиша F21 не действует, если в точку выхода Проверка запроса клиента FTP была добавлена программа выхода.

Примечания:

1. В большинстве серверов ограничено максимальное время простоя, по истечении которого сеанс автоматически завершается. Если на выполнение команды потребуется больше этого времени, то сервер прервет соединение с клиентом.
2. В системе iSeries вместо команды SYSCMD можно указывать восклицательный знак (!).
3. Команда SYSCMD передает указанную пользователем команду CL в систему iSeries без изменений.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

TYPE (Задать тип передачи файлов)

Команда клиента FTP

Для задания типа передачи файла (или способа представления данных при передаче) предназначена команда клиента FTP TYPE:

```
Type [ A
      B [ 1 | 2 | 3 [A|R] | 4 [A|R] | 5 | 6 | 7]
      C ccsid#
      E
      F [ 1 ]
      I ]
```

A

Задает тип передачи ASCII (значение по умолчанию). Эта команда равносильна команде ASCII. Форматирование файла по вертикали сервером не выполняется. В режиме ASCII сервер поддерживает только формат по умолчанию - NON PRINT. Этот тип передачи предназначен для обмена текстовыми файлами в тех случаях, когда хотя бы одна из систем не поддерживает тип EBCDIC.

По умолчанию для TYPE A (ASCII) применяется CCSID, указанный в параметре CCSID команды STRTCPFTP.

B

Канжи JIS с открывающим/закрывающим символами (CCSID 932)

B 1

Канжи JIS с открывающим/закрывающим символами (CCSID 932)

B 2

Расширенный набор символов Канжи для UNIX (CCSID 5050)

B 3

JIS 1983 с открывающей Esc-последовательностью ASCII (CCSID 5054)

B 3 A

JIS 1983 с открывающей Esc-последовательностью ASCII (CCSID 5054)

B 3 R

JIS 1983 с открывающей Esc-последовательностью JISROMAN (CCSID 5052)

B 4

JIS 1978 с открывающей Esc-последовательностью ASCII (CCSID 5055)

B 4 A

JIS 1978 с открывающей Esc-последовательностью ASCII (CCSID 5055)

B 4 R

JIS 1978 с открывающей Esc-последовательностью JISROMAN (CCSID 5053)

B 5

Хангул (CCSID 934)

B 6

Корейский стандартный набор символов KSC-5601, версия 1989 (CCSID 949)

B 7

Традиционный китайский (5550) (CCSID 938)

C номер-CCSID

Задает CCSID (идентификатор кодированного набора символов), поддерживаемый системой. Перед CCSID должен быть указан символ C.

E

Задает тип передачи EBCDIC. Эта команда равносильна команде EBCDIC. Форматирование файла по вертикали сервером не выполняется. В режиме EBCDIC поддерживается только формат по умолчанию - NON PRINT. Тип передачи EBCDIC предназначен для обмена файлами между системами с внутренним представлением символов EBCDIC.

F

Канжи IBM EBCDIC (CCSID 5035)

F 1

Канжи IBM EBCDIC (CCSID 5035)

I

Задаёт двоичный тип передачи. Эта команда равносильна команде BINARY. В двоичном режиме данные передаются как последовательность битов, сгруппированных по 8 в байты. Этот режим предназначен для эффективной передачи файлов и двоичных данных, например, исходного кода. В этом режиме преобразование данных не выполняется.

При вызове команды без параметров сервер показывает текущее значение TYPE.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

USER (Отправить ИД пользователя в удаленную систему)

Команда клиента FTP

User ИД [пароль]

ИД

Ваш ИД пользователя в удаленной системе.

пароль

Ваш пароль в удаленной системе. Пароль указывать не обязательно. Если он не будет указан в команде USER, то при необходимости появится приглашение на ввод пароля.

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

VERBOSE (Просмотр сообщений об ошибках)

Команда клиента FTP

Команда клиента FTP VERBOSE предназначена для просмотра ответов сервера FTP. Она работает как переключатель: включает и выключает режим VERBOSE. Когда этот режим включен, в окне клиента появляются все ответы сервера, включая их коды. Когда этот режим выключен, некоторые ответы и коды ответов сервера не выводятся.

Verbose

Перейти:

- Команды клиента FTP
- Правила описания формата команд клиента FTP

Программы выхода FTP

» Клиент и сервер FTP взаимодействуют с каждой программой выхода через соответствующую точку выхода. Она применяется для передачи параметров от сервера программе выхода. Формат передаваемой информации зависит от формата точки выхода.

Ниже перечислены точки выхода, применяемые FTP. Дополнительная информация, включая описание параметров и примеры кода, приведена в следующих разделах:

- Точка выхода проверки запроса: клиент и сервер
- Точка выхода из процедуры входа в систему сервера

Для обеспечения правильной работы программ выхода необходимо установить и зарегистрировать программы точек выхода. Если программы стали ненужными, вы можете удалить программы точек выхода.

Программы выхода и форматы точек выхода TCP/IP

Приведенная ниже таблица содержит информацию о точках выхода для различных приложений TCP/IP и соответствующих форматах.

Точки выхода TCP/IP	Приложение	VLRQ0100	TCPL0100	TCPL0200	TCPL0300
QIBM_QTMF_CLIENT_REQ	FTP	X			
QIBM_QTMF_SERVER_REQ	FTP	X			
QIBM_QTMF_SVR_LOGON ¹	FTP		X	X	X ²
QIBM_QTMX_SERVER_REQ	REXEC	X			
QIBM_QTMX_SVR_LOGON ¹	REXEC		X		X ²
QIBM_QTOD_SERVER_REQ	TFTP	X			

¹ - Существует несколько форматов точек выхода, однако программа выхода может быть зарегистрирована только для одного из них. Ознакомьтесь с информацией о различных форматах и выберите наиболее удобный из них.

² - Этот формат поддерживается начиная с версии V5R1.



Точка выхода для проверки запроса: клиент и сервер

Точки выхода для проверки запроса позволяют ограничить набор операций, которые разрешено выполнять пользователям FTP. Такие точки выхода поддерживаются как клиентом, так и сервером FTP. Для того чтобы ограничить доступ к клиенту и серверу FTP, необходимо добавить программы выхода для обеих точек выхода.

Примечание:

Поскольку точки выхода клиента и сервера FTP применяют одинаковый формат, можно создать одну программу для обработки обеих точек.

Если вы разрешили анонимным пользователям работать с сервером FTP, создайте программу выхода для проверки запросов к серверу FTP, которая разрешает анонимным пользователям FTP доступ только к командам FTP, но не к командам CL.

Ваша программа должна содержать следующие процедуры:

- Обработка исключительных ситуаций
- Отладка

- Запись сообщений в протокол

Разрешенные и запрещенные команды

Программа выхода для проверки запросов к серверу FTP может разрешить или запретить выполнение операции. Решение программы выхода учитывается вместе с результатами проверок, выполняемых приложением клиента и сервера FTP. Приложение клиента или сервера FTP вызывает программу выхода, зарегистрированную для этого приложения, при обработке следующих запросов:

1. Создание каталога/библиотеки
2. Удаление каталога/библиотеки
3. Изменение текущего каталога
4. Просмотр списка имен файлов
5. Удаление файла
6. Отправка файла
7. Получение файла
8. Переименование файла
9. Выполнение команды CL

Для того чтобы полностью запретить выполнение какой-либо команды, присвойте параметру 8 (Разрешить выполнение операции) формата точки выхода VRLQ0100 значение -1.

Предусмотрен ли тайм-аут для программы выхода?

Для программ выхода FTP тайм-аут не предусмотрен. Если при выполнении программы выхода происходит ошибка или возникает исключительная ситуация, которую программа не может обработать, сервер FTP завершает сеанс.

Примеры программ

В данном руководстве приведено несколько примеров программ, которые помогут вам настроить анонимный доступ к серверу FTP. Эти программы приведены только в качестве примера. В них не заданы конкретные значения для некоторых параметров, поэтому эти программы нельзя применять в рабочей системе. Вы можете использовать примеры в качестве основы для создания собственных программ. Кроме того, вы можете скопировать фрагменты кода примеров в свои программы. Примеры программ не рекомендуется запускать в рабочей системе.

Пример: Программа выхода из процедуры проверки запросов сервера или клиента FTP на языке CL

Пример: Программа выхода из процедуры проверки запросов к серверу FTP на языке ILE RPG

Отказ от гарантий на предоставленные примеры кода

Фирма IBM предоставляет вам неисключительную лицензию на применение всех примеров программ, на основе которых вы можете создать аналогичные функции, адаптированные к вашим требованиям.

Все фрагменты кода приведены фирмой IBM только в качестве примера. Они не прошли тщательного и всестороннего тестирования. В связи с этим фирма IBM не гарантирует и не подразумевает, что эти программы надежно работают и не содержат ошибок.

Все приведенные здесь программы предоставляются на условиях "КАК ЕСТЬ" без каких-либо гарантий. Фирма IBM отказывается от предоставления неявных гарантий соблюдения прав, коммерческой ценности и пригодности для какой-либо цели.

Пример: Программа выхода из процедуры проверки запросов к серверу или клиенту FTP на языке CL: Ниже приведен пример простой программы выхода, предназначенной для проверки запросов FTP. Она написана на управляющем языке (CL). Код этой программы неполон, но на его основе вы можете создать собственную программу для точки выхода клиента или сервера.

Примечание: Ознакомьтесь с важной юридической информацией, приведенной в разделе Отказ от гарантий на предоставляемый код.

(Отформатированный текст данного примера может не поместиться во фрейм.)

```

/*****/
/*                                          */
/*  Sample FTP server request validation exit program for anonymous FTP.  */
/*  Note: This program is a sample only and has NOT undergone any formal */
/*  review or testing.                                          */
/*                                          */
/*  Additional notes:                                          */
/*  1. When the application ID is 1 (FTP server) AND the operation ID is  */
/*  0 (session initialization), the job is running under the QTCP      */
/*  user profile when the exit program is called. In ALL other cases,  */
/*  the job is running under the user's profile.                    */
/*  2. It is highly recommended that the exit program be created in a library */
/*  with *PUBLIC authority set to *EXCLUDE, and the exit program itself */
/*  be given a *PUBLIC authority of *EXCLUDE. The FTP server adopts  */
/*  authority necessary to call the exit program.                    */
/*  3. It is possible to use the same exit program for both the FTP client */
/*  and server request validation exit points. However, this program  */
/*  does not take the client case into account.                      */
/*                                          */
/*****/

TSTREQCL:  PGM          PARM(&APPIDIN &OPIDIN &USRPRF &IPADDRIN +
                    &IPLLENIN &OPINFOIN &OPLLENIN &ALLOWOP)

/* Declare input parameters */
DCL      VAR(&APPIDIN)  TYPE(*CHAR) LEN(4) /* Application ID          */
DCL      VAR(&OPIDIN)  TYPE(*CHAR) LEN(4) /* Operation ID              */
DCL      VAR(&USRPRF)  TYPE(*CHAR) LEN(10) /* User profile              */
DCL      VAR(&IPADDRIN) TYPE(*CHAR)      /* Remote IP address        */
DCL      VAR(&IPLLENIN) TYPE(*CHAR) LEN(4) /* Length of IP address     */
DCL      VAR(&OPLLENIN) TYPE(*CHAR) LEN(4) /* Length of operation-specific info. */
DCL      VAR(&OPINFOIN) TYPE(*CHAR) +
                    LEN(9999) /* Operation-specific information */
DCL      VAR(&ALLOWOP) TYPE(*CHAR) LEN(4) /* allow (output) */

/* Declare local copies of parameters (in format usable by CL) */
DCL      VAR(&APPID)   TYPE(*DEC) LEN(1 0)
DCL      VAR(&OPID)   TYPE(*DEC) LEN(1 0)
DCL      VAR(&IPLLEN) TYPE(*DEC) LEN(5 0)
DCL      VAR(&IPADDR) TYPE(*CHAR)
DCL      VAR(&OPLLEN) TYPE(*DEC) LEN(5 0)
DCL      VAR(&OPINFO) TYPE(*CHAR) LEN(9999)
DCL      VAR(&PATHNAME) TYPE(*CHAR) LEN(9999) /* Uppercased path name */

/* Declare values for allow(1) and noallow(0) */
DCL      VAR(&ALLOW) TYPE(*DEC) LEN(1 0) VALUE(1)
DCL      VAR(&NOALLOW) TYPE(*DEC) LEN(1 0) VALUE(0)

/* Declare request control block for QLGCNVCS (convert case) API: */
/* convert to uppercase based on job CCSID */
DCL      VAR(&CASEREQ) TYPE(*CHAR) LEN(22) +
                    VALUE(X'0000000100000000000000000000000000000000+
                    00000000')
DCL      VAR(&ERROR) TYPE(*CHAR) LEN(4) +
                    VALUE(X'00000000')
```

```

/* Assign input parameters to local copies */
CHGVAR    VAR(&APPID) VALUE(%BINARY(&APPIDIN))
CHGVAR    VAR(&OPID) VALUE(%BINARY(&OPIDIN))
CHGVAR    VAR(&IPLN) VALUE(%BINARY(&IPLNIN))
CHGVAR    VAR(&IPADDR) VALUE(%SUBSTRING(&IPADDRIN 1 &IPLN))
CHGVAR    VAR(&OPLN) VALUE(%BINARY(&OPLNIN))

/* Handle operation specific info field (which is variable length) */
IF        COND(&OPLN = 0) THEN(CHGVAR VAR(&OPINFO) +
        VALUE(' '))
ELSE      CMD(CHGVAR VAR(&OPINFO) VALUE(%SST(&OPINFOIN +
        1 &OPLN)))

/* Operation id 0 (incoming connection): reject if connection is coming */
/* through interface 9.8.7.6, accept otherwise. (The address is just an */
/* example.) This capability could be used to only allow incoming connections */
/* from an internal network and reject them from the "real" Internet, if */
/* the connection to the Internet were through a separate IP interface. */
/* NOTE: For FTP server, operation 0 is ALWAYS under QTCP profile. */
IF        COND(&OPID = 0) THEN(DO)
  IF      COND(&OPINFO = '9.8.7.6') THEN(CHGVAR +
        VAR(%BINARY(&ALLOWOP)) VALUE(&NOALLOW))
  ELSE    CMD(CHGVAR VAR(%BINARY(&ALLOWOP)) +
        VALUE(&ALLOW))
  GOTO    CMDLBL(END)
ENDDO

/* Check for ANONYMOUS user */
IF        COND(&USRPRF = 'ANONYMOUS ') THEN(DO)
/* Don't allow the following operations for ANONYMOUS user: */
/* 1 (Directory/library creation); 2 (Directory/library deletion); */
/* 5 (File deletion); 7 (Receive file); 8 (Rename file); 9 (Execute CL cmd) */
IF        COND(&OPID = 1 | &OPID = 2 | +
        &OPID = 5 | &OPID = 7 | &OPID = 8 | +
        &OPID = 9) THEN(CHGVAR +
        VAR(%BINARY(&ALLOWOP)) VALUE(&NOALLOW))
ELSE      CMD(DO)
/* For operations 3 (change directory), 4 (list directory) and 6 (send file), */
/* only allow if in PUBLIC library OR "/public" directory. Note that all */
/* path names use the Integrated File System naming format. */
IF        COND(&OPID = 3 | &OPID = 4 | &OPID = 6) THEN(DO)
/* First, convert path name to uppercase (since names in "root" and library */
/* file systems are not case sensitive). */
CALL PGM(QLGCNVCS) PARM(&CASEREQ &OPINFO &PATHNAME +
        &OPLNIN &ERROR)
/* Note: must check for "/public" directory by itself and path names starting */
/* with "/public/". */
IF        COND((%SUBSTRING(&PATHNAME 1 20) *NE +
        '/QSYS.LIB/PUBLIC.LIB') *AND +
        (&PATHNAME *NE '/PUBLIC') *AND +
        (%SUBSTRING(&PATHNAME 1 8) *NE '/PUBLIC/')) +
        THEN(CHGVAR +
        VAR(%BINARY(&ALLOWOP)) VALUE(&NOALLOW))
ELSE      CMD(CHGVAR VAR(%BINARY(&ALLOWOP)) +
        VALUE(&ALLOW))
ENDDO
ENDDO
ENDDO
/* Not ANONYMOUS user: allow everything */
ELSE      CMD(CHGVAR VAR(%BINARY(&ALLOWOP)) +
        VALUE(&ALLOW))

END:      ENDPGM

```

Пример: Программа выхода из процедуры проверки запросов к серверу FTP на языке ILE

RPG: Ниже приведен пример простой программы выхода, предназначенной для проверки запросов к серверу FTP. Она написана на языке ILE RPG. Код этой программы неполон, но на его основе вы можете создать собственную программу.

Примечание: Ознакомьтесь с важной юридической информацией, приведенной в разделе Отказ от гарантий на предоставляемый код.

(Отформатированный текст данного примера может не поместиться во фрейм.)

```
* Module Description *****
*
*           PROGRAM FUNCTION
*
* This program demonstrates some of the abilities an FTP Client
* and Server Request Validation Exit Program can have.
*
* Note: This program is a sample only and has NOT undergone any
*       formal review or testing.
*
*****
F/SPACE 3
*****
*
*           INDICATOR USAGE
*
* IND.  DESCRIPTION
*
* LR - CLOSE FILES ON EXIT
*
*****
F/EJECT
*****
* DATA STRUCTURES USED BY THIS PROGRAM
*****
*
* Define constants
*
D Anonym          C          CONST('ANONYMOUS ')
D PublicLib       C          CONST('/QSYS.LIB/ITSOIC400.LIB')
D PublicDir       C          CONST('//ITSOIC.400')
*
* Some CL commands to used later on in the program
*
D ClearSavf       C          CONST('CLRSAVF ITS0IC400/TURVIS')
D SaveLib         C          CONST('SAVLIB LIB(ITSOIC400) -
D                               DEV(*SAVF) -
D                               SAVF(ITSOIC400/TURVIS)')
*
* A value to be used to trigger a benevolent 'Trojan Horse'
*
D Savetti         C          CONST('ITSOIC400.LIB/TURVIS.FILE')  Extension is FILE
*                                                                although it is a
*                                                                SAVF (and entered as
*                                                                SAVF by the user)
*
* Some nice fields to help us through from lower to upper case character conversion
* 1
D LW              C          CONST('abcdefghijklmnopqrstuvwxy')
D UP              C          CONST('ABCDEFGHIJKLMNPNQRSTUVWXYZ')
*
D NeverAllow      C          CONST(-1)
D DontAllow       C          CONST(0)
D Allow           C          CONST(1)
D AlwaysAllw      C          CONST(2)
C/EJECT
```

```

*****
* VARIABLE DEFINITIONS AND LISTS USED BY THIS PROGRAM
*****
C/SPACE 2
*
* Define binary parameters
*
D          DS
D APPIDds      1      4B 0
D OPIDds       5      8B 0
D IPLEndds     9     12B 0
D OPLEndds    13     16B 0
D ALLOWOPds   17     20B 0
*
C *LIKE      DEFINE  APPIDds  APPIDIN
C *LIKE      DEFINE  OPIDds   OPIDIN
C *LIKE      DEFINE  IPLEndds IPLENIN
C *LIKE      DEFINE  OPLEndds OPLENIN
C *LIKE      DEFINE  ALLOWOPds ALLOWOP
*
C *LIKE      DEFINE  OPINFOIN  OPINFO
*
* Define parameter list
*
C *Entry      PLIST
* Input parameters:
C          PARM          APPIDIN          Application ID
*                                     possible values: 0 = FTP Client Program
*                                                         1 = FTP Server Program
C          PARM          OPIDIN          Operation ID
*                                     possible values: 0 = Initialize Session
*                                                         1 = Create Dir/Lib
*                                                         2 = Delete Dir/Lib
*                                                         3 = Set Current Dir
*                                                         4 = List Dir/Lib
*                                                         5 = Delete Files
*                                                         6 = Send Files
*                                                         7 = Receive Files
*                                                         8 = Rename Files
*                                                         9 = Execute CL cmd
C          PARM          USRPRF          10          User Profile
C          PARM          IPADDRIN       15          Remote IP Address
C          PARM          IPLENIN        Length of IP Address
C          PARM          OPINFOIN       999         Operation-spec. Info
C          PARM          OPLENIN        Length of Oper. Spec
* Return parameter:
C          PARM          ALLOWOP        Allow Operation (Out
*                                     possible values: -1 = Never Allow
*                                                         (And don't bother
*                                                         me with this ops
*                                                         in this session)
*                                                         0 = Reject Operation
*                                                         1 = Allow Operation
*                                                         2 = Always Allow Oper.
*                                                         (And don't bother
*                                                         me with this ops
*                                                         in this session)
C/EJECT
*****
* The Main Program
*****
*
C          SELECT
C APPIDIN   WHENEQ   0
C          EXSR    ClientRqs
C APPIDIN   WHENEQ   1
C          EXSR    ServerRqs

```

```

C          ENDSL
*
C          EVAL      *INLR = *ON
C          RETURN
C/EJECT
*****
* S U B R O U T I N E S *
*****
* Here we handle all the FTP Client request validation *
*****
C      ClientRqs      BEGSR
*
* Check user profile
*
C          SELECT
*
* Check for 'bad' users who are not allowed to do anything ever
*
C      USRPRF          WHENEQ      'JOEBAD      '
*
C          Z-ADD      NeverAllow      ALLOWOP          Ops not allowed
*
* Check for 'normal' users who are not allowed to do some things
*
C      USRPRF          WHENEQ      'JOENORMAL '
*
C          SELECT
*
C      OPIDIN          WHENEQ      0          New Connection
C          Z-ADD      Allow          ALLOWOP
*
C      OPIDIN          WHENEQ      1          Create Directory/Lib
C      OPIDIN          OREQ        2          Delete Directory/Lib
C      OPIDIN          OREQ        5          Delete Files
C      OPIDIN          OREQ        7          Receive Files from S
C      OPIDIN          OREQ        8          Rename files
C      OPIDIN          OREQ        9          Execute CL Commands
*
C          Z-ADD      NeverAllow      ALLOWOP          Ops never allowed
*
C      OPIDIN          WHENEQ      3          Set Current Dir
C      OPIDIN          OREQ        4          List Directory/Lib
C      OPIDIN          OREQ        6          Send Files to Server
*
* Extract library and directory names for comparison with allowed areas
*
C      OPLENIN          IFGE        11
C      11              SUBST        OPINFOIN:1      Directory      11
C          ELSE
C      OPLENIN          SUBST(P)    OPINFOIN:1      Directory
C          ENDIF
C 1 LW:UP              XLATE        Directory      Directory
*
C      OPLENIN          IFGE        23
C      23              SUBST        OPINFOIN:1      Library      23
C          ELSE
C      OPLENIN          SUBST(P)    OPINFOIN:1      Library
C          ENDIF
*
C      Directory      IFEQ        PublicDir          Allowed Directory
C      Library        OREQ        PublicLib          or Library
C          Z-ADD      Allow          ALLOWOP
C          ELSE
C          Z-ADD      DontAllow      ALLOWOP
C          ENDIF
*

```

```

C          OTHER
C          Z-ADD      DontAllow    ALLOWOP
C          ENDSL
*
* Check for 'cool' users who are allowed to do everything
*
C  USRPRF      WHENEQ    'JOEGOOD  '
C  USRPRF      OREQ     'A960101B '
C  USRPRF      OREQ     'A960101C '
C  USRPRF      OREQ     'A960101D '
C  USRPRF      OREQ     'A960101E '
C  USRPRF      OREQ     'A960101F '
C  USRPRF      OREQ     'A960101Z '
* Allow All FTP Operations
C          Z-ADD      AlwaysAllw    ALLOWOP
*
2 * Any Other User: We leave the back door open and allow
* all operations. If you want to use this program for securing
* your system, then close this door!
*
C          OTHER
C          Z-ADD      AlwaysAllw    ALLOWOP
C*****      Z-ADD      NeverAllow  ALLOWOP
C          ENDSL
*
C          ENDSR
C/EJECT
*****
* Here we handle all the FTP Server request validation
*****
C  ServerRqs    BEGSR
*
* Check for ANONYMOUS user
*
C  USRPRF      IFEQ     Anonym
*
C          SELECT
*
C  OPIDIN      WHENEQ    1          Create Directory/Lib
C  OPIDIN      OREQ     2          Delete Directory/Lib
C  OPIDIN      OREQ     5          Delete Files
C  OPIDIN      OREQ     7          Receive Files from C
C  OPIDIN      OREQ     8          Rename files
C  OPIDIN      OREQ     9          Execute CL Commands
*
C          Z-ADD      NeverAllow    ALLOWOP          Ops never allowed
*
C  OPIDIN      WHENEQ    3          Set Current Dir
C  OPIDIN      OREQ     4          List Directory/Lib
C  OPIDIN      OREQ     6          Send Files to Client
*
* Extract library and directory names for comparison with allowed areas
*
C  OPLENIN      IFGE     11
C  11          SUBST    OPINFOIN:1  Directory    11
C          ELSE
C  OPLENIN      SUBST(P) OPINFOIN:1  Directory
C          ENDIF
C 1 LW:UP      XLATE    Directory    Directory
*
C  OPLENIN      IFGE     23
C  23          SUBST    OPINFOIN:1  Library      23
C          ELSE
C  OPLENIN      SUBST(P) OPINFOIN:1  Library
C          ENDIF
*
C  Directory    IFEQ     PublicDir          Allowed Directory

```



```

C      Library      OREQ      PublicLib      or Library
C      Z-ADD      Allow      ALLOWOP
C      ELSE
C      Z-ADD      DontAllow      ALLOWOP
C      ENDIF
*
C      OTHER
C      Z-ADD      DontAllow      ALLOWOP
C      ENDSL
*
C      ELSE
*
* Any Other User: Allow All FTP Operations
*
C      OPIDIN      IFEQ      6      Send Files to Client
*
* If client issued GET for save file HESSU in library HESSU then we refresh the contents
*
*
C      LW:UP      XLATE      OPINFOIN      OPINFO
C      Z-ADD      0      i      3 0
C      Savetti      SCAN      OPINFO:1      i
*
C      i      IFGT      0
*
* We assume that the save file exists and here clear the save file
*
C      MOVE(L(p) ClearSavf      Cmd      80
C      Z-ADD      19      Len      15 5
C      CALL      'QCMDEXC'      9999
C      PARM
C      PARM      Cmd
C      PARM      Len
*
* and here we save the library to the save file
*
C      MOVE(L(p) SaveLib      Cmd
C      Z-ADD      46      Len
C      CALL      'QCMDEXC'      9999
C      PARM
C      PARM      Cmd
C      PARM      Len
C      ENDIF
C      ENDIF
*
C      Z-ADD      Allow      ALLOWOP
C      ENDIF
*
C      ENDSR

```

Формат точки выхода VLRQ0100: Точка выхода для проверки запросов к приложению сервера FTP:

QIBM_QTMF_SERVER_REQ

Точка выхода для проверки запросов к приложению клиента FTP:

QIBM_QTMF_CLIENT_REQ

Для управления форматом параметров этой точки выхода применяется следующий интерфейс:

VLRQ0100

В приведенной ниже таблице описаны параметры интерфейса VLRQ0100 и их формат.

Формат параметров интерфейса точки выхода VLRQ0100

Параметр	Описание	Входной или выходной	Тип и длина
1	Идентификатор приложения	Входной	Binary (4)
2	Идентификатор операции	Входной	Binary (4)
3	Пользовательский профайл	Входной	Char (10)
4	Удаленный IP-адрес	Входной	Char (10)
5	Длина удаленного IP-адреса	Входной	Binary (4)
6	Информация об операции	Входной	Char (*)
7	Длина информации об операции	Входной	Binary (4)
8	Разрешить операцию	Выходной	Binary (4)

Описание параметров:

Параметр 1 VLRQ0100:

Идентификатор приложения

Входной; BINARY(4)

Указывает, какая прикладная программа TCP/IP отправила запрос. Интерфейс VLRQ0100 используется четырьмя приложениями TCP/IP. Первый параметр указывает, какое именно приложение вызвало программу выхода. Возможны следующие значения:

0	Клиент FTP
1	Сервер FTP
2	Сервер REXEC
3	Сервер TFTP

Параметр 2 VLRQ0100:

Идентификатор операции

Входной; Binary(4)

Указывает операцию (команду), запрос на выполнение которой поступил от пользователя FTP.

Если идентификатор приложения (1 параметр) соответствует клиенту или серверу FTP, то допускаются следующие значения:

Значение	ИД операции	Команда клиента	Команда сервера
0	Запустить сеанс	Открыть, SECOpen	Новое соединение
1	Создать каталог/библиотеку	*	MKD, XMDK
2	Удалить каталог/библиотеку	*	RMD, XRMD
3	Задать текущий каталог/библиотеку	LCD	CWD, CDUP, XCWD, XCUP
4	Показать список файлов	*	LIST, NLIST
5	Удалить файл	*	DELE
6	Отправить файл	APPEND, PUT, MPUT	RETR
7	Получить файл	GET, MGET	APPE, STOR, STOU
8	Переименовать файл	*	RNFR, RNT0
9	Выполнить команду CL	SYSCMD	RCMD, ADDm, ADDV, CRTL, CRTP, CRTS, DLTF, DLTL

Примечание: Символом * помечены управляющие операции, которые не поддерживаются клиентом FTP. Клиент может выполнить эти операции, только указав соответствующую команду CL в команде FTP SYSCMD. Командам CL соответствует идентификатор 9.

Параметр 3 VLRQ0100:

Пользовательский профайл

Входной; Char(10)

Пользовательский профайл для сеанса FTP.

Параметр 4 VLRQ0100:

Удаленный IP-адрес

Входной; CHAR(*)

IP-адрес удаленного хоста. Значение задается в десятичном формате с точками (123.45.67.89) и выравнивается по левому краю. Удаленный хост может быть сервером или клиентом, в зависимости от указанного идентификатора приложения.

Параметр 5 VLRQ0100:

Длина удаленного IP-адреса (параметр 4) в байтах.

Входной; BINARY(4)

Длина удаленного IP-адреса (параметр 4).

Параметр 6 VLRQ0100:

Информация об операции

Входной; CHAR(*)

Описание запрошенной операции. Содержимое этого поля зависит от значения идентификатора операции (параметр 2) и идентификатора приложения (параметр 1). Например:

Для операции 0 и приложения 0

Информация об операции не задается. В этом случае данное поле будет пустым.

Для операции 0 и приложения 1

Информация об операции содержит IP-адрес интерфейса TCP/IP, применяемого для подключения к локальному хосту (серверу FTP). Значение задается в десятичном формате с точками (123.45.67.89) и выравнивается по левому краю.

Для операций 1-3

В данном поле указывается имя каталога или библиотеки, для которой должна быть выполнена операция. При этом задается полный путь к каталогу или библиотеке.

Для операций 4-8

В данном поле содержится имя файла, над которым выполняется операция. При этом задается полный путь к файлу.

Для операции 9

Информация об операции содержит команду CL, выполнение которой запрошено пользователем.

Дополнительная информация приведена в разделе Примечания к информации о формате точки выхода VLRQ0100.

Параметр 7 VLRQ0100:

Длина информации об операции

Входной; BINARY(4)

Задаёт длину строки, содержащей информацию об операции (параметр 6). Если информация об операции не задана, то в этом поле будет указано значение 0.

Параметр 8 VLRQ0100:

Разрешить выполнение операции

Выходной; BINARY(4)

Указывает, разрешать ли выполнение операции. Возможны следующие значения:

-1	<p><i>Никогда</i> не разрешать выполнение операции с таким идентификатором:</p> <p>Запретить выполнение операции с данным идентификатором до конца сеанса.</p> <p>Операция с таким идентификатором не будет повторно вызывать программу выхода.</p>
0	Запретить операцию
1	Разрешить операцию
2	<p><i>Всегда</i> разрешать выполнение операции с таким идентификатором:</p> <p>Разрешить выполнение операции с данным идентификатором до конца сеанса.</p> <p>Операция с таким идентификатором не будет повторно вызывать программу выхода.</p>

Замечания о работе с форматом VLRQ0100: Формат VLRQ0100 соответствует как точке выхода для проверки запросов к клиенту FTP, так и точке выхода для проверки запросов к серверу FTP.

Недопустимые выходные параметры

Если значение параметра 8 (Разрешить операцию) недопустимо, то сервер FTP отклоняет запрошенную операцию, а в протокол задания заносится следующее сообщение:

Данные программы выхода для точки выхода &1 отсутствуют или недопустимы

Исключительные ситуации

Если при вызове программы выхода возникнет исключительная ситуация, то в протокол задания будет занесено следующее сообщение:

В программе выхода FTP &1 из библиотеки &2 для точки выхода &3 возникла исключительная ситуация

Обзор: Информация об операции

В приведенной ниже таблице указана информация, которая должна быть задана в параметре 6 формата VLRQ0100 для различных идентификаторов операций (параметр 2 формата VLRQ0100).

Идентификатор операции (параметр 2 VLRQ0100)	Информация об операции (параметр 6 VLRQ0100)
0	Нет, если ИД приложения равен 0 (параметр 1)
0	IP-адрес клиента в десятичном формате с точками, если ИД приложения равен 1 или 2 (параметр 1)
1-3	Полное имя библиотеки или каталога. Например: /QSYS.LIB/QGPL.LIB ^(a) /QOpenSys/DirA/DirAB/DirABC ^(b)
4-8	Полное имя файла. Например: /QSYS.LIB/MYLIB.LIB/MYFILE.FILE/MYMEMB.MBR ^(a) /QOpenSys/DirA/DirAB/DirABC/FileA1 ^(b)

Примечания: ^(a) - Имена объектов файловой системы QSYS.LIB всегда задаются прописными буквами

^(b) - Имена объектов файловой системы QOpenSys должны быть заданы с учетом регистра символов.

Точка выхода из процедуры входа в систему сервера

Точка выхода из процедуры входа в систему сервера приложения TCP/IP предназначена для идентификации пользователей. Эта точка выхода предоставляет доступ к серверу на основе адреса инициатора сеанса. Она также позволяет задать начальный рабочий каталог, отличный от каталога, указанного в пользовательском профайле.

Зарегистрированная для этой точки выхода программа выхода вызывается при каждой попытке входа в систему для работы с сервером. Программа выхода выдает код возврата, указывающий, должен ли сервер продолжать обработку запроса на вход пользователя в систему. Кроме того, существуют дополнительные значения кода возврата, указывающие способ обработки запроса на вход в систему и задающие информацию о рабочем каталоге по умолчанию.

Точка выхода из процедуры входа в систему сервера FTP системы iSeries:

QIBM_QTMF_SVR_LOGON

Существует три формата точек выхода:

- Формат точки выхода TCPL0100 предоставляет следующие параметры для управления входом в систему:
 - Для разрешения или отклонения запроса на вход в систему
 - Для задания имени пользователя, пароля и текущей библиотеки
- Формат точки выхода TCPL0200 предоставляет следующие дополнительные параметры для управления входом в систему:
 - Для задания рабочего каталога.
 - Для возврата информации о приложении
 - Для настройки опции шифрования данных, которыми обмениваются клиент и сервер FTP.
- Помимо параметров, предусмотренных в формате TCPL0200, формат точки выхода TCPL0300 позволяет применять расширенную поддержку паролей OS/400 и предоставляет дополнительные параметры, с помощью которых можно задать CCSID пароля и имени каталога. Кроме того, если пользователь был идентифицирован с помощью сертификата клиента, то в этом случае сертификат передается программе выхода.

Примечания:

1. Для точки выхода из процедуры входа в систему сервера FTP можно зарегистрировать только одну программу выхода. Выберите тот формат точки выхода, который вам больше подходит.
2. Для приложения FTP эта точка выхода позволяет реализовать анонимный доступ к FTP и получить информацию, необходимую для регистрации и управления доступом.
3. Все символьные значения параметров в форматах точек выхода TCPL0100 и TCPL0200, а также все символьные значения параметров в формате TCPL0300, для которых не задан CCSID, передаются программе выхода в кодировке, заданной для задания. Если CCSID задания равен 65535, то для символьных данных применяется CCSID по умолчанию. Все символьные значения параметров, возвращаемые программой выхода, должны быть в том же CCSID.

Для поддержки анонимного доступа к серверу FTP создайте программу выхода из процедуры входа в систему сервера FTP, которая выполняет следующие действия:

1. Разрешает анонимным пользователям вход в систему
2. Отправляет приглашение на ввод адреса электронной почты в качестве пароля. Обычно требуется ввести 'правильный адрес электронной почты'. Этот термин может ввести в заблуждение, так как программа выхода проверяет только то, есть ли в середине буквенно-цифровой строки символ '@'. Это также обычная практика. Именно поэтому нужно заносить в протокол IP-адреса пользователей.
3. Проверяет, присутствует ли в строке пароля символ @.
4. Если пользователь указал ИД ANONYMOUS, то ему должен быть предоставлен доступ только к указанной общей библиотеке. Ознакомьтесь с описанием кода возврата 3 параметра 8 формата TCPL0200.

Ниже перечислены процедуры, которые должна содержать ваша программа:

- Обработка исключительных ситуаций
- Отладка
- Ведение протокола
 - Запись в протокол IP-адреса и адреса электронной почты (пароля) пользователя, подключающегося к серверу FTP.

Предусмотрен ли тайм-аут для программы выхода?

Для программ выхода FTP тайм-аут не предусмотрен. Если при выполнении программы выхода происходит ошибка или возникает исключительная ситуация, которую программа не может обработать, сервер FTP завершает сеанс.

Права доступа, необходимые профайлу QTCP

При вызове программы выхода для процедуры входа в систему сервера FTP задание сервера FTP работает под управлением пользовательского профайла QTCP.

Убедитесь, что у профайла QTCP есть права доступа, необходимые для записи информации в файлы протокола и другие файлы, применяемые программами выхода.

Примеры программ

В данном руководстве приведено несколько примеров программ, которые помогут вам настроить анонимный доступ к серверу FTP. Эти программы приведены в качестве примера. В них не заданы конкретные значения для некоторых параметров, поэтому эти программы нельзя применять в рабочей системе. Вы можете использовать примеры в качестве основы для создания собственных программ. Кроме того, вы можете скопировать фрагменты кода примеров в свои программы. Примеры программ не рекомендуется запускать в рабочей системе.

Пример: Программа выхода из процедуры входа в систему сервера FTP на языке CL
 Пример: Программа выхода из процедуры входа в систему сервера FTP на языке C
 Пример: Программа выхода из процедуры входа в систему сервера FTP на языке RPG

Отказ от гарантий на предоставляемый код

Фирма IBM предоставляет вам неисключительное право на использование всех приведенных примеров программного кода, на основе которых вы можете создавать собственные программы.

Все указанные примеры кода приведены фирмой IBM исключительно для иллюстрации. Они не были тщательно и всесторонне протестированы. По этой причине, фирма IBM не может гарантировать их надежность, удобство их обслуживания и отсутствие в них ошибок.

Все приведенные программы предоставляются на условиях “КАК ЕСТЬ” без каких-либо гарантий, включая гарантии соблюдения прав, коммерческой ценности и пригодности для конкретных целей.

Пример: Программа выхода из процедуры входа в систему сервера FTP на языке CL: Ниже приведен пример простой программы выхода, предназначенной для проверки запросов к серверу FTP. Она написана на управляющем языке (CL). Код этой программы неполон, но на его основе вы можете создать собственную программу.

Примечание: Ознакомьтесь с важной юридической информацией, приведенной в разделе Отказ от гарантий на предоставляемый код.

(Отформатированный текст данного примера может не поместиться во фрейм.)

```

/*****
/*
/* Sample FTP server logon exit program.
/* Note: This program is a sample only and has not undergone any formal
/* review or testing.
/*
/* Additional notes:
/* 1. When the FTP server logon exit is called, the FTP server job is
/* running under the QTCP user profile.
/* 2. For the ANONYMOUS case, users can add logging capability (for
/* example, write the E-mail address entered for the password and
/* the client IP address to a log file).
/* 3. IBM strongly recommends that you create the exit program in a library
/* with *PUBLIC authority set to *EXCLUDE, and give the exit program
/* itself a *PUBLIC authority of *EXCLUDE. The FTP server adopts
/* authority when it is necessary to resolve and call the exit program.
/*
/*
/*****

TSTLOGCL: PGM PARM(&APPIDIN &USRIN &USRLENIN &AUTIN &AUTLENIN +
&IPADDRIN &IPLLENIN &RETCDOUT &USRPRFOUT &PASSWDOUT +
&CURLIBOUT)

/* Declare input parameters */
DCL VAR(&APPIDIN) TYPE(*CHAR) LEN(4) /* Application identifier */
DCL VAR(&USRIN) TYPE(*CHAR) LEN(999)/* User ID */
DCL VAR(&USRLENIN) TYPE(*CHAR) LEN(4) /* Length of user ID */
DCL VAR(&AUTIN) TYPE(*CHAR) LEN(999)/* Authentication string */
DCL VAR(&AUTLENIN) TYPE(*CHAR) LEN(4) /* Length of auth. string */
DCL VAR(&IPADDRIN) TYPE(*CHAR) LEN(15) /* Client IP address */
DCL VAR(&IPLLENIN) TYPE(*CHAR) LEN(4) /* IP address length */
DCL VAR(&RETCDOUT) TYPE(*CHAR) LEN(4) /* return code (out) */
DCL VAR(&USRPRFOUT) TYPE(*CHAR) LEN(10) /* user profile (out) */
DCL VAR(&PASSWDOUT) TYPE(*CHAR) LEN(10) /* password (out) */
DCL VAR(&CURLIBOUT) TYPE(*CHAR) LEN(10) /* current library (out) */

/* Declare local copies of parameters (in format usable by CL) */

```

```

DCL      VAR(&APPID)      TYPE(*DEC)  LEN(1 0)
DCL      VAR(&USRLEN)     TYPE(*DEC)  LEN(5 0)
DCL      VAR(&AUTLEN)     TYPE(*DEC)  LEN(5 0)
DCL      VAR(&IPLLEN)     TYPE(*DEC)  LEN(5 0)

/* Assign input parameters to local copies */
CHGVAR  VAR(&APPID)      VALUE(%BINARY(&APPIDIN))
CHGVAR  VAR(&USRLEN)     VALUE(%BINARY(&USRLENIN))
CHGVAR  VAR(&AUTLEN)     VALUE(%BINARY(&AUTLENIN))
CHGVAR  VAR(&IPLLEN)     VALUE(%BINARY(&IPLLENIN))

/* Check for ANONYMOUS user. Allow for ANONYMOUSA, etc. as "regular" */
/* user profile. */
IF      COND(&USRLEN = 9) THEN(DO)
  IF      COND(%SST(&USRIN 1 9) = 'ANONYMOUS') THEN(DO)
/* For anonymous user: want to force user profile ANONYMOUS current library to PUBLIC. */
  CHGVAR VAR(%BINARY(&RETCDOU))  VALUE(6)
  CHGVAR VAR(&USRPRFOU)  VALUE('ANONYMOUS ')
  CHGVAR VAR(&CURLIBOU)  VALUE('PUBLIC ')
  ENDDO
/* Any other user: proceed with normal logon processing. */
  ELSE  CMD(CHGVAR VAR(%BINARY(&RETCDOU))  VALUE(1))
  ENDDO
  ELSE  CMD(CHGVAR VAR(%BINARY(&RETCDOU))  VALUE(1))

END:      ENDPGM

```

Пример: Программа выхода из процедуры входа в систему сервера FTP на языке C: Ниже приведен пример простой программы выхода, предназначенной для проверки запросов к серверу FTP. Она написана на языке C. Код этой программы неполон, но на его основе вы можете создать собственную программу.

Примечание: Ознакомьтесь с важной юридической информацией, приведенной в разделе Отказ от гарантий на предоставляемый код.

(Отформатированный текст данного примера может не поместиться во фрейм.)

```

/* Module Description *****/
/*
/*****
/*
/* Note: This program is a sample only and has NOT undergone any
/   formal review or testing.
/*
/*****
/*
/* Source File Name: qtmfsvrlgn.c
/*
/* Module Name: FTP Server Logon exit program.
/*
/* Service Program Name: n/a
/*
/* Source File Description:
/* This example exit program provides additional control over the
/* process of authenticating a user to a TCP/IP application server.*/
/* When installed, this example exit program would be called each
/* time a user attempts to log on to the server.
/*
/*****
/*
/* Function List: main      - FTP Server Logon exit program main.
/* qtmfsvrlgn - FTP Server Logon exit function.
/* CheckClientAddress - Check originating sessions IP
/* address.
/*
/* End Module Description *****/

```



```

#define _QTMFSVRLGN_C

/*****
/* All file scoped includes go here */
*****/
#ifndef __stdio_h
#include <stdio.h>
#endif

#ifndef __ctype_h
#include <ctype.h>
#endif

#ifndef __string_h
#include <string.h>
#endif

#ifndef __stdlib_h
#include <stdlib.h>
#endif

#include "qusec.h" /* Include for API error code structure */
#include "qsyrusri.h" /* Include for User Information API */

/*****
/* All file scoped Constants go here */
*****/
#define EQ ==
#define NEQ !=
#define BLANK ' '
#define FWIDTH 128 /* Width of one database file record */
#define FNAME 21 /* Qualified database file name width */

/* Valid characters for Client IP address. The CheckClientAddress()
/* function will check the Client IP address input argument
/* (ClientIPAddr_p) to ensure it is in valid dotted-decimal format.
/* This is one example of an input validity check.
const char ValidChars[] = "0123456789.";
/*****
/* All file scoped type declarations go here */
*****/

/*****
/* All file-scoped macro calls go here */
*****/

/*****
/* All internal function prototypes go here */
*****/
static void qtmfsvrlgn
(int,char *,int,char *,int,char *,int,int *,char *,char *,char *);

static int CheckClientAddress(char *, int);

/*****
/* All file scoped variable declarations go here */
*****/

/*****
/* ** NOTE ** */
/* The following client IP address are for example purposes only. Any
/* resemblance to actual system IP addresses is purely coincidental.
*****/
/* EXCLUSIVE system lists, ie - Logon attempts from any client IP */

```

```

/*          addresses NOT in one of these lists */
/*          are allowed to continue.          */
/* Reject server logon attempts of users attempting to log in from */
/* these client systems (return code = 0)      */
char Reject[] = "1.2.3.4 5.6.7.8";
/* Limit logon abilities of users attempting to log in as ANONYMOUS */
/* from these client systems (return code = 6).          */
/* In this example program, the initial current library is set and */
/* returned as an output parameter for users attempting to log in */
/* as ANONYMOUS from these specific client systems.          */
char Limit[] = "9.8.7.6 4.3.2.1 8.7.6.5";

/* Function Specification *****/
/*                               */
/* Function Name: Main          */
/*                               */
/* Descriptive Name: FTP Server Logon exit program main.          */
/*                               */
/* This example exit program allows access to a TCP/IP server to */
/* be controlled by the address of the originating session, gives */
/* additional control over the initial current library to a user, */
/* and provides the capability to implement "anonymous" FTP.     */
/*                               */
/* Notes:                      */
/*                               */
/* Dependencies:                */
/*   FTP Server Logon exit point QIBM_QTMF_SVR_LOGON was registered */
/*   during FTP product installation.                               */
/*                               */
/* Restrictions:                */
/*                               */
/*   None                       */
/*                               */
/* Messages:                    */
/*                               */
/*   None                       */
/*                               */
/* Side Effects:                */
/*                               */
/*   None                       */
/*                               */
/* Functions/Macros called:     */
/*                               */
/*   qtmfsvrlgn - Server Logon exit function.                      */
/*                               */

/* Input:                      */
/* int * argv[1] - Identifies requesting application              */
/*                (FTP Client =0, FTP Server = 1).              */
/* char * argv[2] - User identifier from client program.         */
/*                (For FTP server, this is user CMD data        */
/* int * argv[3] - Length (in bytes) of User ID string.         */
/* char * argv[4] - Authentication string from client.          */
/*                (For FTP server, this is the password)*/
/* int * argv[5] - Length (bytes) Authentication string.        */
/* char * argv[6] - Internet Protocol address from which        */
/*                the session originates.                       */
/* int * argv[7] - Length (in bytes) of IP address.             */
/* int * argv[8] - Return code (received as 0).                 */
/* char * argv[9] - User profile (received as blanks).          */
/* char * argv[10] - Password (received as blanks).             */
/* char * argv[11] - Initial current library (received as blanks)*/
/*                               */
/* Exit Normal: Return Return Code, User Profile, Password, Initial */
/*                Current Library to server application.        */
/*                               */

```

```

/* Exit Error:  None */
/* */
/* End Function Specification *****/
void main(int argc, char *argv[])
{
/*****/
/* Code */
/*****/

/*****/
/* Collect input arguments and call function to determine if client */
/* should be allowed to log in to an FTP server application. */
/*****/
qtmfsvr1gn(*(int *)argv[1]), /* Application Identifier
(Input) */
        argv[2], /* User Identifier (Input) */
        (*(int *)argv[3]), /* Length User of
Identifier(Input) */
        argv[4], /* Authentication String (Input) */
        (*(int *)argv[5]), /* Length of Authentication string */
(Input) */
        argv[6], /* Client IP Address (Input) */
        (*(int *)argv[7]), /* Length of Client IP Address */
(Input) */
        (int *)argv[8], /* Return Code (Output)*/
        argv[9], /* User Profile (Output)*/
        argv[10], /* Password (Output)*/
        argv[11]); /* Initial Current Library (Output)*/
return;
}

```

```

/* Function Specification *****/
/* */
/* Function Name: qtmfsvr1gn */
/* */
/* Descriptive Name: Server Logon exit function. */
/* */
/* This exit function provides control over user authentication to */
/* an FTP server. */
/* */
/* Notes: */
/* */
/* Dependencies: */
/* */
/* FTP Server Logon exit point QIBM_QTMF_SVR_LOGON was */
/* registered during FTP product installation. */
/* */
/* Restrictions: */
/* */
/* None */
/* */
/* Messages: */
/* */
/* None */
/* */
/* Side Effects: */
/* */
/* None */
/* */
/* Functions/Macros called: */
/* */
/* CheckClientAddress - Check the ClientIPAddr_p input argument.*/
/* memcpy - Copy bytes from source to destination. */
/* memset - Set bytes to value. */
/* strstr - Locate first occurrence of substring. */

```

```

/*      sprintf - Formatted print to buffer.          */
/*                                                    */
/* Input:                                           */
/* int    ApplId      - Application Identifier (Server = 1). */
/* char *  UserId_p    - User identifier from client program. */
/*                               (For FTP server, USER subcommand data)*/
/* int    Lgth_UserId  - Length (in bytes) of user ID string. */
/* char *  AuthStr_p   - Authentication string from client.   */
/*                               (For FTP server, this is the password)*/
/* int    Lgth_AuthStr - Length (bytes) Authentication string. */
/* char *  ClientIPAddr_p - Internet Protocol address from which */
/*                               the session originates.         */
/* int *   Lgth_ClientIPAddr - Length (in bytes) of IP address. */
/*                                                    */

/* Output:                                           */
/* int * ReturnCode: Indicates degree of success of operation: */
/* ReturnCode = 0 - Reject logon.                          */
/* ReturnCode = 1 - Continue logon; use initial current library*/
/* ReturnCode = 2 - Continue logon; override initial current */
/*                               library                      */
/* ReturnCode = 3 - Continue logon; override user, password */
/* ReturnCode = 4 - Continue logon; override user, password, */
/*                               current library              */
/* ReturnCode = 5 - Accept logon; override user profile     */
/* ReturnCode = 6 - Accept logon; override user profile,   */
/*                               current library             */
/* char * UserProfile - User profile to use for this session */
/* char * Password    - Password to use for this session     */
/* char * Init_Cur_Lib - Initial current library for this session */
/*                                                    */
/* Exit Normal: (See OUTPUT)                          */
/*                                                    */
/* Exit Error: None                                   */
/*                                                    */
/* End Function Specification *****/
static void qtmfsvrlgn(int ApplId, /* Entry point */
                      char *UserId_p,
                      int Lgth_UserId,
                      char *AuthStr_p,
                      int Lgth_AuthStr,
                      char *ClientIPAddr_p,
                      int Lgth_ClientIPAddr,
                      int *ReturnCode,
                      char *UserProfile_p,
                      char *Password_p,
                      char *InitCurrLib_p)
{
  /* *****/
  /* Local Variables */
  /* *****/
  /* The following lists serve as an example of an additional layer */
  /* of control over user authentication to an application server. */
  /* Here, logon operations using the following user identifiers */
  /* will be allowed to continue, but the output parameters returned */
  /* by this example exit program will vary depending on which list */
  /* a user identifier (UserId_p) is found in. */
  /* For example, attempts to logon as FTPUSR11 or FTPUSR2 will be */
  /* allowed, and this example exit will return the initial current */
  /* library as an output parameter along with a return code of 2. */
  /* *****/
  /* Continue the logon operation, Return Code = 1 */
  char Return1[] = "FTPUSR10 ";
  /* Continue the logon operation, Return Code = 2 */
  char Return2[] = "FTPUSR11 FTPUSR2 ";
  /* Continue the logon operation, Return Code = 3

```

```

char Return3[] = "FTPUSR12 FTPUSR3 FTPUSR23 ";
/* Continue the logon operation, Return Code = 4 */
char Return4[] = "FTPUSER FTPUSR4 FTPUSR24 FTPUSR94 ";
int rc; /* Results of server logon request */
Qsy_USRI0300_T Receiver_var; /* QSYRUSRI API Receiver variable */
int Lgth_Receiver_var; /* Receiver variable length */
char Format_Name[8]; /* Format name buffer */
char User_Id[10]; /* User Identifier buffer */
Qus_EC_t error_code = /* QSYRUSRI API error code structure: */
{
    sizeof(Qus_EC_t), /* Set bytes provided */
    0, /* Initialize bytes available */
    ' ',' ',' ',' ',' ',' ',' ',' ',' ', /* Initialize Exception Id */
};
char *pcTest_p; /* Upper-case User Identifier pointer*/
int i; /* "For" loop counter variable */

/*****
/* Code */
*****/

/* Test validity of application ID input argument. */
if(1 NEQ ApplId)
{
    /* ERROR - Not FTP server application. */
    /* Return Code of 0 is used here to indicate */
    /* that an incorrect input argument was received. */
    /* The server logon operation will be rejected. */
    rc = 0; /* Application ID not valid */
} /* End If the application identifier is NOT for FTP server */
else /* FTP server application identifier */
{
    /* Validate the client IP address input argument. */
    rc = CheckClientAddress(ClientIPAddr_p,
        Lgth_ClientIPAddr);
    if(0 NEQ rc) /* Valid, acceptable client address */
    {
        /* Initialize User_Id; used to hold upper-cased user identifier */
        memset(User_Id, BLANK, sizeof(User_Id));

        /* Initialize pcTest_p to point to UserId_p input argument. */
        pcTest_p = UserId_p;

        /* Uppercase all of the user ID to compare for ANONYMOUS user. */
        for(i = 0; i < Lgth_UserId; i++)
        {
            User_Id[i] = (char)toupper(*pcTest_p);
            pcTest_p += 1;
        }

        /* If user has logged in as ANONYMOUS. */
        if(0 == memcmp("ANONYMOUS ", User_Id, 10))
        {
            /* Determine how to continue with ANONYMOUS logon attempt. */
            if(NULL NEQ strstr(Limit, ClientIPAddr_p))
            {
                /* If users system IP address is found in the "Limit" list, */
                /* return ReturnCode of 6, user profile and initial */
                /* current library values as output parameters. */
                memcpy(UserProfile_p, "USERA1 ", 10);
                memcpy(InitCurrLib_p, "PUBLIC ", 10);
                rc = 6;
            }
        }
        else
        {

```

```

    /* Users system IP address is NOT found in the "Limit" list,*/
    /* return ReturnCode of 5, user profile output parameter; */
    /* use the initial current library that is specified by the */
    /* user profile information. */
    memcpy(UserProfile_p, "USERA1", 10);
    rc = 5;
}
/* End If USER is ANONYMOUS */

else /* Else USER is not ANONYMOUS */
{
    /* Set receiver variable length. */
    Lgth_Receiver_var = sizeof(Qsy_USRI0300_T);
    /* Set return information format. */
    memcpy(Format_Name, "USRI0300", sizeof(Format_Name));
    /* Set user identifier passed in. */
    memset(User_Id, BLANK, sizeof(User_Id));
    memcpy(User_Id, UserId_p, Lgth_UserId);
    /* Call QSYRUSRI - Retrieve User Information API */
    QSYRUSRI(&Receiver_var, /* Return Information receiver var */
            Lgth_Receiver_var, /* Receiver variable length */
            Format_Name, /* Return information format name */
            User_Id, /* User ID seeking information */
            &error_code); /* Error return information */
    /* Check if an error occurred (byte_available not equal 0) */
    if(0 NEQ error_code.Bytes_Available)
    {
        /* Return ReturnCode of 0 only (Reject logon); */
        rc = 0; /* Reject the logon operation */
        *ReturnCode = rc; /* Assign result to ReturnCode */
    }
    else /* No error occurred from Retrieve User Info */
    {
        /* (Bytes_Available = 0) */
        /* Set current library for user profile. */
        memcpy(InitCurrLib_p, Receiver_var.Current_Library, 10);
        if(NULL NEQ strstr("CRTDFT",
            Receiver_var.Current_Library))
        {
            memcpy(InitCurrLib_p, "FTPDEFAULT", 10);
        }
    }
    else
    {
        if(NULL NEQ strstr(Return1, UserId_p))
        {
            /* Return ReturnCode of 1 (Continue logon); */
            /* Also return user profile and password output */
            /* parameters to endure they are ignored by the server.*/
            memcpy(UserProfile_p, UserId_p, Lgth_UserId);
            memcpy>Password_p, AuthStr_p, Lgth_AuthStr);
            rc = 1; /* Continue the logon operation */
        }
    }
    else
    {
        if(NULL NEQ strstr(Return2, UserId_p))
        {
            /* Return ReturnCode of 2, and initial current library*/
            /* Also return user profile and password values */
            /* even though they will be ignored by the server. */
            memcpy(UserProfile_p, UserId_p, Lgth_UserId);
            memcpy>Password_p, AuthStr_p, Lgth_AuthStr);
            memcpy(InitCurrLib_p, "FTPEXT2",
                strlen("FTPEXT2"));
            rc = 2; /* Continue logon; return InitCurLib */
        }
    }
}

```



```

/*                                                     */
/* Messages:                                           */
/*   None                                             */
/*                                                     */
/* Side Effects:                                       */
/*   None                                             */
/*                                                     */
/* Functions/Macros called:                            */
/*                                                     */
/*   strstr - Search for first occurrence of a string. */
/*                                                     */
/* Input:                                              */
/*   char * ClientIPAddr_p - Internet Protocol address from which */
/*                         the session originates.             */
/*   int * Lgth_ClientIPAddr - Length (in bytes) of IP address. */
/*                                                     */
/* Output:                                             */
/*   int rc - Return code indicating validity of IP */
/*           address from ClientIPAddr_p input.           */
/*           0 = Reject the logon operation.              */
/*           ClientIPAddr_p is one that is not          */
/*           allowed, or contains a character           */
/*           that is not valid.                          */
/*           1 = Continue the logon operation.           */
/*                                                     */
/* Exit Normal: (See OUTPUT)                           */
/*                                                     */
/* Exit Error: None.                                   */
/*                                                     */
/* End Function Specification *****/

```

```

static int CheckClientAddress(char *ClientIPAddr_p, /* Entry point */
                             int Lgth_ClientIPAddr)
{
  /******
  /* Local Variables
  /******
  int rc; /* Return code */

  /******
  /* Code
  /******
  /* Check that client IP address input argument is dotted-decimal
  /* format of minimum length, with no leading blanks or periods,
  /* and contains only valid characters.
  if((Lgth_ClientIPAddr < 7) || /* Minimum IP address size */
      (strstr(ClientIPAddr_p, ValidChars) < Lgth_ClientIPAddr)||
      (strstr(ClientIPAddr_p, ".") EQ 1)|| /* Leading '.' in IP */
      (strstr(ClientIPAddr_p, " ") EQ 1)) /* Leading blank in IP */
  {
    /* Client's IP address not valid, or contains an incorrect character */
    rc = 0; /* Client IP address input argument not valid */
  }
  else
  {
    /* Is client system allowed to log in to FTP server? */
    if(NULL NEQ strstr(Reject, ClientIPAddr_p))
    {
      /* Return code = 0 - Reject the server logon operation, as the */
      /* client IP address is found in the global */
      /* "Reject" list. */
      rc = 0; /* Reject the logon operation */
    }
    else
    {
      /* Continue the server logon operation checks. */

```



```

        rc = 1;                /* Continue the logon operation */
    }
}
return(rc);
}

#undef _QTMFSVRLGN_C

```

Пример: Программа выхода из процедуры входа в систему сервера FTP на языке RPG: Ниже приведен пример простой программы выхода, предназначенной для проверки запросов к серверу FTP. Она написана на языке ILE RPG. Код этой программы неполон, но на его основе вы можете создать собственную программу.

Примечание: Ознакомьтесь с важной юридической информацией, приведенной в разделе Отказ от гарантий на предоставляемый код.

(Отформатированный текст данного примера может не поместиться во фрейм.)

```

* Module Description *****
*
*****
*
* Note: This program is a sample only and has NOT undergone any
*       formal review or testing.
*
*****
*
*           PROGRAM FUNCTION
*
* This program demonstrates some of the abilities an FTP Server
* Logon Exit Program can have.
*
*****
F/SPACE 3
*****
*
*           INDICATOR USAGE
*
*  IND.  DESCRIPTION
*
*  LR - CLOSE FILES ON EXIT
*
*****
F/EJECT
*****
* DATA STRUCTURES USED BY THIS PROGRAM
*****
*
* Define constants
*
1 D Anonym          C          CONST('ANONYMOUS ')
D Text1            C          CONST('Anonymous (')
D Text2            C          CONST(') FTP logon')
D InvalidNet       C          CONST('10.')
C/EJECT
*****
* VARIABLE DEFINITIONS AND LISTS USED BY THIS PROGRAM
*****
C/SPACE 2
*
* Define binary parameters
*
D
D APPIDds          DS
D APPIDds          1          4B 0
D USRLENDs         5          8B 0
D AUTLENDs         9          12B 0

```

```

D IPLENDs          13      16B 0
D RETCDds          17      20B 0
*
C *LIKE            DEFINE  APPIDds    APPIDIN
C *LIKE            DEFINE  USRLENds   USRLENIN
C *LIKE            DEFINE  AUTLENds   AUTLENIN
C *LIKE            DEFINE  IPLENDs    IPLENIN
C *LIKE            DEFINE  RETCDds    RETCDOU
*
* Define parameter list
*
C *Entry          PLIST
* Input parameters:
C                PARM                APPIDIN                Application ID
*                possible values:    1 = FTP Server Program
C                PARM                USRIN                999                User ID
C                PARM                USRLENIN                Length of User ID
C                PARM                AUTIN                999                Authentication Strg
C                PARM                AUTLENIN                Length of Auth. Strg
C                PARM                IPADDRIN                15                Client IP Address
C                PARM                IPLENIN                Length of IP Address
* Return parameters:
C                PARM                RETCDOU                Return Code (Out)
*                possible values:    0 = Reject Logon
*                1 = Continue Logon
*                2 = Continue Logon,
*                override current
*                library
*                3 = Continue Logon,
*                override user prf,
*                password
*                4 = Continue Logon,
*                override user prf,
*                password, current
*                library
*                5 = Accept logon with
*                user prf returned
*                6 = Accept logon with
*                user prf returned,
*                override current
*                library
C                PARM                USRPRFOUT                10                User Profile (Out)
C                PARM                PASSWDOUT                10                Password (Out)
C                PARM                CURLIBOUT                10                Current Lib. (Out)
C/EJECT
*****
* THE MAIN PROGRAM *
*****
*
* Check for ANONYMOUS user
* 1
C USRLENIN        SUBST(P)  USRIN:1        User                10
C User            IFEQ      Anonym
C                MOVEL      Anonym        USRPRFOUT
*
* Check if the user entered something as a e-mail address
*
C AUTLENIN        IFGT      *ZERO                E-mail addr. entered
*
* Check if the E-mail address is a valid one
*
C                Z-ADD      0                i                3 0
C '@'            SCAN      AUTIN:1            i
*
*
C i              IFGT      0
C AUTLENIN        SUBST(P)  AUTIN:1        Email                30

```

```

C          Z-ADD      5          RETCDOU      Accept Logon
*
* Log Anonymous FTP Logon to message queue QSYSOPR
* (The logging should be done to a secure physical file!!!!!!)
*
C  Text1      CAT(p)   Email:0     Message      43
C  Message    CAT(p)   Text2:0     Message
C  Message    DSPLY    'QSYSOPR'
*
C          ELSE
C          Z-ADD      0          RETCDOU      Invalid E-mail addr
C          ENDIF      Reject Logon attempt
*
C          ELSE
C          Z-ADD      0          RETCDOU      No E-mail address
C          ENDIF      Reject Logon attempt
*
C          ELSE
*
* Any Other User: Proceed with Normal Logon Processing, but the Client address must not belong
*                  to network 10.xxx.xxx.xxx
*
C  3          SUBST    IPADDRIN:1  TheNet        3
C  TheNet     IFEQ     InvalidNet
C          Z-ADD      0          RETCDOU      Wrong Net
C          ELSE
C          Z-ADD      1          RETCDOU      Reject Logon attempt
C          ENDIF      Right Net
C                                     Continue with Logon
*
C          ENDIF
*
C          EVAL      *INLR = *ON
C          RETURN

```

Формат точки выхода TCPL0100:

Имя формата точки выхода: TCPL0100

Имя точки выхода: QIBM_QTMF_SVR_LOGON

Имя точки выхода: QIBM_QTMX_SVR_LOGON

Группа обязательных параметров:

1	Идентификатор приложения	Входной	Binary(4)
2	Идентификатор пользователя	Входной	Char(*)
3	Длина идентификатора пользователя	Входной	Binary(4)
4	Строка с идентификационной информацией	Входной	Char(*)
5	Длина строки с идентификационной информацией	Входной	Binary(4)
6	IP-адрес клиента	Входной	Char(*)
7	Длина IP-адреса клиента	Входной	Binary(4)
8	Код возврата	Выходной	Binary(4)
9	Пользовательский профайл	Выходной	Char(10)
10	Пароль	Выходной	Char(10)

11	Начальная библиотека	Выходной	Char(10)
----	----------------------	----------	----------

Описания параметров

Идентификатор приложения Входной; BINARY(4) Идентификатор сервера приложений, к которому обращен запрос. Допустимые значения:

- 1**
Сервер FTP
- 2**
Сервер REXEC

Идентификатор пользователя

Входной; CHAR(*) Идентификатор пользователя, переданный программой-клиентом. При работе с сервером FTP этот параметр содержит информацию, переданную в команде USER.

Длина идентификатора пользователя

Входной; BINARY(4) Задаёт длину идентификатора пользователя (в байтах).

Строка с идентификационной информацией

Входной; CHAR(*) Строка, переданная программой-клиентом (например, пароль).

При работе с сервером FTP этот параметр содержит информацию, переданную в команде PASS (пароль). В выпусках V5R1 и старше значение этого параметра не указывается, если пользователь был идентифицирован с помощью сертификата клиента.

Длина строки с идентификационной информацией

Входной; BINARY(4) Длина строки с идентификационной информацией (в байтах).

Примечание: Если при работе с сервером FTP клиент был идентифицирован с помощью сертификата клиента, то значение этого параметра равно 0.

IP-адрес клиента

Входной; CHAR(*) IP-адрес, от которого получен запрос на открытие сеанса. Значение должно быть задано в десятичном формате с точками и выровнено по левому краю.

Длина IP-адреса клиента

Входной; BINARY(4) Задаёт длину IP-адреса клиента (в байтах).

Код возврата

Выходной; BINARY(4) Указывает, будет ли разрешен вход в систему, будет ли выполнена идентификация пароля и будет ли изменена начальная библиотека. Допустимые значения:

- 0**
Запретить вход в систему. Значения выходных параметров, задающих имя пользовательского профайла, пароль и начальную библиотеку, игнорируются.
- 1**
Разрешить вход в систему с использованием указанного идентификатора пользователя, строки с идентификационной информацией и значения начальной библиотеки. В качестве имени пользовательского профайла будет применяться идентификатор пользователя, а в качестве пароля - строка с идентификационной информацией. Программа игнорирует значения выходных параметров, задающих имя пользовательского профайла, пароль и

начальную библиотеку.

Примечание: Для успешного входа в систему строка с идентификационной информацией должна совпадать с паролем пользовательского профайла.

2

Разрешить вход в систему с использованием указанного идентификатора и строки с идентификационной информацией. В качестве текущей библиотеки применять библиотеку, указанную в параметре начальной библиотеки. В качестве имени пользовательского профайла применяется идентификатор пользователя. В качестве пароля принимается строка идентификации. Требуется задать выходной параметр начальной текущей библиотеки. Программа игнорирует значения выходных параметров, задающих имя пользовательского профайла и пароль.

Примечание: Для успешного входа в систему строка с идентификационной информацией должна совпадать с паролем пользовательского профайла.

3

Разрешить вход в систему. Заменить пользовательский профайл и пароль на значения выходных параметров данной программы выхода. Применять начальную библиотеку, указанную в пользовательском профайле, имя которой возвращает программа выхода. Программа игнорирует выходной параметр начальной библиотеки.

Примечание: Для успешного входа в систему указанный в выходном параметре пароль должен совпадать с паролем пользовательского профайла. **Внимание!** Фирма IBM настоятельно рекомендует **никогда** не задавать пароли непосредственно в программах выхода. Существуют алгоритмы для определения зашифрованных паролей.

4

Разрешить вход в систему. В качестве имени и пароля пользовательского профайла, а также имени начальной библиотеки будут применяться значения, указанные в соответствующих выходных параметрах данной программы выхода.

Примечание: Для успешного входа в систему указанный в выходном параметре пароль должен совпадать с паролем пользовательского профайла. **Внимание!** Фирма IBM настоятельно рекомендует **никогда** не задавать пароли непосредственно в программах выхода. Существуют алгоритмы для определения зашифрованных паролей.

5

Разрешить вход в систему. В качестве имени пользовательского профайла применять значение, указанное в соответствующем выходном параметре. В качестве начальной библиотеки применять библиотеку, указанную в пользовательском профайле, имя которого возвращается данной программой выхода. Значения выходных параметров, задающих начальную библиотеку и пароль, игнорируются.

Примечание: Это значение переопределяет обычную процедуру проверки пароля OS/400. Проверка пароля выполняется только один раз.

6

Разрешить вход в систему. В качестве имени пользовательского профайла, а также имени начальной библиотеки применять значения, указанные в соответствующих выходных параметрах. Значение выходного параметра, задающего пароль, игнорируется.

Примечание: Это значение переопределяет обычную процедуру проверки пароля OS/400. Проверка пароля выполняется только один раз.

Пользовательский профайл

Выходной; CHAR(10) Имя пользовательского профайла, который должен применяться в данном сеансе. Значение параметра должно быть выровнено по левому краю и дополнено справа пробелами.

Пароль

Выходной; CHAR(10) Пароль, который должен применяться в данном сеансе. Значение параметра должно быть выровнено по левому краю и дополнено справа пробелами.

Начальная библиотека

Выходной; CHAR(10) Текущая библиотека, которая должна применяться по умолчанию после начала сеанса. Значение параметра должно быть выровнено по левому краю и дополнено справа пробелами.

Перейти:

- Замечания о работе с форматом TCPL0100

Дополнительная информация приведена в следующих разделах:

- Формат TCPL0200: Данная точка выхода предоставляет дополнительные параметры для управления входом в систему.
- Формат TCPL0300: Данная точка выхода задает сервер приложений, от которого был получен запрос.
- Точка выхода и процедуры входа в систему сервера: Точка выхода из процедуры входа в систему сервера приложения TCP/IP предназначена для идентификации пользователей.

Замечания о работе с форматом TCPL0100: Если при работе с FTP программа выхода возвращает недопустимое значение выходного параметра, то сервер FTP запрещает выполнять операцию. В этом случае в протокол задания заносится сообщение Данные программы выхода для точки выхода &1 отсутствуют или недопустимы.

Если при вызове программы выхода для FTP возникает исключительная ситуация, то сервер FTP отправляет следующее сообщение: В программе выхода FTP &1 из библиотеки &2 для точки выхода &3 возникла исключительная ситуация

В приведенной ниже описаны действия, выполняемые сервером FTP при получении различных кодов возврата от программы выхода (код возврата задается в параметре 8).

Примечание: В поле 'Возвращаемое значение' указывается значение, которые программа выхода должна вернуть в соответствующем параметре. Это значение применяется сервером FTP для выполнения оставшейся части процедуры входа в систему.

Возвращаемое значение	Пользовательский профайл (9)	Пароль (10)	Начальная библиотека (11)
0	Игнорируется	Игнорируется	Игнорируется
1	(Идентификатор пользователя, параметр 2)	(Пароль, параметр 4)	(Значение из пользовательского профайла)

Возвращаемое значение	Пользовательский профайл (9)	Пароль (10)	Начальная библиотека (11)
2	(Идентификатор пользователя, параметр 2)	(Пароль, параметр 4)	Возвращаемое значение
3	Возвращаемое значение	Возвращаемое значение	(Значение из пользовательского профайла)
4	Возвращаемое значение	Возвращаемое значение	Возвращаемое значение
5	Возвращаемое значение	Игнорируется	(Значение из пользовательского профайла)
6	Возвращаемое значение	Игнорируется	Возвращаемое значение

В приведенной выше таблице в скобках указано значение, которое будет применяться приложением TCP/IP в том случае, если выходной параметр игнорируется. Значение Игнорируется указывает, что с данным кодом возврата этот параметр не используется, и его можно не задавать.

При работе с сервером FTP (точка выхода QIBM_QTMF_SVR_LOGON, идентификатор приложения 1): если для точки выхода задана программа выхода и указан идентификатор пользователя ANONYMOUS, то сервер отправит клиенту специальное сообщение с предложением ввести пароль: 331 Выполняется вход в систему, укажите в качестве пароля полный адрес электронной почты. Это сообщение выдается приложением до вызова программы выхода.

Если приложение разрешает вход в систему, сервер FTP отправляет ответ 230 Вход в систему с гостевым профайлом разрешен, права доступа ограничены

Для сервера REXEC (идентификатор приложения 2):

1. Если возвращается недопустимое значение выходного параметра Разрешить выполнение операции, то сервер REXEC запрещает выполнение операции. В протокол задания сервера REXEC будет занесено сообщение "Данные программы выхода для точки выхода &1 отсутствуют или недопустимы"
2. Если при вызове программы выхода возникает исключительная ситуация, сервер REXEC запрещает выполнение операции. В протокол задания будет занесено следующее сообщение: "В программе выхода REXEC &1 из библиотеки &2 для точки выхода &3 возникла исключительная ситуация."

Формат точки выхода TCPL0200:

Имя формата точки выхода: TCPL0200

Имя точки выхода: QIBM_QTMF_SVR_LOGON

Группа обязательных параметров:

1	Идентификатор приложения	Входной	Binary(4)
2	Идентификатор пользователя	Входной	Char(*)
3	Длина идентификатора пользователя	Входной	Binary(4)
4	Строка с идентификационной информацией	Входной	Char(*)

1	Идентификатор приложения	Входной	Binary(4)
5	Длина строки с идентификационной информацией	Входной	Binary(4)
6	IP-адрес клиента	Входной	Char(*)
7	Длина IP-адреса клиента	Входной	Binary(4)
8	Разрешить вход в систему	Выходной	Binary(4)
9	Пользовательский профайл	Выходной	Char(10)
10	Пароль	Выходной	Char(10)
11	Начальная библиотека	Входной/выходной	Char(10)
12	Начальный домашний каталог	Выходной	Char(*)
13	Длина имени начального домашнего каталога	Входной/выходной	Binary(4)
14	Информация о приложении	Входной/выходной	Char(*)
15	Длина информации о приложении	Входной	Binary(4)

Описания параметров

Идентификатор приложения

Входной; BINARY(4) Задаёт сервер приложения, от которого получен запрос. Допустимые значения:

1

Сервер FTP

Идентификатор пользователя

Входной; CHAR(*) Идентификатор пользователя, переданный программой-клиентом. При работе с сервером FTP этот параметр содержит информацию, переданную в команде USER.

Длина идентификатора пользователя

Входной; BINARY(4) Задаёт длину идентификатора пользователя (в байтах).

Строка с идентификационной информацией

Входной; CHAR(*) Строка, переданная программой-клиентом (например, пароль).

При работе с сервером FTP этот параметр содержит информацию, переданную в команде PASS (пароль). В выпусках V5R1 и старше значение этого параметра не указывается, если пользователь был идентифицирован с помощью сертификата клиента.

Длина строки с идентификационной информацией

Входной; BINARY(4) Длина строки с идентификационной информацией (в байтах).

Примечание:

Если при работе с сервером FTP клиент был идентифицирован с помощью сертификата клиента, то значение этого параметра равно 0.

IP-адрес клиента

Входной; CHAR(*) IP-адрес, от которого получен запрос на открытие сеанса. Значение должно быть задано в десятичном формате с точками и выровнено по левому краю.

Длина IP-адреса клиента

Входной; BINARY(4) Задаёт длину IP-адреса клиента (в байтах).

Разрешить вход в систему

Выходной; BINARY(4) Указывает, следует ли разрешить вход в систему, и задаёт способ идентификации с помощью пароля. Допустимые значения:

0

Запретить вход в систему. Остальные выходные параметры игнорируются.

1

Разрешить вход в систему с указанным идентификатором пользователя и строкой идентификационной информации. В качестве имени пользовательского профайла будет применяться идентификатор пользователя, а в качестве пароля - строка с идентификационной информацией. В качестве текущей библиотеки и рабочего каталога применяются значения выходных параметров. Приложение игнорирует значения выходных параметров, задающих имя пользовательского профайла и пароль.

Примечание: Для успешного входа в систему строка с идентификационной информацией должна совпадать с паролем пользовательского профайла.

2

Разрешить вход в систему. В качестве имени и пароля пользовательского профайла будут применены возвращаемые значения, указанные в соответствующих выходных параметрах программы выхода. В качестве текущей библиотеки и рабочего каталога применяются соответствующие значения выходных параметров.

Примечание: Для успешного входа в систему указанный в выходном параметре пароль должен совпадать с паролем пользовательского профайла. **Внимание!** Фирма IBM настоятельно рекомендует **никогда** не задавать пароли непосредственно в программах выхода. Существуют алгоритмы для определения зашифрованных паролей.

3

Разрешить вход в систему. В качестве имени пользовательского профайла применять возвращаемое значение, указанное в соответствующем выходном параметре данной программы выхода. В качестве текущей библиотеки и рабочего каталога применяются соответствующие значения выходных параметров. Значение выходного параметра, задающего пароль, игнорируется.

Примечание: В системе с уровнем защиты не ниже 20 это значение переопределяет обычную процедуру проверки пароля OS/400. Проверка пароля выполняется только один раз.

Пользовательский профайл

Выходной; CHAR(10) Имя пользовательского профайла, который должен применяться в данном сеансе. Значение параметра должно быть выровнено по левому краю и дополнено справа пробелами.

Пароль

Выходной; CHAR(10) Пароль, который должен применяться в данном сеансе. Значение параметра должно быть выровнено по левому краю и дополнено справа пробелами.

Начальная библиотека

Выходной; CHAR(10) Начальная библиотека, которая должна применяться в данном сеансе. Значение параметра должно быть выровнено по левому краю и дополнено справа пробелами. При вызове программы выхода этому параметру присваивается следующее специальное значение:

***CURLIB**

Оно означает, что в качестве текущей будет применяться библиотека, указанная в пользовательском профайле.

Начальный домашний каталог

Выходной; CHAR(*) Начальный домашний каталог, который должен применяться в данном сеансе. Имя каталога должно быть задано полностью.

Длина начального домашнего каталога

Входной/выходной параметр; BINARY(4) Длина имени начального домашнего каталога, возвращаемого программой выхода. При вызове программы выхода этому параметру присваивается нулевое значение. Если программа выхода не изменяет значение этого параметра, то в качестве домашнего каталога применяется каталог, указанный в пользовательском профайле.

Информация о приложении

Входной/выходной; CHAR(*) Информация для задания параметров входа в систему, предусмотренных для конкретного приложения. Информация о формате этого значения приведена в разделе Формат информации о приложении.

Длина информации о приложении Входной; BINARY(4) Длина (в байтах) информации о приложении.

Дополнительная информация приведена в следующих разделах:

- Точка выхода и процедуры входа в систему сервера: Точка выхода из процедуры входа в систему сервера приложения TCP/IP предназначена для идентификации пользователей.

Формат информации о приложении: Если в качестве идентификатора приложения задан идентификатор сервера FTP, то информация о приложении должна быть задана в следующем формате:

Смещение десятич.	Смещение шестнад.	Тип	Поле
0	0	BINARY(4)	Начальный формат имени
4	4	BINARY(4)	Начальный рабочий каталог
8	8	BINARY(4)	Начальный формат списка файлов
12	C	BINARY(4)	Способ защиты соединения для передачи управляющей информации
16	10	BINARY(4)	Опция шифрования передаваемых данных
20	14	BINARY(2)	Программа для шифрования управляющей информации
22	16	BINARY(2)	Программа для шифрования данных

Описания полей

Начальный формат имени

Задаёт начальный формат имен файлов для данного сеанса. При вызове программы выхода в этом поле устанавливается значение, указанное для параметра NAMEFMT в файле конфигурации сервера FTP. Допустимы следующие значения:

0

Применять формат имен Библиотека/Файл.Элемент. Это значение соответствует опции NAMEFMT(*LIB) команды CHGFTPА и эквивалентно выполнению команды SITE NAMEFMT 0 сервера FTP.

1

Использовать формат имен с указанием пути. Это значение соответствует опции NAMEFMT(*PATH) команды CHGFTPА и эквивалентно выполнению команды SITE NAMEFMT 1 для сервера FTP.

Начальный рабочий каталог

Задаёт начальный рабочий каталог сервера FTP, который будет использоваться по умолчанию для операций с файлами и просмотра содержимого каталога. При вызове программы выхода в этом поле устанавливается значение, указанное для параметра CURDIR в файле конфигурации сервера FTP. Допустимы следующие значения:

0

Применять текущую библиотеку в качестве начального каталога сервера FTP. Это значение соответствует опции CURDIR(*CURLIB) команды CHGFTPА.

1

Применять домашний каталог в качестве начального каталога сервера FTP. Это значение соответствует опции CURDIR(*HOMEDIR) команды CHGFTPА.

Примечание:

Если в этом поле указано значение 1, то в поле начального формата имен также необходимо задать значение 1.

Начальный формат списка файлов

Задаёт начальный формат списка файлов для данного сеанса. При вызове программы выхода в этом поле устанавливается значение, указанное для параметра LISTFMT в файле конфигурации сервера FTP. Допустимы следующие значения:

0

Применять формат списка файлов сервера iSeries. Это значение соответствует опции LISTFMT(*DFT) команды CHGFTPА и эквивалентно выполнению команды SITE LISTFMT 0 сервера FTP.

1

Применять формат списков файлов UNIX. Это значение соответствует опции LISTFMT(*UNIX) команды CHGFTPА и эквивалентно выполнению команды SITE LISTFMT 1 сервера FTP.

Способ защиты соединения для передачи управляющей информации

Задаёт способ защиты соединения для передачи управляющей информации, применяемый в данном сеансе FTP. Допустимы следующие значения:

0

Управляющая информация передается по незащищенному соединению. **1**

Для защиты управляющей информации применяется протокол SSL. Это значение соответствует вызову команды клиента FTP AUTH с параметром TLS-P или SSL.

2

Для защиты управляющей информации применяется протокол SSL. Это значение аналогично вызову команды клиента AUTH с параметром TLS-C или TLS.

Примечания:

- Программа выхода может только считывать данные из этого поля. Все изменения, вносимые программой выхода, игнорируются.
- Если сеанс устанавливается через защищенный порт FTP, то этому параметру присваивается значение 1. Можно считать, что при установлении соединений через защищенный порт FTP неявно вызывается команда AUTH SSL.

Опция шифрования передаваемых данных

Указывает, следует ли шифровать данные, передаваемые в данном сеансе FTP. Допустимы следующие значения:

-1

В данном сеансе FTP шифрование данных не применяется.

0

В данном сеансе FTP разрешено шифрование данных, однако его применение необязательно.

1

В данном сеансе FTP могут передаваться только зашифрованные данные.

Примечания:

- Если в качестве способа защиты соединения для передачи управляющей информации выбрано значение 1, а опция шифрования передаваемых данных равна -1, то для передачи данных клиенту придется применять дополнительные команды FTP. (Если установлен способ защиты TLS-P/SSL, то данные по умолчанию передаются в зашифрованном виде.)
- Если в качестве способа защиты соединения для передачи управляющей информации выбрано значение 2, а опция шифрования передаваемых данных равна 1, то для передачи данных клиенту придется применять дополнительные команды FTP. (Если установлен способ защиты TLS-C/TLS, то по умолчанию данные передаются в незашифрованном виде.)

Программа шифрования соединения для передачи управляющей информации

Указывает, какая программа для шифрования SSL применяется при передаче управляющей информации в данном сеансе FTP. Возможные варианты программ шифрования перечислены в определении API SSL. Информация об этих API приведена в разделе API SSL, который расположен в категории **Программирование** справочной системы iSeries Information Center.

Примечания:

- Программа выхода может только считывать данные из этого поля. Все изменения, вносимые программой выхода, игнорируются.
- Это значение допустимо только в том случае, если в качестве способа защиты управляющей информации выбрано значение 1 или 2.

Программа для шифрования передаваемых данных

Задаёт программу шифрования SSL, которая применяется для передачи данных в сеансе FTP. При вызове программы выхода этому параметру присваивается значение 0, означающее, что программа шифрования должна быть согласована при настройке соединения SSL. Программа выхода может изменить это значение, указав одну из поддерживаемых программ шифрования.

Возможные варианты программ шифрования перечислены в определении API SSL. Информация об этих API приведена в разделе API SSL, который расположен в категории **Программирование** справочной системы iSeries Information Center.

Примечания:

- Это поле игнорируется, если в поле Способ защиты управляющей информации указано значение 0 или в поле Опция шифрования передаваемых данных указано значение -1.
- Если значение этого поля будет отлично от нуля или совпадать с программой для шифрования управляющей информации, то при настройке соединения между клиентом и сервером FTP может возникнуть ошибка, так как указанная программа шифрования может не поддерживаться клиентом FTP.

Формат точки выхода TCPL0300:

Имя формата точки выхода: TCPL0300

Имя точки выхода: QIBM_QTMF_SVR_LOGON

Имя точки выхода: QIBM_QTMX_SVR_LOGON

Группа обязательных параметров:

1	Идентификатор приложения	Входной	Binary(4)
2	Идентификатор пользователя	Входной	Char(*)
3	Длина идентификатора пользователя	Входной	Binary(4)
4	Строка с идентификационной информацией	Входной	Char(*)
5	Длина строки с идентификационной информацией	Входной	Binary(4)
6	CCSID строки с идентификационной информацией	Входной	Binary(4)
7	IP-адрес клиента	Входной	Char(*)
8	Длина IP-адреса клиента	Входной	Binary(4)
9	Разрешить вход в систему	Выходной	Binary(4)
10	Пользовательский профайл	Выходной	Char(10)
11	Пароль	Выходной	Char(*)
12	Длина пароля	Выходной	Binary(4)
13	CCSID пароля	Выходной	Binary(4)
14	Начальная библиотека	Входной/выходной	Char(10)
15	Начальный домашний каталог	Выходной	Char(*)
16	Длина имени начального домашнего каталога	Входной/выходной	Binary(4)
17	CCSID начального домашнего каталога	Входной/выходной	Binary(4)
18	Информация о приложении	Входной/выходной	Char(*)

1	Идентификатор приложения	Входной	Binary(4)
19	Длина информации о приложении	Входной	Binary(4)

Описания параметров

Идентификатор приложения

Входной; BINARY(4) Задаёт сервер приложения, от которого получен запрос. Допустимые значения:

- 1**
Сервер FTP
- 2**
Сервер REXEC

Идентификатор пользователя

Входной; CHAR(*) Идентификатор пользователя, переданный программой-клиентом.

При работе с сервером FTP этот параметр содержит информацию, переданную в команде USER.

Длина идентификатора пользователя

Входной; BINARY(4) Задаёт длину идентификатора пользователя (в байтах).

Строка с идентификационной информацией

Входной; CHAR(*) Строка, переданная программой-клиентом (например, пароль).

При работе с сервером FTP в этом параметре указывается информация, переданная в команде PASS (пароль), если для идентификации пользователя не применялся сертификат клиента.

Длина строки с идентификационной информацией

Входной; BINARY(4) Длина строки с идентификационной информацией (в байтах).

CCSID строки с идентификационной информацией

Входной; BINARY(4) Идентификатор набора символов, связанный со строкой с идентификационной информацией. Если при работе с сервером FTP пользователь был идентифицирован с помощью сертификата клиента, то этому параметру присваивается значение -2.a

IP-адрес клиента

Входной; CHAR(*) IP-адрес, от которого получен запрос на открытие сеанса. Значение должно быть задано в десятичном формате с точками и выровнено по левому краю.

Длина IP-адреса клиента

Входной; BINARY(4) Задаёт длину IP-адреса клиента (в байтах).

Разрешить вход в систему

Выходной; BINARY(4) Указывает, следует ли разрешить вход в систему, и задаёт способ идентификации с помощью пароля. Допустимые значения:

- 0** Запретить вход в систему. Остальные выходные параметры игнорируются.

1 Разрешить вход в систему с указанным идентификатором пользователя и строкой идентификационной информации. В качестве имени пользовательского профайла будет применяться идентификатор пользователя, а в качестве пароля - строка с идентификационной информацией. В качестве текущей библиотеки и рабочего каталога применяются значения выходных параметров. Приложение игнорирует значения выходных параметров, задающих имя пользовательского профайла и пароль.

Примечание: Для успешного входа в систему строка с идентификационной информацией должна совпадать с паролем пользовательского профайла.

2 Разрешить вход в систему. В качестве имени и пароля пользовательского профайла будут применены возвращаемые значения, указанные в соответствующих выходных параметрах программы выхода. В качестве текущей библиотеки и рабочего каталога применяются соответствующие значения выходных параметров.

Примечание: Для успешного входа в систему указанный в выходном параметре пароль должен совпадать с паролем пользовательского профайла. **Внимание!** Фирма IBM настоятельно рекомендует **никогда** не задавать пароли непосредственно в программах выхода. Существуют алгоритмы для определения зашифрованных паролей.

3 Разрешить вход в систему. В качестве имени пользовательского профайла применять возвращаемое значение, указанное в соответствующем выходном параметре данной программы выхода. В качестве текущей библиотеки и рабочего каталога применяются соответствующие значения выходных параметров. Значение выходного параметра, задающего пароль, игнорируется.

Примечание: В системе с уровнем защиты не ниже 20 это значение переопределяет обычную процедуру проверки пароля OS/400. Проверка пароля выполняется только один раз.

Пользовательский профайл

Выходной; CHAR(10) Имя пользовательского профайла, который должен применяться в данном сеансе. Значение параметра должно быть выровнено по левому краю и дополнено справа пробелами.

Пароль

Выходной; CHAR(*) Пароль, который должен применяться в данном сеансе. Вместе с этим параметром необходимо задать параметры Длина пароля и CCSID пароля. Значение параметра должно быть выровнено по левому краю. Если системное значение QPWDLVL равно 0 или 1, то длина пароля не должна превышать 10 символов. Если системное значение QPWDLVL равно 2 или 3, то пароль может содержать до 128 символов.

Длина пароля

Выходной; BINARY(4) Длина пароля (в байтах). Допустимы значения от 1 до 512 байт.

CCSID пароля

Выходной; BINARY(4) Идентификатор набора символов, связанный с паролем. Это значение обязательно должно быть задано программой выхода, если указан пароль. Допустимые значения:

0

Для определения CCSID данных применяется CCSID задания. Если CCSID задания равен 65535, то применяется CCSID задания по умолчанию, указанный в атрибуте DFTCCSID.

1-65533

Допустимые значения CCSID.

Начальная библиотека

Выходной; CHAR(10) Начальная библиотека, которая должна применяться в данном сеансе. Значение параметра должно быть выровнено по левому краю и дополнено справа пробелами. При вызове программы выхода этому параметру присваивается следующее специальное значение: *CURLIB- Использовать текущую библиотеку, указанную в пользовательском профайле.

Начальный домашний каталог

Выходной; CHAR(*) Начальный домашний каталог, который должен применяться в данном сеансе. В качестве значения параметра должен быть задан полный путь к каталогу. Вместе с этим параметром необходимо задать параметры Длина имени начального домашнего каталога и CCSID начального домашнего каталога.

Длина начального домашнего каталога

Входной/выходной параметр; BINARY(4) Длина имени начального домашнего каталога, возвращаемого программой выхода. При вызове программы выхода этому параметру присваивается нулевое значение. Если программа выхода не изменяет значение этого параметра, то в качестве домашнего каталога применяется каталог, указанный в пользовательском профайле.

CCSID начального домашнего каталога Выходной; BINARY(4) Идентификатор набора символов, связанный с начальным домашним каталогом. Это значение обязательно должно быть задано программой выхода, если указан начальный домашний каталог. Допустимые значения:

0

Для определения CCSID данных применяется CCSID задания. Если CCSID задания равен 65535, то применяется CCSID задания по умолчанию, указанный в атрибуте DFTCCSID.

1-65533

Допустимые значения CCSID.

Информация о приложении

Входной/выходной; CHAR(*) Информация для задания параметров входа в систему, предусмотренных для конкретного приложения. Информация о формате этого значения приведена в разделе Формат информации о приложении.

Длина информации о приложении

Входной; BINARY(4) Длина (в байтах) информации о приложении.

Дополнительная информация приведена в следующих разделах:

- Точка выхода и процедуры входа в систему сервера: Точка выхода из процедуры входа в систему сервера приложения TCP/IP предназначена для идентификации пользователей.

Удаление программ выхода

Для удаления установленной программы выхода выполните следующие действия:

1. Введите в командной строке iSeries **WRKREGINF**.
2. Найдите запись для точки выхода из процедуры входа в систему сервера FTP:

```
QIBM_QTMF_SERVER_REQ  VLRQ0100
QIBM_QTMF_SVR_LOGON   TCPL0100
QIBM_QTMF_SVR_LOGON   TCPL0200
QIBM_QTMF_SVR_LOGON   TCPL0300
```

3. В поле Опц слева от записи точки выхода укажите **8** и нажмите **Enter**.

4. В меню **Работа с программой выхода** введите **4** (Удалить).
5. Введите имя программы выхода в поле **Программа выхода**.
6. В поле Библиотека укажите имя библиотеки, в которой находится программа выхода.
7. Нажмите **Enter**.
8. После удаления точек выхода перезапустите сервер FTP.

Способы передачи данных

Перед отправкой файлов нужно выбрать тип передачи файлов. Вы можете оставить тип по умолчанию (ASCII) или выбрать другой тип. ASCII - это стандарт Internet, применяемый для кодирования символов. EBCDIC - это стандарт для системы iSeries. Выберите тип передачи файлов с учетом следующих рекомендаций:

- Тип ASCII может применяться для передачи текстовых файлов (файлов, содержащих “только текст”).
- Тип EBCDIC может применяться для передачи данных EBCDIC в том случае, если локальная и удаленная системы поддерживают EBCDIC. В этом случае не потребуется дважды преобразовывать данные из кодировки EBCDIC в ASCII и обратно.
- Тип BINARY может применяться для передачи файлов, содержащих не текст, а другую информацию, например, двоичных файлов, графических файлов или файлов сохранения системы iSeries.

Выбрав тип передачи файлов, вы можете Передать файл по FTP.

Следующие разделы содержат дополнительную информацию об отдельных типах файлов:

- Передача файлов, содержащих упакованные десятичные данные, с одного сервера iSeries на другой
- Передача файлов *SAVF
- Передача документов QDLS
- Передача файлов “root”, QOpenSys и QLANSrv
- Передача файлов с помощью QfileSvr.400
- Передача файлов Qsys.lib
- Информация о предварительном создании файлов
- Преобразование CCSID

Передача файлов, содержащих упакованные десятичные данные, с одного сервера iSeries на другой

➤ FTP не поддерживает преобразование таких специальных числовых форматов, как упакованный десятичный и зонный десятичный.

Передача упакованных десятичных и зонных десятичных данных с одного сервера iSeries на другой поддерживается способами передачи TYPE I (BINARY) и TYPE E (EBCDIC) в режиме передачи BLOCK; при этих способах передачи данные отправляются без преобразования. В случае применения других способов передачи результаты могут быть непредсказуемы.

При передаче упакованных или зонных данных во внешне описанном файле QSYS.LIB целевой файл должен быть создан заранее и тем же способом, что и исходный файл. Это ограничение относится к данным, содержащим специальные числовые форматы, а также к передаче с доступом по ключу.

При двоичном способе передачи данных длина записей целевого файла должна совпадать с длиной записей исходного файла.

Для передачи упакованных десятичных или зонных десятичных данных в системы с другой архитектурой (например S/390^R или UNIX) или из таких систем необходимо преобразовать данные в текстовый формат.



Передача файлов *SAVF

» Файлы *SAVF передаются как образы, поэтому перед запуском команд GET и PUT необходимо запустить команду FTP BINARY.

При передаче файла *SAVF с применением формата имен 0 файл в целевой системе должен быть создан заранее. Для обеспечения целостности и максимальной производительности предварительное создание файлов рекомендуется и в других ситуациях.

Так как файлы сохранения применяются только в системах iSeries, то такие файлы имеет смысл передавать только с одного сервера iSeries на другой. Однако файлы сохранения можно отправлять и на серверы, отличные от iSeries, для резервного хранения. Впоследствии такие файлы можно будет передать на сервер iSeries с помощью FTP.

Пример: Передача файла *SAVF из системы VM на сервер iSeries

В приведенном ниже примере продемонстрирована передача файла *SAVF из системы VM на сервер iSeries в форматах имен NAMEFMT 0 и 1. Сеанс FTP уже открыт, вызвана команда BINARY и указано значение NAMEFMT 0.

Сначала необходимо передать файл P162484 SAVF310L с диска A VM на сервер iSeries. В FTP VM имя файла следует отделять от его типа точкой. Укажите имя файла P162484 в библиотеке P162484 на сервере iSeries, а также значение REPLACE, так как файл был создан предварительно, даже если он ранее не применялся. Предварительное создание обязательно для формата имен NAMEFMT 0.

Измените значение NAMEFMT на 1 и повторно передайте файл, применяя новый формат имен. Еще раз укажите REPLACE, так как файл был создан на предыдущем шаге.

Примечания:

- Если не создать файл на сервере iSeries до передачи с применением формата имен NAMEFMT 0, то сообщений об ошибках показано не будет. Однако при проверке файла на сервере iSeries окажется, что вместо файла сохранения (*SAVF) был создан физический файл (*PF).
- В зависимости от способа передачи файла *SAVF в систему VM, в ней может потребоваться выполнить дополнительные операции:
 - Если файл *SAVF был передан в систему VM по протоколу FTP, то для его отправки обратно на сервер iSeries достаточно вызвать команду GET.
 - Если файл *SAVF был отправлен в систему VM с помощью команды Отправить сетевой файл (SNDNETF), то сначала необходимо преобразовать этот файл в системе VM из переменного формата записи (RECFM) в фиксированный и лишь затем отправить файл с помощью FTP на сервер iSeries. Для этого в системе VM необходимо вызвать команду COPYFILE. Пример:

```
COPYFILE P162484 SAVF310L A = = = (RECFM F REPLACE
```

```

> GET P162484.SAVF310L P162484/P162484 (REPLACE
200 Запрос порта ОК.
150 Отправка файла 'P162484.SAVF310L'
250 Передача выполнена.
384912 байт передано за 3,625 секунды. Скорость передачи 106,183 Кб/с

> namefmt 1
202 Команда SITE необязательна; можно продолжать
NAMEFMT клиента - 1.
> GET P162484.SAVF310L/QSYS.LIB/P162484.LIB/P162484.savf (REPLACE
200 Запрос порта ОК.
150 Отправка файла 'P162484.SAVF310L'
250 Передача выполнена.
384912 байт передано за 3,569 секунды. Скорость передачи 107,839 Кб/с
Введите команду FTP.
====>

```

Рисунок 1. Передача файла *SAVF из системы VM на сервер iSeries с применением NAMEFMT 0 и NAMEFMT 1.



Передача документов QDLS

» В случае передачи документов QDLS всем документам, кроме текстовых документов исправимой формы (RFT), по умолчанию присваивается тип PCFILE на целевом сервере iSeries. Документам RFT по умолчанию присваивается тип RFTDCA. Документы типа RFTDCA можно просматривать и редактировать с помощью команды CL WRKDOC. Изменять и просматривать документы типа PCFILE с помощью этой команды нельзя.



Передача файлов “root”, QOpenSys, QLANSrv, QDLS и QOPT

» При передаче файлов файловых систем “root”, QOpenSys, QLANSrv, QDLS и QOPT следует применять потоковый режим (MODE S) и файловую структуру (STRUCT F).

Файлы “root”, QOpenSys, QDLS и QOPT могут содержать информацию в любой кодовой странице. К файлам, передаваемым в файловую систему QLANSrv, добавляется метка кодовой страницы, определенной для каталога этих файлов.

Преобразование данных и присваиваемые CCSID зависят от применяемого типа передачи. Дополнительная информация приведена в разделе Теги кодовых страниц CCSID для файлов iSeries. При передаче в файловую систему QLANSrv не поддерживается тип передачи TYPE E.

При добавлении данных в существующий файл метка CCSID этого файла не изменяется. Данные, добавляемые в существующий файл с применением типа передачи TYPE A, преобразуются в кодовую страницу данного файла.



Передача файлов с помощью QfileSvr.400

» Файловая система QFileSvr.400 обеспечивает доступ к другим файловым системам, расположенным на удаленных серверах iSeries. Поддерживается передача файлов в файловые системы “root”, QOpenSys, QLANSrv, QDLS и QOPT. Передача файлов в файловые системы QSYS.LIB не поддерживается.

Следует применять потоковый режим (MODE S) и файловую структуру (STRUCT F). Например, на рисунке 9-30 файл FILE.ABC передается между тремя различными файловыми системами в системе AS012 с помощью файловой системы QFileSvr.400 в системе AS009.

После подключения к системе AS009 с помощью команд клиента FTP, приведенных на рисунке 9-31, выполняется передача данных.

Примечание: ИД пользователя и пароль в системах AS009 и AS012 должны совпадать.

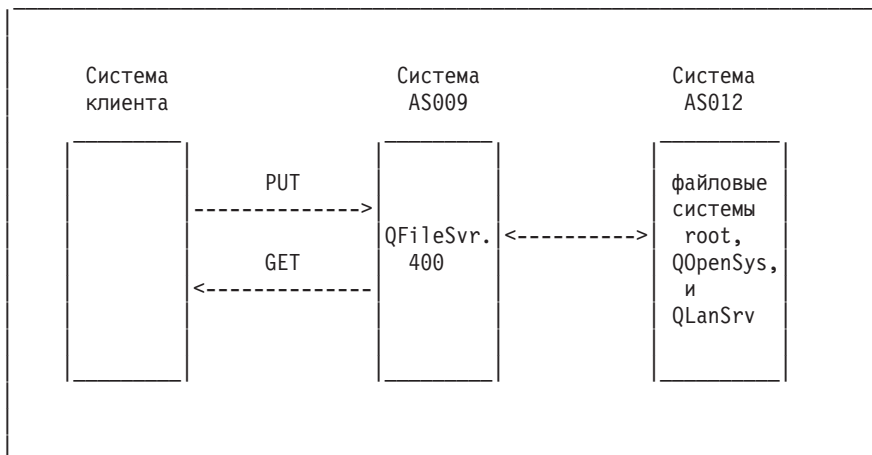


Рисунок 9-30. Пример файловой системы QFileSvr.400

```
NAMEFMT 1
LCD /CLIENTDIR1
CD /QFileSvr.400/AS012/FLSDIR
PUT FILE.ABC
GET FILE.ABC /CLIENTDIR2/FILE.ABC
CD /QFileSvr.400/AS012/QOpenSys/FLSDIR
PUT FILE.ABC
GET FILE.ABC /CLIENTDIR2/FILE.ABC (REPLACE
CD /QFileSvr.400/AS012/QLANSrv/NWS/LANSRV/DSK/K/FLSDIR
PUT FILE.ABC
GET FILE.ABC /CLIENTDIR2/FILE.ABC (REPLACE
SYSCMD RMLNK '/CLIENTDIR2/FILE.ABC'
DELETE /QFileSvr.400/AS012/FLSDIR/FILE.ABC
DELETE /QFileSvr.400/AS012/QOpenSys/FLSDIR/FILE.ABC
DELETE /QFileSvr.400/AS012/QLANSrv/NWS/LANSRV/DSK/K/FLSDIR/FILE.ABC
QUIT
```

Рисунок 9-31. Команды для передачи файлов с помощью файловой системы QFileSvr.400

Передача файлов QSYS.LIB

Приведенные ниже таблицы 1 и 2 содержат описание операций FTP с потоковым режимом передачи и типом передачи изображения для файловой системы QSYS.LIB. При работе с этими таблицами следует учитывать следующее:

Совместимые длины записи и размеры файлов

При передаче данных в существующий файл длина записи и размер целевого и передаваемого файлов должны быть совместимыми, иначе возникнет ошибка передачи данных. Длина записей и размер целевого файла должны быть не меньше соответствующих параметров исходного файла. Для того чтобы определить, совместим ли размер существующего файла, необходимо выяснить текущее число записей, разрешенное число расширений и максимальный размер записи. Для просмотра этой информации можно воспользоваться командой Показать описание файла (DSPFD) iSeries.

Автоматическое создание файлов на сервере iSeries

При получении файла сервер iSeries автоматически создает физический файл, если последний не существует. Тем не менее, рекомендуется заранее создать файл на сервере iSeries.

Тип данных

При передаче с применением типа TYPE I преобразование данных не выполняется. Несуществующие файлы создаются с CCSID 65535.

Примечание:

Настоятельно рекомендуется заранее создавать файлы в случае передачи файлов, состоящих из нескольких элементов, с помощью команд MGET и MPUT. В противном случае, FTP создаст файл с максимальным размером записи, равным длине самой большой записи первого обработанного элемента. Если в последующих элементах будут обнаружены более длинные записи, произойдет ошибка усечения данных. Предварительное создание файла с необходимым размером записи позволяет избежать этой ошибки.

Таблица 1: Потоковый режим передачи для файловой системы QSYS.LIB

Библиотека существует	Файл существует	Элемент существует	Выбрана замена	Совместимая длина записи	Совместимый размер файла	Результат
Да	Да	Да	Да	Да	Да	Данные записываются в элемент.
Да	Да	Да	Нет	нд	нд	Передача не выполняется, отправляется сообщение.
Да	Да	Нет	нд	Нет	Да	Передача файла выполняется, записи усекаются и возвращается сообщение.
Да	Да	Нет	Да	Нет	Да	Передача файла выполняется, записи усекаются и возвращается сообщение.
Да	Да	Нет	нд	Да	Да	Создается элемент, и в него записываются данные.
Да	Да	Нет	Нет	нд	Нет	Передача не выполняется, отправляется сообщение.
Да	Нет	нд	нд	нд	нд	Создается файл с длиной записи, равной максимальной длине записи исходного файла. Создается элемент, и в него записываются данные.
Нет	нд	нд	нд	нд	нд	Передача не выполняется, отправляется сообщение. Создайте библиотеку на удаленном сервере iSeries с помощью команды CRTLIB.

Таблица 2: Тип передачи изображения для файловой системы QSYS.LIB

Библиотека существует	Файл существует	Элемент существует	Выбрана замена	Результат
Да	Да	Да	Да	Данные записываются в элемент.
Да	Да	Да	Нет	Передача не выполняется, отправляется сообщение.
Да	Да	Нет	нд	Создается элемент, и в него записываются данные.
Да	Нет	нд	нд	
Нет	нд	нд	нд	



Копирование текстовых файлов в библиотеку QSYS.LIB: ➤ Так как в файловой системе iSeries QSYS.LIB предусмотрена внутренняя поддержка структуры записей, FTP iSeries преобразует файлы, получаемые сервером iSeries, в структуру записей, а файлы, отправляемые с сервера iSeries, - в файловую структуру FTP. Текстовые файлы, получаемые сервером iSeries, преобразуются протоколом FTP в структуру записей следующим образом:

- При получении файла, существующего на сервере iSeries, FTP применяет длину записи этого файла.
- При создании нового файла на сервере iSeries FTP применяет длину (исключая конечные пробелы) самой большой строки или записи этого файла.

При передаче текстовых файлов с сервера iSeries FTP преобразует их в файловую структуру, удаляя из каждой строки или записи конечные пробелы и отправляя усеченные записи.



Информация о предварительном создании файлов

➤ Настоятельно рекомендуется заранее создавать целевые файлы при передаче в файловую систему QSYS.LIB iSeries. Таким образом обеспечивается целостность данных и их надежная и эффективная передача с максимальной производительностью.

Выделите достаточное количество записей для всего файла. В системе iSeries для этого предназначен параметр SIZE команды Создать физический файл (CRTPF).

Убедитесь в соответствии значения параметра RCDLEN команды Создать физический файл (CRTPF) ожидаемой максимальной длине записи.

Примечание: Заранее создать файл в системе сервера FTP можно с помощью команды QUOTE, в системе клиента FTP - с помощью команды SYSCMD.



Преобразование CCSID

» Система iSeries интерпретирует входные данные и определяет формат вывода выходных данных согласно Идентификатору набора символов (CCSID). Входные данные могут быть предоставлены в формате ASCII или EBCDIC. Следующие разделы содержат подробную информацию о преобразовании CCSID:

- Задание таблиц преобразования
- Теги кодовых страниц CCSID для файлов iSeries
- Информация о NLS, относящаяся к FTP



Задание таблиц преобразования: » Таблицы преобразования ASCII для клиента FTP задаются в команде FTP. Для сервера FTP эти таблицы задаются в команде Изменить атрибуты FTP (CHGFTPA). Для того чтобы задать таблицы преобразования клиента FTP, выполните следующие действия:

1. Введите команду FTP.
2. Нажмите **PF4**. Будет показано меню **Запустить FTP TCP/IP**.
3. Нажмите **F10**. Будет показано приглашение для ввода таблиц преобразования ASCII/EBCDIC передаваемых и принимаемых данных.

```

                                Запустить протокол передачи файлов TCP/IP (FTP)
Введите опции, нажмите Enter.
Удаленная система . . . . .

IP-адрес . . . . .
Идентификатор набора символов      *DFT      1-65533, *DFT

                                Дополнительные параметры
Таблица преобраз. EBCDIC/ASCII .   *CCSID    Имя, *CCSID, *DFT
Библиотека . . . . .                Имя, *LIBL, *CURLIB
Таблица преобраз. ASCII/EBCDIC .   *CCSID    Имя, *CCSID, *DFT
Библиотека . . . . .                Имя, *LIBL, *CURLIB

                                                                Конец
F3=Выход  F4=Приглаш. F5=Обновить  F12=Отмена  F13=Как работать с этим меню
F24=Доп. клавиши
```

Рисунок 1. Задание таблиц преобразования ASCII с помощью значения *CCSID

Укажите CCSID (т.е. таблицы преобразования) для клиента FTP. Если значение *DFT не изменено, то применяется значение CCSID 00819 (ISO 8859-1 8-разрядный ASCII). Кроме того, можно указать конкретный CCSID для преобразования передаваемых и принимаемых данных. Дополнительная информация о применении CCSID приведена в разделе Информация о поддержке национальных языков, относящаяся к FTP.

Примечания:

- Значения CCSID наборов двухбайтовых символов (DBCS) нельзя указывать в параметре CCSID команды CHGFTPА. Значения DBCS CCSID можно указать с помощью команды TYPE (Задать тип передачи файлов).
- Поддержка преобразования включена в FTP фирмой IBM для обеспечения совместимости с выпусками, предшествующими V3R1. Применение таблиц преобразования для передачи типа А входящих файлов приводит к утере меток CCSID, если при этом создается целевой файл. Фирма IBM настоятельно рекомендует применять поддержку CCSID в обычных операциях.



Теги кодовых страниц CCSID для файлов iSeries: ➤ При создании нового файла на сервере iSeries протокол FTP для идентификации символьных данных добавляет к файлу метку CCSID или кодовой страницы этого CCSID. При замене данных или добавлении данных в существующий файл метка этого файла не изменяется. Приведенная ниже таблица содержит информацию о значениях, присваиваемых протоколом FTP для различных файловых систем и типов передачи.

Таблица 1: Теги кодовых страниц CCSID файлов iSeries

Целевая файловая система	Тип передачи А (ASCII)	Тип передачи С ('ccsid')	Тип передачи Е (EBCDIC)	Тип передачи I (изображение/ двоичный)
QSYS.LIB	CCSID, указанный в параметре Идентификатор набора символов EBCDIC для новых файлов базы данных (CRTCCSID).	'ccsid' при CCSID EBCDIC. Если указан CCSID ASCII, то применяется связанный CCSID EBCDIC по умолчанию.	65535	65535
"root", QOpenSys, QDLS, QOPT	CCSID ASCII по умолчанию.	Значение 'ccsid', указанное в команде TYPE С ccsid#.	CCSID задания, если он не равен 65535. При CCSID задания 65535, - CCSID задания по умолчанию.	CCSID ASCII по умолчанию.
QLanSrv	Кодовая страница ASCII описания сетевого сервера для каталога файлов.	Кодовая страница ASCII описания сетевого сервера для каталога файлов.	Не поддерживается.	Кодовая страница ASCII описания сетевого сервера для каталога файлов.

Примечание:

CCSID ASCII по умолчанию определяется при запуске задания FTP: для клиента - параметром CCSID команды STRTCPFTP (и FTP), для сервера - параметром CCSID атрибутов конфигурации FTP. Этот параметр можно изменить командой CHGFTPА. Значения для файлов QFileSvr.400 зависят от целевой файловой системы.



Информация о NLS, относящаяся к FTP: ➤ Ниже перечислены особенности работы с FTP в средах с другими основными языками.

- При передаче типа Е (и EBCDIC) данные сохраняются 'как есть', т.е. в кодовой странице EBCDIC исходного файла. Это может привести к добавлению неправильной метки CCSID, если основные языки двух серверов iSeries различаются.

Например, при отправке данных в кодовой странице 237 с применением типа передачи E в файловую систему QSYS.LIB на сервере, где этот файл не существует, данные сохраняются в новом файле с меткой CCSID 65535. Если целевой файл уже существует, то данные будут приняты 'как есть' и снабжены CCSID существующего файла, который может отличаться от 237.

Во избежание добавления неверных меток CCSID можно указать CCSID передаваемых данных в команде TYPE C CCSID (например, TYPE C 237). Если в команде передачи указан CCSID, то при записи в существующий файл данные преобразуются в CCSID этого файла. В случае, если файл не существует, в создаваемый файл добавляется метка указанного CCSID.

В предыдущем примере, если файл не существует, то в целевой системе создается файл со значением CCSID 237. Если целевой файл уже существует, то данные преобразуются из CCSID 237 в CCSID целевого файла.

- При запуске клиента FTP может быть выдано сообщение TCP3C14: Не удается преобразовать данные из CCSID &1 в CCSID &2. Это может произойти в том случае, если преобразование из CCSID EBCDIC задания в CCSID ASCII данного сеанса FTP и обратно не поддерживается.

Значение CCSID ASCII можно изменить с помощью параметра Идентификатор набора символов команды CL STRTCPFTP. Для всех допустимых значений CCSID заданий поддерживается преобразование в CCSID 850, содержащий набор символов IBM PC Latin-1.

- Когда файлы передаются по протоколу FTP в режиме ASCII из одной системы EBCDIC в другую, данные в исходной системе преобразуются из кодовой страницы EBCDIC в ASCII, а затем - из ASCII в кодовую страницу EBCDIC целевой системы. Обычно это не вызывает проблем, поскольку обе системы применяют одну и ту же кодовую страницу 7-разрядного ASCII; исключение составляет случай, когда символы EBCDIC в исходной системе не определены в кодовой странице ASCII. Кроме того, некоторые символы кодовой страницы ASCII могут преобразовываться по-разному в разных кодовых страницах EBCDIC. Это возможно в том случае, если некоторые символы ASCII являются переменными (символу соответствует другой шестнадцатеричный код в кодовой странице EBCDIC). Переменный символ может быть интерпретирован неправильно, если кодовые страницы EBCDIC исходной и целевой систем не совпадают.



Файловые системы и соглашения о присвоении имен

Сервер FTP представляет информационные объекты файловой системы в виде многоуровневой иерархической структуры.

Набор файловых систем OS/400, поддерживаемых сервером FTP, зависит от выпуска системы iSeries. В файловых системах OS/400 для обозначения объектов данных и иерархических структур данных применяются разные названия.

Соглашения о присвоении имен

В каждой файловой системе OS/400 применяются собственные правила именования файлов. Формат имени файла должен соответствовать соглашениям о присвоении имен той файловой системы, в которой он расположен. Форматы и примеры имен файлов для файловых систем OS/400, поддерживаемых сервером FTP, описаны в разделе, посвященном IFS. Дополнительная информация приведена в разделе Работа с файловыми системами. Информацию о формате имен файлов в системах, отличных от iSeries, можно получить с помощью команды QUOTE HELP.

NAMEFMT сервера FTP

При запуске сервера FTP параметру NAMEFMT присваивается значение "0". Изменить значение NAMEFMT можно с помощью команды SITE.

Сервер автоматически изменяет значение по умолчанию NAMEFMT 0 на NAMEFMT 1, если 'первый' файл или каталог, указанный в команде:

- Начинается с символа косой черты (/) или тильды (~) или
- пуст (кроме команд LIST и NLST)

Последующие команды сервера с параметром файла или каталога не влияют на значение NAMEFMT. В случае изменения значения NAMEFMT это будет отражено в ответе сервера на команду.

Например, значение NAMEFMT будет изменено на "1", если первой командой сервера с параметром файла или каталога является:

```
CWD /DIR1/DIR2A
```

Ответ сервера будет следующим:

```
250-NAMEFMT изменено на 1.
250 Текущий каталог изменен на /DIR1/DIR2A.
```

Примечание: Эта функция позволяет Web-браузерам, для которых необходимо значение NAMEFMT 1, работать с серверами FTP iSeries, не передавая команду SITE NAMEFMT 1.

Дополнительная информация о значении NAMEFMT приведена в разделе Файловые системы и соглашения о присвоении имен.

Информация о применении команды NAMEFMT для настройки формата имен файлов приведена в разделе NAMEFMT (Выбор формата имен файлов).

Файловые системы OS/400, поддерживаемые FTP

Набор файловых систем, поддерживаемых FTP, зависит от выпуска сервера iSeries.

Библиотечная файловая система QSYS.LIB - библиотеки, файлы, элементы

FTP поддерживает передачу файлов сохранения и элементов физических файлов, логических файлов, файлов DDM и исходных физических файлов. При работе с файловой системой QSYS.LIB данные передаются в виде элементов физических файлов, хранящихся в библиотеке.

Библиотека документов QDLS - папки и документы

При работе с файловой системой библиотек документов (QDLS) передаваемые данные обычно называются документами. Документы QDLS располагаются в каталогах, которые называются папками.

"root"

Файловая система /. Эта файловая система обладает всеми преимуществами поддержки потоковых файлов и иерархической структуры каталогов интегрированной файловой системы. Она схожа с файловыми системами DOS и OS/2^R.

QOpenSys

Файловая система открытых систем. Эта файловая система полностью адаптирована под стандарты открытых систем UNIX, в частности, POSIX и XPG. Как и корневая файловая система, она поддерживает потоковые файлы и структуру каталогов интегрированной файловой системы. В ней учитывается регистр букв в именах файлов.

QOPT

Файловая система на оптических носителях QOPT. Она обеспечивает доступ к потоковым данным, хранящимся на оптическом носителе.

QFileSvr.400

Файловая система файлового сервера OS/400. Она обеспечивает доступ к другим файловым

системам, расположенным в удаленных системах iSeries. FTP не поддерживает доступ к файловым системам QSYS.LIB, QDLS и QOPT через QFileSvr.400.

Полная информация о файловых системах, поддерживаемых FTP, приведена в разделе Интегрированная файловая система.

Коды ответа сервера FTP

При вызове команды в сеансе клиента FTP на экране появляется сообщение о состоянии, представляющее собой трехзначный код: *xuz*.

Первая цифра (*x*) указывает тип ответа: положительный, отрицательный или неполный. Возможны пять различных значений:

- 1yz = Положительный. Запрошенная операция выполняется; будет отправлен еще один ответ.
- 2yz = Положительный. Запрошенная операция выполнена; сервер готов к приему следующего запроса.
- 3yz = Неполный. Команда принята, но выполнение операции заблокировано до получения дополнительной информации.
- 4yz = Неполный. Сервер не принял команду. Запрошенная операция не выполнена. Произошла случайная ошибка, можно повторить операцию.
- 5yz = Отрицательный. Команда была отклонена, а запрошенная операция не выполнена.

Вторая цифра (*y*) задает функциональную категорию ответа.

- x0z=Синтаксис. Обозначает синтаксические ошибки, а также недопустимые и лишние команды.
- x1z=Информация. Относится к запросам на получение справки или информации о состоянии.
- x2z=Соединения. Обозначает управляющие или информационные соединения.
- x3z=Идентификация. Относится к процедуре входа в систему.
- x5z=Файловая система. Относится к информации о состоянии сервера при выполнении запроса на передачу файла.

Третья цифра (*z*) уточняет функциональную категорию.

Ниже перечислены стандартные коды ответов и их описания. Текст сообщений зависит от типа сервера.

Код	Описание
110	Перезапуск маркера
120	Служба будет готова через npp минут
125	Соединение для передачи данных уже открыто; начинается передача данных
150	Файл запущен успешно; открывается соединение для передачи данных
200	Команда выполнена
202	Команда не выполнена; она не поддерживается системой
211	Информация о состоянии системы или справка сервера
212	Состояние каталога
213	Состояние файла
214	Справочное сообщение

Код	Описание
220	Служба готова к работе с новым пользователем
226	Закрытие соединения для передачи данных; запрошенная операция над файлом успешно выполнена
230	Пользователь вошел в систему
250	Запрошенная операция над файлом успешно выполнена
257	Каталог создан
331	Требуется пароль
332	Требуется учетная запись
425	Невозможно открыть соединение для передачи данных
426	Соединение закрыто; передача данных завершилась аварийно
450	Запрошенная операция не выполнена; файл используется
451	Запрошенная операция завершена аварийно; на сервере произошла ошибка
452	Запрошенная операция не выполнена; в системе недостаточно памяти
500	Синтаксическая ошибка; команда не распознана
501	Синтаксическая ошибка в параметрах или аргументах
502	Команда не поддерживается
503	Недопустимая последовательность команд
504	В команде не предусмотрен указанный параметр
530	Попытка входа в систему отклонена
532	Для сохранения файлов требуется учетная информация
550	Запрошенная операция не выполнена; файл не найден (или недоступен)
551	Запрошенная операция завершена аварийно; неизвестный тип страницы
552	Запрошенная операция завершена аварийно; недостаточный объем выделенной памяти
553	Запрошенная операция не выполнена; недопустимое имя файла

Правила описания формата команд сервера FTP

В приведенных в данном разделе описаниях команд сервера FTP применяются следующие соглашения о синтаксисе команд:

Прописные буквы

Прописными буквами выделяется обязательная часть команды. Сами буквы можно вводить без учета регистра.

Слова в нижнем регистре и термины, указанные через дефис

Слова, записанные строчными буквами или содержащие дефис, например, "файл" и "учетная-информация", представляют имена переменных, вместо которых нужно указать фактическое значение.

Квадратные скобки []

Слова, символы и словосочетания, указанные в квадратных скобках, в команде задавать не обязательно.

Открывающие круглые скобки (и звездочки *

При вводе команды открывающие круглые скобки и звездочки должны располагаться в тех же позициях, что и в определениях команд.

Фигурные скобки { }

Фигурными скобками обозначаются группы параметров, значений или переменных, которые могут повторяться.

Многоточие ...

Многоточие указывает, что предыдущая переменная, указанная в квадратных скобках, может быть задана ноль и более раз.

Вертикальная черта |

Вертикальная черта между параметрами или значениями указывает, что они взаимоисключающие, и их нельзя указать одновременно. Вертикальной чертой разделяются значения, указанные в квадратных или фигурных скобках.

Правила описания формата команд клиента FTP

В описании формата команд клиента FTP применяются следующие соглашения:

Прописные буквы

Прописные буквы в определении формата команд клиента задают обязательную часть команды. Сами команды клиента FTP можно вводить без учета регистра.

Слова, набранные строчными буквами, и термины, содержащие дефис

Слова, написанные строчными буквами, или термины, указанные через дефис, например "удаленный-файл" или "учетная-информация", являются переменными, вместо которых нужно подставить фактические значения.

Квадратные скобки []

Слова, символы и словосочетания, указанные в квадратных скобках, в команде задавать не обязательно.

Открывающие круглые скобки (и звездочки *

При вводе команды открывающие круглые скобки и звездочки должны располагаться в тех же позициях, что и в определении формата команды.

Фигурные скобки { }

Фигурными скобками обозначаются группы параметров, значений или переменных, которые могут быть заданы любое число раз.

Многоточие ...

Многоточие указывает, что предыдущая переменная, указанная в квадратных скобках, может быть задана ноль и более раз.

Вертикальная черта |

Вертикальная черта между параметрами или значениями указывает, что они взаимоисключающие, и их нельзя указать одновременно. Вертикальная черта может разделять параметры, заключенные в квадратные или фигурные скобки.

Дополнительные сведения о формате команд:

- Ограничители параметров команд: Здесь приведена информация о применении апострофов (') и кавычек (") для ограничения параметров команды.
- Имена файлов по умолчанию в командах передачи файлов клиента: Здесь приведена информация об именах файлов, которые применяются по умолчанию.
- Правила именования передаваемых файлов: В этом разделе приведена подробная информация о значениях параметров локальный-файл и удаленный-файл.

Параметры команд и знаки препинания

Вы можете заключать параметры команд в апострофы (') или кавычки ("). Если параметр, заключенный в апострофы, сам содержит апостроф, то этот апостроф должен записываться в виде двойного апострофа ("). Внутри параметра, заключенного в кавычки ("), апостроф удваивать не нужно.

Аналогичным образом в параметрах указываются кавычки ("):

- Кавычки (") не нужно удваивать в параметрах, заключенных в апострофы.
- Кавычки следует удваивать в параметрах, заключенных в кавычки (").

Вы можете руководствоваться следующим принципом:

1. Если в параметре присутствует символ, который применяется в качестве ограничителя, то этот символ нужно удвоить. Например:

```
'ABCD' '12345'  
соответствует значению ABCD'12345  
"ABCD" "12345"  
соответствует значению ABCD"12345
```

2. Если символ внутри параметра отличается от ограничителей параметра, то его не нужно удваивать. Например:

```
"ABCD'12345"  
соответствует значению ABCD'12345  
'ABCD"12345'  
соответствует значению ABCD"12345
```

3. Если в параметре присутствуют одновременно и кавычки, и апостроф, то следует выбрать один из этих символов в качестве ограничителя. Например:

```
"ABC'12" "345" и 'ABC'12"345'  
соответствуют значению ABC'12"345
```

Дополнительные сведения о формате команд:

- Правила описания формата команд клиента FTP

Имена файлов, применяемые в командах передачи файлов клиента

Если в командах PUT, APPEND и GET не указано имя целевого файла, то клиент FTP подставляет имя файла по умолчанию. Кроме того, поскольку в командах MPUT и MGET указываются имена исходных файлов, то для этих команд FTP также задает имена целевых файлов по умолчанию. Формат этих команд описан в приведенной ниже таблице Команды передачи данных. В колонке *Целевой файл* указан параметр, в котором задается имя по умолчанию.

Команда	Исходный файл	Целевой файл	Прочая информация
APPEND	имя локального файла	[имя файла сервера]	
PUT	имя локального файла	[имя файла сервера]	

Команда	Исходный файл	Целевой файл	Прочая информация
GET	имя файла сервера	[имя локального файла]	[(Replace)]
MPUT	имя локального файла		
MGET	имя файла сервера		[(Replace)]

PUT и APPEND

Правила создания имен по умолчанию для команд PUT и APPEND задаются в зависимости от типа сервера:

- Сервер iSeries
 - Если целевая файловая система - это библиотека файлов или документов, то имя по умолчанию создается в соответствии с правилами именования объектов этих файловых систем, с учетом формата имен.
 - Если целевая файловая система не является ни библиотекой файлов, ни файловой системой с документами, то в качестве имени по умолчанию принимается:
 - имя, следующее за последней косой чертой в имени исходного файла
 - имя, *совпадающее* с именем исходного файла, если в нем нет ни одной косой черты.
- Другой сервер
 - Если исходный файл находится в библиотечной файловой системе, то имя по умолчанию будет выглядеть следующим образом: *имя-файла.имя-элемента*. Если имя элемента не задано, то имя целевого файла будет совпадать с именем исходного файла.
 - Если исходный файл находится в библиотеке документов, то имя целевого файла будет состоять из имени файла и расширения.
 - Если исходный файл находится в другой файловой системе, то по умолчанию будет применяться часть его имени, указанная после последней косой черты. Если в имени исходного файла нет символов косой черты, то имя целевого файла будет совпадать с именем исходного файла.

При работе с сервером iSeries имя по умолчанию для этих команд будет задаваться по тем же правилам, что и для команды PUT.

GET и MGET

Серверы, отличные от iSeries, создают имя по умолчанию для команд GET и MGET на основе той части имени исходного файла, которая расположена после последней косой черты. Если в имени исходного файла нет символов косой черты, то имя целевого файла будет совпадать с именем исходного файла. Ниже приведены правила выбора имен по умолчанию:

- Если файловая система клиента - это *библиотечная файловая система* (база данных iSeries), то применяются следующие правила:
 - Если в имени удаленного файла есть точка (.), то имя локального файла будет состоять из первых 10 символов, указанных перед точкой. Имя элемента будет состоять из первых 10 символов, указанных после точки.
 - Если имя удаленного файла не содержит точки, то и для имени файла, и для имени элемента, составляющих имя локального файла, будут приняты первые 10 символов имени удаленного файла.
 - Если при этом будет выбран формат имен 1, то к именам файлов и элементов будут добавлены соответствующие расширения.
- Если файловая система клиента - это *библиотека документов*, то применяются следующие правила:

- Если имя удаленного файла содержит точку, то в качестве имени локального файла применяются первые восемь символов, расположенные до точки. После точки остаются только три символа.
- Если в имени удаленного файла нет точки, то имя локального файла будет содержать первые 8 символов имени удаленного файла, а расширения не будет.
- При работе с другими файловыми системами в качестве имени по умолчанию применяется та часть имени удаленного файла, которая расположена после последней косой черты.

Примечания:

1. В файлах сохранения нет элементов, поэтому в их именах по умолчанию отсутствует часть имени, задающая элемент.
2. Сервер показывает имена по умолчанию, когда включен режим DEBUG.

Дополнительные сведения о формате команд:

- Правила описания формата команд клиента FTP

Правила именования передаваемых файлов

В командах передачи файлов клиента FTP указываются параметры **локальный-файл** и/или **удаленный-файл**. Эти параметры задают имена передаваемых объектов. Предусмотрены следующие команды передачи файлов:

APPEND
локальный-файл [удаленный-файл]

DELETE
удаленный-файл

GET
удаленный-файл [локальный-файл]

MDELETE
удаленные-файлы

MGET
удаленные-файлы

MPUT
локальные-файлы

PUT
локальный-файл [удаленный-файл]

Имя локального и удаленного файла можно указывать как в полной, так и в сокращенной форме. Сокращенное имя объекта содержит собственно имя объекта, а также имена одного или нескольких вложенных каталогов иерархической ветви, содержащей этот объект. В полное имя входят все вложенные каталоги в иерархической файловой системе.

Если указано сокращенное имя, то будет выполнена обработка файла с таким именем, расположенного в текущем рабочем каталоге. Для перехода в другой рабочий каталог локальной системы клиента выполните команду LCD .Для перехода в другой рабочий каталог удаленной системы вызовите команду CD.

Имена локальных файлов, указанные в команде, должны быть заданы в соответствии с соглашениями о присвоении имен файлам iSeries. Имена удаленных файлов должны соответствовать соглашениям о присвоении имен, принятым в удаленной системе.

Дополнительные сведения о формате команд:

- Ограничители параметров команды: Параметры можно заключать в апострофы (') или кавычки (").
- Имена файлов по умолчанию в командах передачи файлов клиента: Здесь приведена информация об именах файлов, которые по умолчанию применяются в командах передачи данных клиента.
- Правила описания формата команд клиента FTP: Здесь описаны некоторые соглашения о формате команд клиента FTP.

Устранение неполадок FTP

Этот раздел содержит основные сведения об устранении неполадок FTP.

Обнаружение неполадок FTP

Содержит пошаговые инструкции для проверки правильности работы SMTP.

Данные, необходимые для создания отчета о неполадках FTP

Этот раздел содержит сведения о данных, которые могут потребоваться сервисному центру.

Трассировка сервера FTP

Команда DBUG позволяет отслеживать неполадки сервера FTP.

Трассировка клиентов FTP

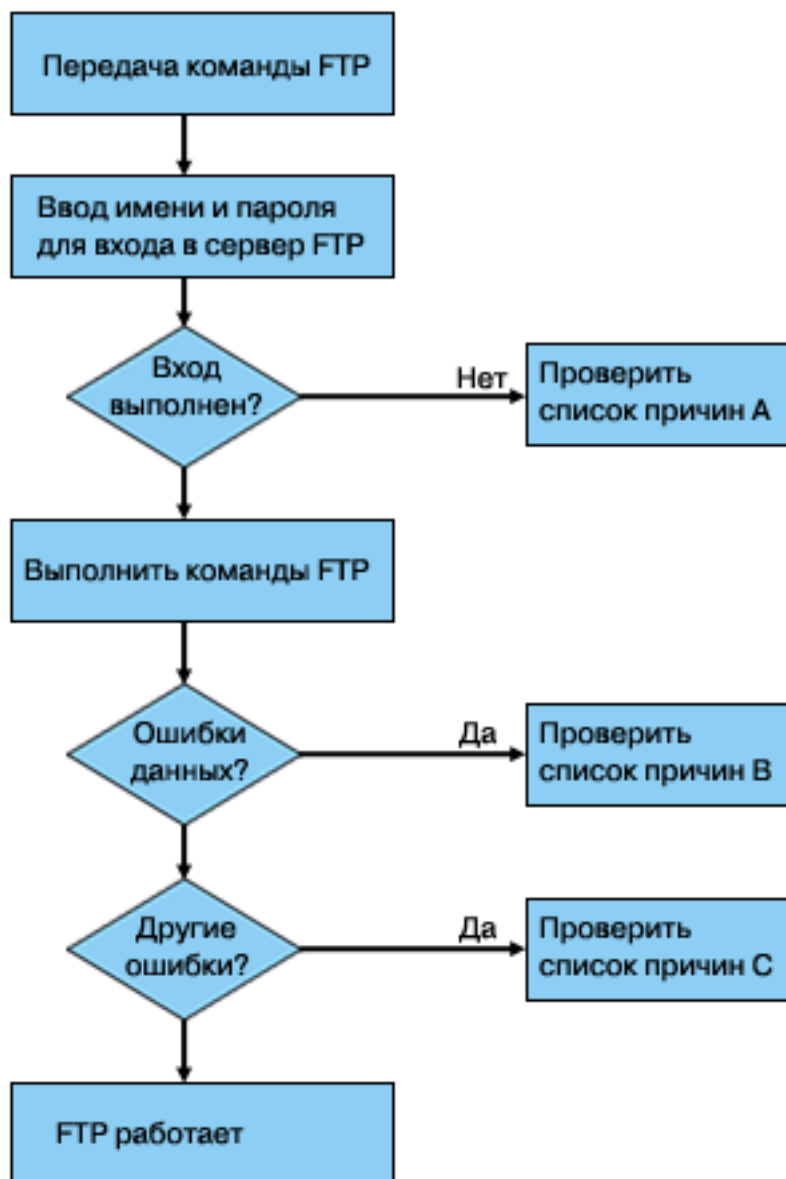
Команда DEBUG позволяет отслеживать неполадки клиента FTP.

Работа с протоколом задания сервера FTP

Этот раздел содержит информацию о применении буферного файла протокола задания сервера FTP для анализа неполадок.

Устранение неполадок FTP

Если при работе с протоколом FTP возникла неполадка, обратитесь к диаграмме со списком общих неполадок TCP/IP, а затем к приведенной ниже диаграмме. В приведенном списке возможных причин неполадки указаны рекомендуемые действия по обнаружению причин неполадки.



*

Анализ неполадок FTP

Список причин А

1. Сколько времени проходит между подключением к серверу FTP системы iSeries и появлением приглашения для ввода ИД пользователя? Если задержка велика, проверьте конфигурацию сервера имен доменов на сервере iSeries. Сервер FTP отправляет запрос серверу DNS при получении запроса на соединение. Из-за неполадок сервера DNS задержка в получении ответа может составлять несколько минут.
2. Проверьте, не задана ли программа выхода для входа в систему сервера FTP. Обратитесь к разделу Точка выхода из процедуры входа в систему сервера. Если да, проверьте, разрешает ли программа выхода подключение клиентов с неверным именем или паролем.
3. Если удаленная система запрашивает у вас пароль, выясните, действительно ли ввод пароля необходим. Некоторые системы запрашивают у пользователя пароль, однако ввод пароля приведет к отклонению запроса на соединение, так как на самом деле указывать пароль не нужно.

4. Задайте сведения о пароле в удаленной системе. Возможно, что после изменения информации о защите потребуются перезапустить систему.
5. Попробуйте войти в удаленную систему, указав свой ИД и пароль. Если войти в систему не удалось, обратитесь к администратору удаленной системы и попросите его проверить правильность ваших ИД и пароля.

Список причин В

1. При передаче двоичных файлов убедитесь, что включен двоичный режим передачи.
2. Убедитесь, что таблицы преобразования, установленные на клиенте и сервере, совместимы. Эта проверка обязательна только в том случае, если вы применяете пользовательские таблицы преобразования.
3. Убедитесь, что при передаче файлов применяется правильный CCSID. Если это не так, то перед началом передачи данных введите команду TYPE или LTYPE для настройки нужного CCSID.
4. Создайте файл в системе, в которой вы хотите сохранить передаваемые данные. Задайте нужные длину записи, число элементов и шаг приращения. Повторите операцию передачи данных и убедитесь, что она прошла успешно.
5. Убедитесь, что у вас есть права доступа, необходимые для работы с файлом и его элементами.
6. Проверьте, не содержит ли файл данные в упакованном или зонном десятичном формате.
7. При передаче файлов сохранения убедитесь, что применяется правильный способ передачи.

Список причин С

1. Проверьте ограничение на размер файла, принятое в удаленной системе.
2. Проверьте, не истекло ли время ожидания, заданное для сервера FTP. Время ожидания сервера iSeries можно задать с помощью команды QUOTE TIME.
3. С помощью команды NETSTAT убедитесь, что интерфейс *LOOPBACK активен. Вызовите команду FTP LOOPBACK (внутреннее соединение системы iSeries) и повторите действия, которые привели к возникновению неполадки.
 - Если неполадка не возникла снова, то, скорее всего, она была вызвана ошибкой в удаленной системе.
 - Если неполадка возникла снова, выполните следующие действия:
 - a. Если неполадка связана с работой сервера FTP, запустите трассировку сервера с помощью команды TRCTCPAPP.
 - b. Воспроизведите неполадку еще раз.
 - c. Закройте соединение FTP. За дополнительной информацией обратитесь к разделу Запуск и завершение работы сервера FTP.
 - d. Выключите трассировку сервера FTP с помощью команды TRCTCPAPP.
 - e. Найдите следующий буферный файл:
 - Имя файла - QTMFFTRC
 - С файлом должно быть связано имя пользователя, запустившего команду TRCTCPAPP.Информация трассировки заносится в буферный файл и помещается в применяемую по умолчанию очередь вывода системы, связанную с заданием сервера FTP.
 - f. Отправьте этот буферный файл.
 - g. Если неполадка связана с работой клиента FTP системы iSeries, соберите данные трассировки с помощью команды клиента DEBUG 100.
 - h. При интерактивной работе с клиентом FTP нажмите клавишу F6 (Печать) для создания буферного файла, который будет содержать хронологию команд клиента FTP и ответов сервера на эти команды. Если же клиент FTP работает в неконтролируемом пакетном режиме, хронология команд и ответов сервера сохраняется в заданном файле вывода. Дополнительная информация приведена в разделе "Запуск задания FTP в пакетном режиме".

Данные, необходимые для создания отчета о неполадках FTP

В отчет о неполадке FTP, предназначенный для отправки в фирму IBM, должны быть включены следующие данные:

- Информация трассировки, полученная в момент возникновения неполадки (относящаяся только к TCP/IP), сохраненная в форматах ASCII и EBCDIC.
- Если клиент или сервер FTP занесли в протокол сообщения об ошибке программы, эту информацию необходимо будет включить в отчет.

Примечание: Для того чтобы в протокол заносилась информация об ошибках программ, системное значение QSFWERRLOG должно быть равно *LOG. Если на момент возникновения сбоя системное значение QSFWERRLOG было равно *NOLOG, присвойте ему значение *LOG, попытайтесь воспроизвести неполадку, а затем включите в отчет занесенные в протокол сообщения об ошибке программного обеспечения. При наличии в протоколе сообщений об ошибке программы не нужно выполнять трассировку FTP.

- Протокол задания QTCP/IP, а также все протоколы заданий сервера или клиента FTP.
- Информация трассировки клиента FTP и сервера FTP.
- В случае неполадок, связанных с клиентом FTP, - буферный файл, содержащий протокол сеанса. Сохранить протокол можно во время сеанса FTP, нажав клавишу F6 (Печать).
- Если нарушена целостность данных, следует включить в отчет как текст файла, элемента или библиотеки, вызвавшей неполадку, так и копию описания этого файла, элемента или библиотеки.

Трассировка сервера FTP

Трассировку сервера FTP можно включить в любой системе, в которой установлен протокол TCP/IP. Существует два способа трассировки сервера FTP. Команда сервера FTP DBUG выполняет трассировку в течение сеанса работы с сервером FTP. Команда Трассировка приложения TCP/IP (TRCTCPAPP) собирает данные трассировки для всех серверов FTP, запущенных в системе.

Трассировка сервера FTP с помощью команды DBUG

Ниже приведен пример работы с командой DBUG сервера FTP:

```
Протокол передачи файлов

Предыдущие команды и сообщения FTP:
Подключение к хосту xxxxxnnn.xxxxxxxx.xxx.xxx с адресом
n.nnn.nn.nnn using через порт 21.
220-QTCP в системе xxxxxnnn.nnnnnnnn.nnn.nnn.
220 Соединение будет прервано, если время простоя составит больше 5 минут.
215 Удаленная операционная система - OS/400. Версия TCP/IP -
"V4R4M0".
>
331 Введите пароль.
230 Пользователь TEST вошел в систему.
250 Применяется формат имен "0".
257 Текущая библиотека - "QGPL".

Введите команду FTP.
===> quote dbug

F3=Выход      F6=Печать      F9=Восстановить
F17=Начало    F18=Конец      F21=Командная строка CL
```

Для включения трассировки сервера FTP:

1. Для включения трассировки введите команду QUOTE DBUG.

```

                                Протокол передачи файлов

Предыдущие команды и сообщения FTP:
Подключение к хосту xxxxxppp.xxxxxxxx.xxx.xxx с адресом
n.ppp.pp.ppp using через порт 21.
220-QTSP в системе xxxxxppp.xxxxxxxx.xxx.xxx.
220 Соединение будет прервано, если время простоя составит больше 5 минут.
215 Удаленная операционная система - OS/400. Версия TCP/IP -
"V4R4M0".
>
331 Введите пароль.
230 Пользователь TEST вошел в систему.
250 Применяется формат имен "0".
257 Текущая библиотека - "QGPL".
> quote dbug
250 Включен режим отладки.
Введите команду FTP.
===> quote dbug

F3=Выход      F6=Печать      F9=Восстановить
F17=Начало    F18=Конец     F21=Командная строка CL
```

2. Запустите операцию FTP, трассировку которой вы хотите выполнить.
3. Для выключения трассировки введите QUOTE DBUG. Данные трассировки будут записаны в буферный файл QTMFFTRC. Этот файл будет помещен в очередь вывода по умолчанию. В качестве пользователя всегда будет указываться пользователь, работавший с сервером FTP на момент завершения трассировки.
4. Для завершения сеанса FTP введите QUIT.
5. Для того чтобы определить, в какую очередь вывода был помещен буферный файл, введите следующую команду:
DSPSYSVAL QPRTDEV
Появится меню, аналогичное приведенному ниже:

```

                                Показать системное значение
Системное значение . . . : QPRTDEV
Описание . . . . . : Описание принтера
Печатающее устройство . . . . : PRT01          Имя
```

Имя принтера совпадает с именем системной очереди вывода по умолчанию.

6. Запишите имя принтера. В данном случае это имя принтера равно PRT01.
7. Нажмите F12 (Отмена) для возврата в меню, из которого вы вызвали команду DSPSYSVAL.
8. Введите следующую команду:
WRKOUTQ OUTQ(имя-принтера)
В качестве имени-принтера нужно указать имя, записанное на предыдущем шаге. В нашем примере именем очереди вывода также будет PRT01. Появится меню, аналогичное приведенному ниже:

Работа с очередью вывода								
Очередь: PRT01			Библиотека: QGPL			Состояние: RLS		
Введите опции, нажмите Enter.								
1=Отправить 2=Изменить 3=Блокировать 4=Удалить 5=Показать 6= Разблокировать								
7=Сообщения 8=Атрибуты 9= Работа с состоянием печати								
Опц	Файл	Польз.	Польз.данные	Сост.	Стр.	Копий	Тип формы	Приор.
-	QTCPPRT	QTCP	QTMSMTP	HLD	46	1	*STD	5
-	QTMFFTRC	QSECOFR		HLD	44	1	*STD	5

9. Если в нижней части меню показано слово Еще..., нажмите F18 (Конец) для просмотра остальных записей буферного файла.
10. Найдите последний из файлов с именем QTMFFTRC, созданный для пользователя, работавшего с сервером FTP в момент запуска трассировки.
11. Нажмите F11 (Просмотр 2), для того чтобы узнать время и дату создания файла, с которым вы хотите работать.
12. Убедитесь, что вы работаете с буферным файлом QTMFFTRC, который был создан последним.

Если выполнить сбор информации трассировки не удалось, укажите это в отчете о неполадке. Включите в отчет о неполадке те данные трассировки, которые вы смогли получить.

Трассировка сервера FTP с помощью команды Трассировка приложения TCP/IP (TRCTCPAPP)

Команда Трассировка приложения TCP/IP (TRCTCPAPP), предусмотренная в выпусках V4R4 и выше, служит для *глобальной* трассировки *всех* серверов FTP системы.

Команда TRCTCPAPP предназначена для опытных специалистов по обслуживанию системы. Для запуска этой команды необходимы специальные права доступа *SERVICE. Команда TRCTCPAPP предназначена для сбора данных трассировки, необходимых для обслуживания системы и разработки программ. С ее помощью опытные пользователи могут динамически включать и выключать трассировку приложений.

Команда TRCTCPAPP позволяет собрать информацию трассировки для приложения FTP:

- Для сервера FTP системы iSeries можно собрать внутреннюю информацию трассировки. В качестве фильтра для этой информации можно указать удаленный IP-адрес и номер порта, либо имя пользовательского профайла iSeries. В каждый момент времени в системе может применяться только одно задание трассировки.

Рассмотрим два примера применения команды TRCTCPAPP:

Пример 1:

```
TRCTCPAPP APP(*FTP) SET(*ON)
```

Запуск трассировки всех серверов FTP. Для всех остальных приложений TCP/IP трассировка не включается.

Пример 2:

```
TRCTCPAPP APP(*FTP) SET(*CHK)
```

Проверка состояния трассировки заданий сервера FTP. Пусть предварительно была введена команда >

```
TRCTCPAPP APP(*FTP) SET(*ON) USER(JOEC00L)
```

Тогда результатом работы команды TRCTCPAPP APP(*FTP) SET(*CHK) будет набор сообщений следующего вида:

```
TCP45B7 TRCTCPAPP APP(*FTP) SET(*ON) USER(JOEC00L)
      MAXSTG(*DFT) TRCFULL(*WRAP)
TCP45B1 Включена трассировка для *FTP.
TCP45B2 Начат сбор данных для *FTP.
TCP45B3 Начат новый цикл записи в буфер данных для *FTP.
```

Трассировка клиента FTP

Для включения трассировки клиента FTP или просмотра команд, отправляемых на сервер FTP, вызовите команду клиента FTP DEBUG. Команда DEBUG включает и выключает режим отладки. При необходимости в ней можно задать уровень отладки. В режиме отладки все команды, передаваемые на сервер, выводятся на экран с символами '>>>'. Для включения трассировки клиента FTP уровень отладки должен быть равен 100.

```
DEBUg [уровень-отладки]
```

уровень-отладки

Если это значение равно 0, то режим отладки будет выключен. Если будет указано положительное целое число, то режим отладки будет включен.

Если значение не указано, то либо будет включен режим отладки 1 (если отладка была выключена), либо отладка будет выключена (если она была включена).

100

Включает режим трассировки клиента FTP. Трассировка будет выполняться до отключения режима отладки или завершения работы клиента FTP. (После выключения режима трассировки некоторое время потребуется на форматирование полученных данных.)

Примечание: Трассировку клиента FTP следует включать только для сбора информации, необходимой для составления отчета о неполадке. Трассировка может существенно снизить производительность системы.

В версии V4R4 добавлена новая функция отладки. Эта функция аналогична описанной выше команде DEBUG 100. При запуске клиента сначала проверяется наличие области данных QTMFTPDP100.

Создайте область данных QTMFTPDP100 в библиотеке QTEMP, введя следующую команду:

```
CRTDTAARA DTAARA(QTEMP/QTMFTPDP100) TYPE(*LGL) AUT(*USE)
```

Если область данных QTMFTPDP100 существует, то устанавливается уровень отладки, равный 100, и запускается трассировка клиента FTP. Это позволяет выполнять трассировку клиента в тех случаях, когда ее *нельзя* запустить с помощью команды DEBUG 100.

Работа с заданиями сервера FTP и протоколом задания

Протокол задания сервера FTP содержит дополнительную информацию об ошибках, возникших при работе этого сервера. Сервер FTP автоматически записывает протокол задания в буферный файл при сбое.

Протокол задания можно записать в файл, не завершая работу сервера, с помощью следующей команды клиента FTP:

```
QUOTE RCMD DSPJOBLOG
```

Эта команда создает копии сообщений о ранее возникших ошибках. Теперь можно просмотреть протокол задания, введя команду WRKSPLF.

Создавать копию протокола задания рекомендуется в тех случаях, когда ответные сообщения, получаемые клиентом от сервера, содержат только краткие сведения об ошибке, возникшей на сервере. Например, протокол может содержать более подробную информацию об ошибках ввода/вывода на сервере.

Если при попытке создать копию протокола задания сервера FTP описанным выше способом возникает ошибка, введите приведенную ниже команду для принудительного создания буферных файлов протоколов для всех сеансов FTP:

```
CHGJOB JOB(QUSRSYS/QTMFTPS) LOG(4 00 *SECLVL)
```

Затем попытайтесь воспроизвести неполадку. Для восстановления прежних параметров ведения протокола задания после получения необходимой информации введите:

```
CHGJOB JOB(QUSRSYS/QTMFTPS) LOG(4 00 *NOLOG)
```

Для записи протокола задания в буферный файл после каждого сеанса FTP и завершения работы сервера FTP (как нормального, так и с ошибками) введите следующую команду Изменить описание задания (CHGJOB):

```
CHGJOB JOB(QUSRSYS/QTMFTPS) LOG(4 00 *SECLVL)
```

Для создания протокола задания только при завершении работы сервера введите следующую команду CHGJOB:

```
CHGJOB JOB(QUSRSYS/QTMFTPS) LOG(4 00 *NOLIST)
```

Задания и имена заданий сервера FTP

Задания сервера FTP запускаются при выполнении команды STRTCP, если в параметре FTP AUTOSTART задано значение *YES, либо при выполнении команды STRTCPSVR со значением параметра SERVER, равным *FTP или *ALL. Эти задания выполняются в подсистеме QSYSWRK; они собирают информацию о подключающихся пользователях FTP. Имена этих заданий указываются в формате QTFTPnnnnn, где nnnnn - номер задания сервера FTP, переданного на сервер для выполнения.

Для выполнения операций с заданиями сервера FTP введите следующую команду CL:

```
WRKACTJOB JOB(QTFTP*)
```




Напечатано в Дании