

IBM

@server

iSeries

FTP





@server

iSeries

FTP

© Copyright International Business Machines Corporation 1998, 2002. All rights reserved.

© Copyright IBM Japan 2002

目次

FTP	1
V5R2 の新機能	2
トピックの印刷	3
FTP シナリオ	3
シナリオ: リモート・ホストからのファイルの転送	4
シナリオ: SSL を使用した FTP の保護	6
構成についての詳細	9
FTP サーバーの構成	10
iSeries ナビゲーターを使用した FTP サーバー	11
グラフィカル FTP クライアントおよび Web ツール用の FTP サーバーの構成	11
匿名 FTP の構成	12
匿名 FTP の準備	12
匿名 FTP 用の出口プログラムの作成	14
OS/400 ユーザー・プロファイル: ANONYMOUS の作成	14
公用ライブラリー / ディレクトリーの作成	15
出口プログラムの導入および登録	16
FTP の保護	16
FTP サーバーへのアクセスの防止	17
FTP のアクセス制御	18
SSL を使用した FTP サーバーの保護	19
ローカル認証局の作成	19
証明書の FTP サーバーへの関連付け	21
FTP サーバーのクライアント認証を必須にする (オプション)	21
FTP サーバーで SSL を使用可能にする	23
TLS/SSL を使用した FTP クライアントの保護	24
FTP 出口プログラムを使用したアクセス管理	24
iSeries ナビゲーターを使用したアクセス管理	25
着信 FTP ユーザーのモニター	25
FTP サーバーの管理	26
FTP サーバーの始動および停止	26
使用可能な FTP サーバー数の設定	26
構成可能なサブシステムのサポートによる FTP サーバーのパフォーマンスの改善	27
iSeries での FTP クライアントの使用	27
クライアント・セッションの開始と終了	32
サーバーのタイムアウトにおける考慮事項	32
FTP を使用したファイル転送	33
FTP のバッチ・ジョブ	37
FTP 参照情報	41
FTP サーバー・サブコマンド	44
ADDM (物理ファイル・メンバーの追加)	44
ADDV (物理ファイル可変長メンバーの追加)	44
APPE (既存ファイルへの追加)	45
AUTH (許可)	46
CRTL (ライブラリーの作成)	46
CRTP (物理ファイルの作成)	46
CRTS (ソース物理ファイルの作成)	46
CWD (作業ディレクトリーまたはライブラリーの変更)	47
DEBUG (FTP サーバー・トレースをオンに設定)	47

DELE (ファイルまたは文書の削除)	48
DLTF (ファイルの削除)	48
DLTL (ライブラリーの削除)	48
HELP (iSeries リモート・サーバーからのヘルプの取得)	48
LIST (ファイルのリスト)	49
MKD (ディレクトリーの作成)	49
MODE (転送モードの設定)	50
NLST (名前のリスト)	50
NOOP (サーバー応答の獲得)	50
PASS (パスワード)	51
PASV (受動データ接続の使用)	51
PBSZ (保護バッファ・サイズ)	51
PORT (データ・ポート)	52
PROT (データ・チャンネル保護レベル)	52
PWD (作業ディレクトリーまたはライブラリーの表示)	53
QUIT (FTP サーバー・セッションの終了)	53
RCMD (FTP サーバー・システムへの CL コマンドの送信)	53
REIN (システム間のセッションの再初期化)	54
RETR (ファイルの検索)	54
RMD (ディレクトリーの除去)	54
RNFR (変更前の名前)	55
RNTO (変更後の名前)	55
SITE (サーバー・システムが使用する情報の送信)	57
STOR (ファイルの保管)	58
STOU (固有の保管)	58
STRU (ファイル構造の指定)	58
SYST (オペレーティング・システム・システムの名前の識別)	59
TIME (FTP サーバーのタイムアウト値の設定)	59
TYPE (表示タイプの指定)	60
USER (サーバーへのユーザー・ログオン ID の送信)	62
FTP クライアント・サブコマンド	62
ACCT (アカウント情報の送信)	64
APPEND (リモート・ファイルへのローカル・ファイル・メンバーの追加)	64
ASCII (ファイル・タイプの ASCII への変更)	65
BINARY (転送タイプのイメージへの変更)	65
CD (作業ディレクトリーまたはライブラリーの変更)	66
CLOSE (リモート・システムとの FTP セッションの終了)	67
DEBUG (クライアント・トレースの作成およびリモート・システムへ送信されたサーバー・サブ コマンドの表示の制御)	67
DEBUG (クライアント・タイムアウト制限値の変更)	68
DELETE (リモート・システムのファイルの削除)	69
DIR (ディレクトリー項目、ライブラリー、またはファイルのリスト)	69
EBCDIC (ファイル・タイプの EBCDIC への変更)	70
GET (リモート・システムからローカル・システムへのファイルのコピー)	71
HELP (FTP サブコマンドのヘルプの取得)	72
LCD (ローカル・システムの作業ライブラリーまたは作業ディレクトリーの変更)	72
LOCSITE (ローカル・サイト情報の指定)	74
LOCSTAT (ローカル状況情報の表示)	74
LS (リモート・ファイル名のリスト)	75
LTYPE (ローカル・タイプ)	75
MDELETE (リモート・システムの複数ファイルの削除)	76
MGET (リモート・システムからローカル・システムへの複数ファイルのコピー)	77

MKDIR (ディレクトリーの作成)	77
MODE (データの伝送モードの指定)	78
MPUT (ローカル・システムからリモート・システムへの複数ファイル・メンバーの送信)	79
NAMEFMT (ファイル命名形式の選択)	80
NULLFLDS (NULL フィールドのあるファイルの転送の許可)	80
OPEN (リモート・システムの FTP サーバーへの接続)	81
PASS (パスワードの送信)	81
PUT (ローカル・システムのファイル・メンバーをリモート・システムのファイルへコピー)	82
PWD (現行ディレクトリー、現行フォルダー、または現行ライブラリーの表示)	82
QUOTE (FTP サーバーへのサブコマンドの送信)	83
REINITIALIZE (システム間のセッションの再初期化)	84
RENAME (リモート・システムのファイルの名前変更)	84
RESET (リセット)	84
RMDIR (ディレクトリーの除去)	85
SECData (データ・セキュリティ保護の設定)	85
SECOpen (データ・セキュリティ保護の設定)	87
SENDPASV (PASV サブコマンドを送信するかどうかの指定)	88
SENDPORT (PORT サブコマンドを送信するかどうかの指定)	89
SENDSITE (SITE サブコマンドを送信するかどうかの指定)	89
SITE (リモート・システムが使用する情報の送信)	90
STATUS (リモート・システムからの状況情報の検索)	90
STRUCT (ファイル構造の指定)	90
SUNIQUE (ファイルの上書きの制御)	91
SYSCMD (iSeries CL コマンドをローカルの iSeries に渡す)	92
TYPE (ファイル転送タイプの指定)	92
USER (リモート・システムへのユーザー ID の送信)	94
VERBOSE (エラー応答メッセージのテキスト表示の制御)	94
FTP 出口プログラム	95
要求妥当性検査出口点: クライアントおよびサーバー	95
例: CL で作成された FTP クライアントまたはサーバー要求妥当性検査用出口プログラム	97
例: ILE RPG コードで作成された FTP サーバー要求の妥当性検査出口プログラム	99
VLRQ0100 出口点形式	107
サーバー・ログオン出口点	108
例: CL コードで作成された FTP サーバー・ログオン出口プログラム	109
例: C コードで作成された FTP サーバー・ログオン出口プログラム	110
例: ILE RPG コードで作成された FTP サーバー・ログオン出口プログラム	119
TCPL0100 出口点形式	125
TCPL0200 出口点形式	131
TCPL0300 出口点形式	133
出口プログラムの削除	134
データの転送方法	135
iSeries サーバー間でのパック 10 進データを含むファイルの転送	135
*SAVF ファイルの転送	136
QDLS 文書の転送	137
“root”、QOpenSys、QLANSrv、QDLS、および QOPT ファイルの転送	137
QfileSvr.400 を使用したファイルの転送	138
QSYS.LIB ファイルの転送	140
QSYS.LIB でのテキスト・ファイルの受信	141
ファイルを事前に作成する場合の考慮事項	141
CCSID の変換	141
マッピング・テーブルの指定	142
iSeries ファイルの CCSID コード・ページのタグ付け	143

FTP における各国語サポートについての考慮事項	143
ファイル・システムと命名規則	145
FTP がサポートする OS/400 ファイル・システム	145
FTP サーバーの応答状況メッセージ	146
FTP サーバーの構文規則	147
FTP クライアントの構文規則	148
サブコマンド・パラメーターの囲み方	148
クライアント転送サブコマンドのファイル名	150
転送するファイルの命名	151
FTP のトラブルシューティング	151
FTP についての問題の判別	152
FTP の問題報告に必要な資料	155
FTP サーバーのトレース	155
FTP クライアントのトレース	159
FTP サーバー・ジョブおよびジョブ・ログでの作業	159

FTP

ファイル転送プロトコル (FTP) を利用すると、ネットワーク全体でファイルの送受信や共有を行うよう iSeries™ サーバーをセットアップすることができます。また、ファイルの名前変更、追加、および削除も可能になります。ファイル転送を行えるようにシステムをセットアップするには、その前にシステムで TCP/IP を構成して開始する必要があります。

iSeries サーバーでの FTP の使い方の詳細については、以下を参照してください。

▶ V5R2 の新機能

V5R2 iSeries の FTP の新機能を確認してください。

トピックの印刷

PDF 形式の FTP 資料をダウンロードまたは印刷してください。

FTP シナリオ

FTP の構成方法および使用法を理解するために、シナリオを読んでください。

FTP サーバーの構成

グラフィカル FTP クライアント、Web ブラウザー、および Web ツールを使用できるように iSeries FTP サーバーをセットアップします。また、匿名 FTP の構成を行います。

FTP の保護

FTP を SSL で保護し、FTP ユーザーをモニターし、FTP 機能へのユーザー・アクセスを管理することで、データを保護します。

FTP サーバーの管理

サーバー、FTP のセキュリティー、および SSL の使用の開始と停止など、FTP サーバーの管理を行います。

iSeries での FTP クライアントの使用

クライアント・セッションの開始と停止、ファイルの転送と受信、および FTP バッチ・ジョブのセットアップを行います。

FTP 参照情報

サーバー・サブコマンドとクライアント・サブコマンド、FTP 出口プログラム、データ転送方法などについての情報が記載されています。

FTP のトラブルシューティング

FTP サーバーあるいは FTP クライアントの問題のトラブルシューティングに関する情報が記載されています。

注: 重要な法的情報については、コードについての特記事項をお読みください。



V5R2 の新機能

iSeries FTP のバージョン 5 リリース 2 には、以下の拡張機能が備わっています。

拡張機能

- FTP サーバーで、独立補助記憶域プール (IASP) に常駐するライブラリーがサポートされるようになりました。
- トランスポート層セキュリティー (TLS)/Secure Sockets Layer (SSL) のサポート
V5R2 の FTP では、OS/400^R の FTP クライアントから TLS/SSL を使用できます (サーバー認証の場合のみ)。このサポートにより、認証に使用するパスワードなど、FTP 制御接続で転送するデータを暗号化するかどうかを指定できます。この新規機能をサポートするために、以下のような変更が行われました。

新規のクライアント・サブコマンド

- SECOpen (データ・セキュリティー保護の設定)
- SECData (データ・セキュリティー保護の設定)

STRTCPFTP の新規パラメーター

- セキュア接続 (SECCNN)
- データ保護 (DTAPROT)

LOCSITE サブコマンドの新規パラメーター

- DTAPROT C
- DTAPROT P
- DTAPROT



新規情報


V5R2 の Information Center の FTP についてのトピックが更新されました。トピック内でより簡単に情報にたどり着けるように、情報の編成が変更された箇所があります。V5R2 の FTP については、以下の情報が追加されました。

- 『FTP シナリオ』では、基本的な使用の概念を説明するために例が記載されています。iSeries 上で行う FTP サーバーの計画および構成に応じて、シナリオを参照してください。
- 『SSL を使用した FTP サーバーの保護』の説明を、SSL のトピックからこのトピックに移動しました。
- 匿名 FTP を構成するための出口プログラムの例を、技術情報サイトからこのトピックに移動しました。
 - 『サーバー・ログオン出口点』には、CL、C、および ILE RPG のコードの FTP サーバー・ログオン出口プログラムの例が記載されています。
 - 『要求妥当性検査出口点: クライアントおよびサーバー』には、FTP サーバーおよび FTP クライアントの要求妥当性検査用出口プログラムの例が CL および ILE RPG コードで記載されています。
- 以前は構成および解説書のトピックに含まれていた『FTP 参照情報』が FTP トピックに追加されました。

新規箇所または変更箇所を見つける方法

技術に関する変更箇所を見つけるには、以下の情報を使用します。

-  記号は、新規の情報または変更された情報が開始する箇所です。
-  記号は、新規の情報または変更された情報が終了する箇所です。

今回のリリースでの新規の情報や変更された情報については、「iSeries プログラム資料説明書」 を参照してください。

トピックの印刷


PDF 版をダウンロードし、表示するには、『FTP』（約 1450 KB、170 ページ）を選択します。

PDF ファイルの保存

表示用または印刷用の PDF ファイルを Netscape Navigator からワークステーションに保存するには、次のようにします。

1. ブラウザーで PDF を開く（上記のリンクをクリックする）。
2. 「名前を付けて保存」をクリックする。（IE の場合はフロッピーディスクのアイコン（名前を付けて保存）をクリックする。）
3. PDF を保存したいディレクトリーに進む。
4. 「保存」をクリックする。

Adobe Acrobat Reader のダウンロード

PDF ファイルを表示したり印刷したするには、Adobe Acrobat Reader が必要です。これは、Adobe Web サイト (www.adobe.com/products/acrobat/readstep.html)  から、ダウンロードできます。

FTP シナリオ

▶ 以下のシナリオは、FTP の動作方法およびご使用のネットワークでの FTP 環境の使用方法を理解するのに役立ちます。これらのシナリオには、初心者や熟練ユーザーが計画および構成タスクを準備する段階で役立つ、基本的な FTP の概念が記載されています。

リモート・ホストからのファイルの転送

テスト・サーバーにファイルを転送します。基本的な FTP を使用して、リモート・ホストにファイルを送信します。

SSL を使用した FTP の保護

パートナー企業に転送されるデータを、Secure Sockets Layer (SSL) で保護します。



シナリオ: リモート・ホストからのファイルの転送



目標

同僚がリモート・サーバーで何らかの Java™ 開発作業を行ったと想定します。システム・テストのエンジニアは、example.jar ファイルをリモート・サーバーからローカルのテスト・サーバーに転送する必要があります。基本的な FTP を使用して、ファイル（バイナリー・モード）を TCP/IP ネットワークに転送します。クライアントとサーバーは両方とも、OS/400 FTP を使用する iSeries です。

詳細

ファイルを転送するには、**制御接続** および**データ接続** という、2 つの接続を使用します。制御接続は、クライアントからサーバーにサブコマンドを送信し、サーバーからのこれらのサブコマンドに対する応答をクライアントで受信するために使用します。クライアントは、FTP サーバーに送信される FTP サブコマンドを開始します。データ接続は、実際のファイルを転送するために使用します。クライアントとサーバーの両方は、OS/400 ファイル・システムとインターフェースを取ります。ファイルを転送するには通常、両方のシステムでユーザー ID が必要です。その他の要件については、以下を参照してください。

- OS/400 を実行する iSeries サーバー
- TCP/IP Connectivity Utilities (5722-TC1)
- 構成済みの FTP サーバー
- リモート・システムのホスト名
- リモート・システムでのユーザー名とパスワード
- 転送するファイルの名前
- 転送するファイルの場所
- ファイル形式 (バイナリーや ASCII などのファイル転送の形式)

構成タスク

単純なファイル転送を実行するには、これらのタスクそれぞれを完了させる必要があります。

1. FTP クライアント・セッションを開始します。
このシナリオの場合: iSeries の文字ベースのインターフェースで、STRTCPFTP と入力し、**Enter** キーを押します。
2. ファイルの送信先のリモート・システムの名前を指定します。
このシナリオの場合: `theirco.com` を指定します。
3. リモート・システムに、リモート・サーバー用のユーザー名を通知します。
このシナリオの場合:

```
Enter login ID (yourid):  
====>yourid
```
4. リモート・システムに、リモート・サーバー用のパスワードを通知します。
このシナリオの場合:

```
Enter password:  
====>yourpassword
```
5. ファイルの転送元の TheirCo サーバー上のディレクトリーを見つけます。
このシナリオの場合: `====>cd /qibm/userdata/os400/dirserv/usrtools/windows`
6. ファイルの転送先のローカル・サーバー上のディレクトリーにナビゲートします。
このシナリオの場合: `====>lcd /qibm/userdata/os400/dirserv/usrtools/windows`
7. ファイル・タイプ、ASCII または BINARY を指定します。省略時のファイル・タイプは、ASCII です。
`.jar` ファイルの場合、ファイル転送タイプをバイナリーに切り替える必要があります。
このシナリオの場合: `====> binary`
8. リモート・サーバー・システムからクライアント・システムへのファイル転送を要求します。
このシナリオの場合: `====> get example.jar`
9. 終了したら、FTP を終了します。
このシナリオの場合: `====> QUIT`

次のステップ

さらに次のステップに進みます。バッチ FTP を使用して、自動的にファイルを転送することもできます。



シナリオ: SSL を使用した FTP の保護

▶ ベンチャー企業を調査し、その調査結果を投資計画業界の企業に販売する MyCo という会社の従業員だと想定します。そのような企業の 1 つである TheirCo は、MyCo が提供するサービスを必要とし、FTP 経由で調査報告書を受け取ることを望んでいます。MyCo は、どのような形式を使用する場合でも、配布するデータのプライバシーとセキュリティを常に保証してきました。この場合、MyCo は TheirCo との間に SSL 保護の FTP セッションを確立する必要があります。

目標

このシナリオでの目標は以下のとおりです。

- ローカル認証局 (CA) を MyCo の iSeries サーバーに作成して、運用する
- SSL を MyCo の FTP サーバーで使用できるようにする
- MyCo のローカル CA の証明書のコピーをファイルにエクスポートする
- *SYSTEM 証明書ストアを TheirCo のサーバーに作成する
- MyCo のローカル CA の証明書を TheirCo の *SYSTEM 証明書ストアにインポートする
- MyCo のローカル CA を TheirCo の FTP クライアントのトラステッド CA として指定する

前提条件

MyCo

- OS/400 の V5R1 以降を実行する iSeries が備わっている。
- V5R1 以降の TCP/IP Connectivity Utilities (5722-TC1) が iSeries サーバーに導入済みである。
- Cryptographic Access Provider 128-bit for iSeries サーバー (5722-AC3) が iSeries サーバーに導入済みである。
- IBM[®] デジタル証明書マネージャー (DCM) (5722-SS1 オプション 34) が iSeries サーバーに導入済みである。
- IBM HTTP Server (5722-DG1) が iSeries サーバーに導入済みである。
- 公用アプリケーションやリソースへのアクセスは、証明書を使用して保護している (詳細な説明は、『シナリオ: 証明書を使用して公用アプリケーションおよびリソースへのアクセスを保護する』を参照)。

TheirCo

- OS/400 の V5R2 以降を実行する iSeries サーバーが備わっている。
- V5R2 の TCP/IP Connectivity Utilities (5722-TC1) が iSeries サーバーに導入済みである。
- Cryptographic Access Provider 128-bit for iSeries サーバー (5722-AC3) が iSeries サーバーに導入済みである。
- IBM デジタル証明書マネージャー (5722-SS1 オプション 34) が iSeries サーバーに導入済みである。
- IBM HTTP Server (5722-DG1) が iSeries サーバーに導入済みである。
- FTP セッションに OS/400 TCP/IP FTP クライアントを使用する。

詳細

TheirCo は、OS/400 FTP クライアントを使用して、MyCo の FTP サーバーからのセキュアな FTP ファイル転送を要求します。『TLS/SSL を使用した FTP クライアントの保護』を参照してください。サーバー認証が行われます。

TheirCo は、SSL 保護の FTP セッションを使用して MyCo から財務報告書を受信します。

構成タスク

MyCo および TheirCo は、FTP セッションを SSL で保護するために、以下のタスクを完了させます。

MyCo のタスク

1. ローカル認証局を MyCo の iSeries サーバーに作成して、運用する
2. SSL を MyCo の FTP サーバーで使用できるようにする
3. MyCo のローカル CA の証明書のコピーをファイルにエクスポートする

TheirCo のタスク

1. *SYSTEM 証明書ストアを TheirCo のサーバーに作成する
2. MyCo のローカル CA の証明書を TheirCo の *SYSTEM 証明書ストアにインポートする
3. MyCo のローカル CA を TheirCo の FTP クライアントのトラステッド CA として指定する



構成についての詳細

SSL を使用して FTP を保護するために、以下のタスクのステップを完了させてください。

ステップ 1: ローカル認証局 (CA) を MyCo の iSeries サーバーに作成して、運用する

このシナリオでは、MyCo は iSeries サーバーの証明書のセットアップに、デジタル証明書マネージャー (DCM) を以前は使用していなかったことが想定されています。MyCo は、このシナリオの目標に基づいて、証明書を FTP サーバーに発行するためにローカル認証局 (CA) を作成し、運用することを選択しました。しかし、この代わりに、MyCo は FTP サーバーが SSL 用に公開証明書を使用するように構成することもできます。

デジタル証明書マネージャー (DCM) を使用してローカル CA を作成する場合、SSL を使用可能にするために必要なものを確実にすべて構成するためのプロセスのガイドが用意されています。

MyCo は、デジタル証明書マネージャー (DCM) を使用してサーバー上でローカル CA を作成し、運用するために、以下のステップを使用します。

1. DCM を開始します。
2. DCM のナビゲーション・フレームで、「**認証局 (CA) の作成 (Create a Certificate Authority (CA))**」を選択し、一連のフォームを表示します。これらのフォームは、ローカル CA の作成、および SSL 用のデジタル証明書の使用、オブジェクトの署名、および署名の検証の開始に必要な他のタスクを完了させるプロセス全体のガイドとなります。
3. 表示されるすべてのフォームを完了させます。iSeries サーバー上でローカル CA を作成し、運用するために実行する必要があるタスクそれぞれについて、フォームが用意されています。これらのタスクは、以下のとおりです。

- a. ローカル CA の証明書の秘密鍵を保管する方法を選択します。このステップは、IBM 4758-023 PCI Cryptographic Coprocessor が iSeries に導入されている場合にのみ行います。システムに暗号化コプロセッサが導入されていない場合は、DCM は証明書およびその秘密鍵を自動的にローカル CA の証明書ストアに保管します。
- b. 本人確認情報をローカル CA に提供します。
- c. ローカル CA 証明書をご使用の PC またはブラウザに導入します。これにより、ソフトウェアがローカル CA を認識し、CA が発行する証明書を検証できます。
- d. ローカル CA のポリシーのデータを選択します。
- e. 新規のローカル CA を使用して、アプリケーションが SSL 接続に使用できるサーバー証明書またはクライアント証明書を発行します。IBM 4758-023 PCI Cryptographic Coprocessor が iSeries サーバーに導入されている場合は、このステップを行うことでサーバー証明書やクライアント証明書の秘密鍵の保管方法を選択できます。システムにコプロセッサが導入されていない場合は、DCM は証明書およびその秘密鍵を自動的に *SYSTEM 証明書ストアに保管します。DCM は、このタスクの一部として *SYSTEM 証明書ストアを作成します。
- f. SSL 接続用のサーバー証明書またはクライアント証明書を使用するアプリケーションを選択します。注: OS/400 TCP/IP FTP サーバーのアプリケーション ID (QIBM_QTMF_FTP_SERVER) を必ず選択してください。
- g. 新規のローカル CA を使用して、アプリケーションがオブジェクトにデジタル署名するのに使用するオブジェクト署名の証明書を発行します。これにより、オブジェクト署名の証明書の管理に使用する *OBJECTSIGNING 証明書ストアが作成されます。注: このシナリオでは、オブジェクト署名の証明書を使用しませんが、必ずこのステップは完了させてください。タスクをこの時点で取り消すとタスクは終了し、SSL 証明書の構成を完了させるのに、別のタスクを実行する必要があります。
- h. ローカル CA を信頼するアプリケーションを選択します。注: OS/400 TCP/IP FTP サーバーのアプリケーション ID (QIBM_QTMF_FTP_SERVER) を必ず選択してください。

このガイド付きのタスクのフォームが完了すると、SSL を使用するように FTP サーバーを構成することができます。

ステップ 2: SSL を MyCo の FTP サーバーで使用できるようにする

この時点では、FTP サーバーには証明書が割り当てられているので、以下のステップに従って、MyCo は SSL を使用するように FTP サーバーを構成します。

1. iSeries ナビゲーターで、「**iSeries サーバー**」→「**ネットワーク**」→「**サーバー**」→「**TCP/IP**」と展開します。
2. 「**FTP**」を右クリックします。
3. 「**プロパティ**」を選択します。
4. 「**一般**」タブを選択します。
5. SSL サポートのオプション、「**セキュアのみ (Secure only)**」を選択します。
これを選択して、FTP サーバーと SSL セッションだけを行うことができるようにします。セキュアでない FTP ポートに接続されることもありますが、ユーザーがログインできるようになるにはまず、FTP クライアントが SSL セッションを折衝する必要があります。

このタスクを完了したので、MyCo の FTP サーバーは SSL を使用して通信セッションを暗号化し、これらのセッション中に送信されるデータのプライバシーを保護できるようになりました。しかし、FTP クライアントを FTP サーバーとの SSL セッションに参加するように構成するには、MyCo は顧客企業の TheirCo にローカル CA の証明書のコピーを与える必要があります。これを行うには、MyCo はファイル

にローカル CA の証明書のコピーをエクスポートし、TheirCo がこのファイルを使用できるようにする必要があります。TheirCo がこのファイルを取得したら、DCM を使用して、ローカル CA の証明書を *SYSTEM 証明書ストアにインポートし、SSL を使用するように OS/400 FTP クライアントを構成することができます。

ステップ 3: MyCo のローカル CA の証明書のコピーをファイルにエクスポートする

MyCo は、TheirCo にローカル CA の証明書のコピーを与える必要があります。TheirCo のクライアント・アプリケーションが SSL セッションに参加するにはまず、CA の証明書を信頼するようにこのクライアント・アプリケーションを構成する必要があります。

MyCo は、以下のステップを使用して、ファイルにローカル CA の証明書のコピーをエクスポートします。

1. DCM を開始します。
2. 「証明書ストアの選択 (Select a Certificate Store)」を選択します。
3. 「*SYSTEM」をオープンする証明書ストアとして選択し、「続行」をクリックします。
4. 「証明書ストアおよびパスワード (Certificate Store and Password)」ページが表示されたら、証明書ストアの作成時に指定したパスワードを入力して、「続行」をクリックします。
5. ナビゲーション・フレームが最新表示されたら、「証明書の管理 (Manage Certificates)」を選択し、「証明書のエクスポート (Export certificate)」タスクを選択します。
6. 「認証局 (CA) (Certificate Authority (CA))」を選択し、「続行」をクリックして CA の証明書のリストを表示します。
7. MyCo のローカル CA の証明書をリストから選択し、「エクスポート」をクリックします。
8. 「ファイル」をエクスポート先として指定し、「続行」をクリックします。
9. エクスポートされたローカル CA の証明書の完全修飾パスおよびファイル名を指定し、「続行」をクリックして、証明書をエクスポートします。
10. 「OK」をクリックして、「エクスポートの確認 (Export confirmation)」ページを終了します。

これで、証明書と共に作成する署名を検証する iSeries エンドポイント・システムに、これらのファイルを転送できるようになりました。これらのファイルはセキュアに送信する必要がないため、ファイルの転送には、電子メールまたは FTP を使用できます。

次に、TheirCo は DCM を使用して、ローカル CA の証明書を *SYSTEM 証明書ストアにインポートし、MyCo のローカル CA (およびローカル CA が発行する証明書) を信頼することを指定します。

ステップ 4: *SYSTEM 証明書ストアを TheirCo のサーバーに作成する

SSL セッションに参加するには、MyCo の FTP サーバーが SSL セッションを確立する際に提示する証明書を、TheirCo の OS/400 FTP クライアントが認識して受け入れる必要があります。サーバーの証明書を認証するには、TheirCo の FTP クライアントが認証局 (CA) の証明書のコピーを *SYSTEM 証明書ストアに保持している必要があります。*SYSTEM 証明書ストアには、ほとんどの一般的な公開 CA の証明書のコピーが保管されています。しかし、MyCo の FTP サーバーがローカル CA の証明書を使用する場合は、TheirCo の FTP クライアントは、ローカル CA の証明書のコピーを取得し、*SYSTEM 証明書ストアにインポートする必要があります。

このシナリオでは、以前にデジタル証明書マネージャー (DCM) を使用して証明書を作成または管理されていなかったことが想定されています。このため、TheirCo はまず、以下のステップに従って、*SYSTEM 証明書ストアを作成する必要があります。

1. DCM を開始します。
2. デジタル証明書マネージャー (DCM) のナビゲーション・フレームで、「**新規の証明書ストアの作成 (Create New Certificate Store)**」を選択し、「*SYSTEM」を、作成する証明書ストアとして選択し、「**続行**」をクリックします。
3. 「**いいえ**」を選択して、*SYSTEM 証明書ストアの作成の一部として証明書を作成し、「**続行**」をクリックします。
4. 新規の証明書ストアのパスワードを指定し、「**続行**」をクリックして、確認ページを表示します。
5. 「**OK**」をクリックします。

これで、TheirCo はローカル CA の証明書を証明書ストアにインポートし、その証明書ストアを証明書の信頼の置けるソースとして指定できるようになりました。

ステップ 5: MyCo のローカル CA の証明書を TheirCo の *SYSTEM 証明書ストアにインポートする

TheirCo はこれらのステップを使用して、ローカル CA の証明書を *SYSTEM 証明書ストアにインポートし、その証明書ストアを証明書の信頼の置けるソースとして指定します。

1. DCM のナビゲーション・フレームで、「**証明書ストアの選択 (Select a Certificate Store)**」をクリックし、「*SYSTEM」をオープンする証明書ストアとして選択します。
2. 「証明書ストアおよびパスワード (Certificate Store and Password)」ページが表示されたら、証明書ストアの作成時に指定したパスワードを入力して、「**続行**」をクリックします。
3. ナビゲーション・フレームが最新表示されたら、「**証明書の管理 (Manage Certificates)**」を選択してタスクのリストを表示します。
4. タスク・リストから、「**証明書のインポート (Import certificate)**」を選択します。
5. 「**認証局 (CA) (Certificate Authority (CA))**」を証明書タイプとして選択し、「**続行**」をクリックします。
6. CA の証明書の完全修飾パスおよびファイル名を指定し、「**続行**」をクリックします。インポートのプロセスが正常に実行されたことを確認するメッセージが表示されるか、またはプロセスが失敗した場合はエラー情報を提供するメッセージが表示されます。

これで、TheirCo の FTP クライアントが MyCo のローカル CA の証明書を使用するサーバー・アプリケーションとの SSL セッションに参加できるように、TheirCO の FTP クライアントが MyCo のローカル CA の証明書を信頼することを指定できるようになりました。

ステップ 6: MyCo のローカル CA を TheirCo の FTP クライアントのトラステッド CA として指定する

TheirCo が FTP クライアントを使用して MyCo の FTP サーバーとのセキュアな接続を確立するにはまず、TheirCo が DCM を使用してクライアントがどの CA を信頼するかを指定する必要があります。つまり、TheirCo は、直前にインポートされたローカル CA の証明書を信頼することを指定する必要があります。

TheirCo は、以下のステップを使用して、FTP クライアントが MyCo のローカル CA の証明書を信頼することを指定します。

1. DCM を開始します。

2. 「**証明書ストアの選択 (Select a Certificate Store)**」をクリックし、「*SYSTEM」をオープンする証明書ストアとして選択します。
3. 「**証明書ストアおよびパスワード (Certificate Store and Password)**」ページが表示されたら、証明書ストアの作成時に指定したパスワードを入力して、「**続行**」をクリックします。
4. ナビゲーション・フレームで、「**アプリケーションの管理 (Manage Applications)**」を選択して、タスクのリストを表示します。
5. タスク・リストから、「**信頼する CA のリストの定義 (Define CA trust list)**」を選択します。
6. 「**クライアント (Client)**」をリストを定義するアプリケーションのタイプとして選択し、「**続行**」をクリックします。
7. リストから OS/400 TCP/IP FTP クライアント・アプリケーション (QIBM_QTMF_FTP_CLIENT) を選択し、「**続行**」をクリックして、CA の証明書のリストを表示します。
8. 直前にインポートした MyCo のローカル CA の証明書を選択し、「**OK**」をクリックします。DCM は、信頼するリストの選択を確認するメッセージを表示します。

これらのステップが完了すると、MyCo の FTP サーバーは、TheirCo の FTP クライアントおよび FTP サーバーと SSL セッションを確立できます。『**TLS/SSL を使用した FTP クライアントの保護**』を参照してください。

FTP サーバーの構成

▶ TCP/IP Connectivity Utilities のライセンス・プログラムでは、出荷時に TCP/IP FTP サーバーが構成されています。TCP/IP を開始すると、FTP サーバーが同時に始動します。

インターネット上で FTP サーバーを構成するにはまず、データを保護するための以下の保護対策を検討する必要があります。

- iSeries サーバーとインターネット間でファイアウォールを使用する。
- FTP サーバーには実動に使用しない iSeries を使用する。
- FTP サーバーを、社内の他の LAN や WAN に接続しない。
- FTP サーバーに対するアクセスを保護するために、FTP 出口プログラムを利用する。
- FTP 出口プログラムにセキュリティーの抜け穴がないことを確認するテストを月に 1 回実行する。
- 匿名 FTP ユーザーには、同一ディレクトリーに対する読み取りアクセスと書き込みアクセスの両方を許可しない。これを許可すると、インターネット上で匿名ユーザーをトレースできなくなります。
- iSeries FTP サーバーへの全アクセスをログに記録し、攻撃されていないかどうか毎日または週に 1 回ログを調べる。
- FTP サーバーに正しい出口プログラムが登録されているか、月に 1 回検査する。
- 『FTP の保護』を参照して、iSeries FTP サーバーを保護する方法を検討する。

以下のトピックでは、FTP サーバーを表示し、カスタマイズする方法を説明しています。

iSeries ナビゲーターを使用した FTP サーバー

iSeries ナビゲーターを使用して、iSeries FTP サーバーを構成および管理します。

グラフィカル FTP クライアントおよび Web ツール用の FTP サーバーの構成

グラフィカル FTP クライアント、Web ブラウザー、および他の Web ツールをサポートするように、iSeries で FTP サーバーを構成します。

匿名 FTP の構成

匿名 FTP を使用すると、ユーザー ID とパスワードが割り当てられていないリモート・ユーザーでも、FTP サーバーを使用できます。



iSeries ナビゲーターを使用した FTP サーバー

iSeries ナビゲーターを使用して FTP サーバーの構成作業ができます。iSeries ナビゲーターの FTP 用のグラフィカル・ユーザー・インターフェースを使用するには、次の手順に従います。

1. iSeries ナビゲーターで、「ユーザーの iSeries サーバー」 → 「ネットワーク」 → 「サーバー」 → 「TCP/IP」と展開します。
2. 右側のペインで、「FTP」を右クリックし、「プロパティ」を選択します。
3. ここで、FTP サーバーのプロパティを変更することができます。ヘルプ・ボタンをクリックすればオンライン・ヘルプが表示されます。特定のフィールドのヘルプ情報を取得するには、疑問符 (?) をクリックして、次にそのフィールドをクリックします。

グラフィカル FTP クライアントおよび Web ツール用の FTP サーバーの構成

▶ iSeries FTP サーバーでは、FTP クライアント、Web ブラウザー、および Web 開発ツールがサポートされています。

ほとんどのグラフィカル FTP クライアントは、UNIX^R をリスト形式として使用し、パス・ファイルをファイル名形式として使用します。サポートされている形式を使用するには、次の手順に従って、FTP サーバーのプロパティを設定します。

1. iSeries ナビゲーターで、「ユーザーの iSeries サーバー」 → 「ネットワーク」 → 「サーバー」 → 「TCP/IP」と展開します。
2. 右側のペインで、「FTP」を右クリックし、「プロパティ」を選択します。
3. 「プロパティ」ページで、「初期形式」タブをクリックします。
 - パスをファイル命名形式として使用可能にする。
 - UNIX リスト形式をファイル・リスト形式として使用可能にする。

注: FTP サーバー・ログイン出口点の TCPL0200 形式または TCPL0300 形式の出口プログラムを使用して、特定の FTP セッションの LISTFMT 設定および NAMEFMT 設定を制御できます。

リスト形式は、進行中の FTP セッションを終了した後でも、FTP サーバーの SITE (サーバー・システムが使用する情報の送信) サブコマンド用オプションを使用して変更できます。これらの設定は、FTP サーバー・サブコマンド、LIST (ファイルのリスト) および NLST (名前のリスト) から戻される結果を制御します。

以下も参照してください。

- iSeries 形式の LIST 情報
- UNIX スタイル形式の LIST 情報



匿名 FTP の構成

匿名 FTP は、リモート・システム上の選択された情報への無保護アクセス (パスワード不要) を可能にします。リモート側では、一般のアクセスで利用される情報を決定します。このような情報は、共用にアクセスできるものとして扱われ、誰でも読み取り可能です。適切な情報のみを利用可能にするのは、その情報およびシステムの所有者の責任です。

ユーザーがこの情報をアクセスするには、ユーザー ID ANONYMOUS を使用してホストにログオンします。ユーザー ANONYMOUS の、そのサーバー上のファイルへのアクセス権および操作は制限されます。通常許可される操作は以下のとおりです。

- FTP を使用するログオン
- 限定されたディレクトリー内の内容のリスト表示
- それらのディレクトリー内のファイルの検索

通常匿名ユーザーは、ファイルを FTP サーバー・システムに転送できません。システムによっては、匿名ユーザーがデータを送信できる着信ディレクトリーを提供しているものがあります。慣習的に、特別な匿名ユーザー・アカウントはパスワードとしてストリングを受け入れます。ただしこのパスワードは、「guest」またはそのユーザーの e-mail アドレスを使用するのが一般的です。アーカイブ・サイトには、「guest」パスワードを許可せず、そのユーザーの e-mail アドレスを明示的に要求するものがあります。e-mail アドレスを指定することは、アーカイブ・サイトのオペレーターが、サービスを使用しているユーザーが誰であるのかを知る、手助けになります。

iSeries 上の匿名 FTP

基本的なファイル転送プロトコル (FTP) サーバーでは、匿名 FTP はサポートされません。ユーザーの iSeries サーバー上に匿名 FTP を設定するには、2 つの FTP サーバー出口点 (サーバー・ログオン出口点および要求の妥当性検査出口点) に、出口プログラムを提供する必要があります。

匿名 FTP は便利で要求度の高いサービスであるため、提供したいという要望が高いかもしれません。ただし、匿名 FTP を使用すると、iSeries サーバーのセキュリティについての懸念が生じます。システム保護についての詳細は、『FTP 出口プログラムを使用したアクセス管理』を参照してください。

匿名 FTP を構成するには、以下のタスクを完了させます。

1. **匿名 FTP の準備**
技術的な要件を検討し、セキュリティ・ポリシーを定義します。
2. **匿名 FTP 用の出口プログラムの作成**
匿名 FTP サポート用の 2 つの出口プログラムを作成します。このトピックでは、多くの出口プログラムと、役に立つヒントや技法を例示しています。
3. **OS/400 ユーザー・プロファイル: ANONYMOUS の作成**
ユーザー ID ANONYMOUS による iSeries サーバーへの直接 サインオンを防止します。
4. **公用ライブラリー / ディレクトリーの作成**
公衆ライブラリーまたはディレクトリーを作成、ロード、および設定します。
5. **出口プログラムの導入および登録**
出口プログラムとそのログ・ファイルを格納するライブラリーを作成し、プログラムをコンパイルし、そのプログラムを FTP サーバーが使用できるように登録します。

匿名 FTP の準備

匿名 FTP の構成を準備する際には、以下の情報を考慮してください。

スキルの要件

匿名 FTP をセットアップするには、以下のスキルが必要になります。

- iSeries の文字ベースのインターフェース、および複数のパラメーターとキーワードを使用するコマンドについての十分な理解。
- iSeries 上でライブラリー、メンバー、ソースとなる物理ファイルを作成できること (最低でも *SECOFR 権限が必要)。
- ライブラリー、ファイル、メンバー、およびプログラムに対して権限の割り当てができること。
- iSeries サーバーでプログラムを作成、変更、コンパイル、およびテストができること。

セキュリティについての考慮事項

匿名 FTP をインプリメントする際に最初のステップとなるのは、匿名 FTP サーバーのサイト・ポリシーを定義することです。以下のプランは、FTP サイトのセキュリティ定義を行い、出口プログラムのコーディング方法を決めるためのものです。FTP サーバーを使用すると世界中の誰でもデータにアクセスできるようになるため、FTP サーバーをどのように使用するか、どのデータを保護する必要があるかについて、慎重に考慮する必要があります。

FTP サイト・ポリシーのプランにおける以下の推奨事項を検討してください。

- iSeries サーバーとインターネット間でファイアウォールを使用する。
- FTP サーバーには実動に使用しない iSeries を使用する。
- FTP サーバーを、社内の他の LAN や WAN に接続しない。
- FTP サーバーに対するアクセスを保護するために、FTP 出口プログラムを利用する。
- FTP 出口プログラムにセキュリティの抜け穴がないことを確認するテストを実行する。
- 匿名 FTP ユーザーには、同一ディレクトリーに対する読み取りアクセスと書き込みアクセスの両方を許可しない。これを許可すると、インターネット上で匿名ユーザーをトレースできなくなります。
- ANONYMOUS アクセスのみを許可する。ANONYMOUS 以外のユーザー ID の使用を許可せず、パスワードの確認も行わない。
- ANONYMOUS アクセスの対象を 1 つの公用ライブラリーまたはディレクトリーのみに制限する。(その設定場所と名前を確認)
- 公用ライブラリーまたはディレクトリーには公用アクセス・ファイルのみを格納する。
- ANONYMOUS ユーザーの権限を「表示」と「検索」に関するサブコマンド (get、mget) のみに制限する。環境を問わず、**ANONYMOUS ユーザーの CL コマンドの使用は禁止します。**
- iSeries FTP サーバーへの全アクセスをログに記録する。
- FTP サーバーのログは侵入に備えて毎日または毎週チェックする。
- FTP サーバーが正しい出口プログラムを登録していることを、月に 1 回検査する。
- 月に 1 回、FTP サーバーにセキュリティの抜け穴がないか確認するテストを実行する。

次の操作:

匿名 FTP 用の出口プログラムの作成

匿名 FTP 用の出口プログラムの作成

▶▶ 匿名 FTP をサポートするには、FTP サーバー・ログオン出口プログラムと FTP サーバー要求妥当性検査用出口プログラムの 2 つの出口プログラムを作成する必要があります。FTP サーバー・ログオン出口プログラムは、ANONYMOUS ユーザー ID を使用可能にし、ANONYMOUS ユーザーを強制的に公用ライ

ブラリーまたはディレクトリーにアクセスさせます。FTP サーバー要求の妥当性検査出口プログラムは、ANONYMOUS ユーザーが使用できるコマンド、ファイル、ディレクトリー、またはライブラリーを制限します。

出口点および出口点形式

FTP サーバーは、特定の出口点を通じて各出口プログラムとの通信を行います。パラメーターは、サーバーおよび出口プログラム間で受け渡しされます。交換される情報の形式は、出口点形式で指定されます。FTP の出口プログラムの詳細は、『FTP 出口プログラム』を参照してください。匿名 FTP には、以下の出口点を使用します。

プログラム	出口点	形式
サーバー・ログオン	QIBM_QTMF_SVR_LOGON	TCPL0100、TCPL0200、または TCPL0300。 ¹
要求妥当性検査	QIBM_QTMF_SERVER_REQ	VLRQ0100

¹ 出口点は、複数の形式を持っている場合があります。ただし、一つの出口プログラムの登録は、1 つの出口点形式に対してのみ可能です。これらの形式をそれぞれ検査して、ユーザーのシステムに最も適切な形式を 1 つ選択します。

プログラム例

サーバー上に匿名 FTP をセットアップする際に役に立つプログラム例を利用できます。これらのサンプルは、プログラムを構築する際の開始点として使用できます。サンプルのコードの一部をコピーして、それを、独自に作成するプログラムに追加できます。サンプル・プログラムの実行は、実動システム以外のシステムで行うことをお勧めします。

注: 本書に記載したコード例は、実例をあげることを目的としています。これらのコードには、実動マシン上で現状のまま実行できるだけの機能は含まれていません。独自のプログラムを作成する際にこれらのコードを開始点として使用したり、コードの一部を使用することは、ご自由に行ってください。

プログラム例を表示するには、『サーバー・ログオン出口点』および『要求妥当性検査出口点: クライアントおよびサーバー』のサーバー・プログラムを参照してください。

次の操作:

OS/400 ユーザー・プロファイル: ANONYMOUS の作成



OS/400 ユーザー・プロファイル: ANONYMOUS の作成

ユーザー・プロファイル ANONYMOUS を使用してユーザーが直接 iSeries サーバーにサインオンすることがないように、ユーザー・プロファイル ANONYMOUS をパスワード *NONE にして作成することを推奨します。このプロファイルは、iSeries ナビゲーターを使用して作成できます。

1. **iSeries ナビゲーター**で、「**ユーザーおよびグループ**」を展開します。
2. 「**すべてのユーザー**」を右クリックし、「**新規ユーザー (New Users)**」を選択します。
3. 「**新規ユーザー**」パネルで、以下の情報を入力します。
「**ユーザー名**」 = ANONYMOUS
「**パスワード**」 = パスワードなし。
4. 「**ジョブ**」ボタンをクリックし、「**一般**」タブを選択します。

5. 「一般」タブで、匿名ユーザーが使用する必要がある「現行ライブラリー」および「ホーム・ディレクトリー」を割り当てます。
6. 「OK」をクリックして、他の設定を完了させます。
7. 「追加」をクリックして、プロファイルを作成します。

次の操作:

公用ライブラリー / ディレクトリーの作成

公用ライブラリー / ディレクトリーの作成

匿名ユーザーを作成した後は、匿名ユーザーが使用する公用ライブラリーまたはディレクトリーを作成してください。通常は、匿名ユーザーは公用ファイルにしかアクセスできないようにします。匿名ユーザーを単一のライブラリーまたは単一のディレクトリー・ツリーに限定することをお勧めします（これらは、「公用」ファイルのみを含むため）。

1. 匿名 FTP からアクセス可能なファイルを含んでいる公用ライブラリーまたはディレクトリーを作成します。
2. 公用ライブラリーまたはディレクトリーに公用アクセス・ファイルをロードします。
3. 公用ライブラリーまたはディレクトリーおよびファイル権限を PUBLIC *USE に設定します。

次の操作:

出口プログラムの導入および登録

出口プログラムの導入および登録

出口プログラムの導入

1. 出口プログラムと出口プログラムのログ・ファイルを格納するディレクトリーを作成します。
2. このライブラリーで作成した出口プログラムをコンパイルします。
3. PUBLIC *EXCLUDE 権をライブラリー、プログラム、ファイル・オブジェクトに対して認可します。
FTP サーバー・アプリケーションは、必要に応じて出口プログラムの決定と呼び出しを行う権限を採用します。

出口プログラムの登録

1. iSeries の文字ベースのインターフェースで、**WRKREGINF** と入力します。
2. ページを送って次の FTP サーバー・ログオン出口点を表示します。

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

3. 出口点の入力箇所の左側にある Opt フィールドに **8** を入力し、**Enter** を押します。
4. 「出口プログラムの処理」画面に、**1** (追加) を入力します。
5. 「出口プログラム」フィールドに出口プログラムの名前を入力します。
6. 出口プログラムを格納するライブラリーの名前を「ライブラリー」フィールドに入力します。
7. **Enter** キーを押します。
8. すべての FTP サーバー・インスタンスが出口プログラムを使用することを確定させるために、FTP サーバーをいったん終了させて再始動します。
9. 出口プログラムのテストを十分に行います。

注: 出口プログラムは、FTP サーバーが新しいセッションを要求すると、すぐに有効になります。すでに実行されているセッションは、影響を受けません。

関連トピック: 出口プログラムの削除

FTP の保護

インターネット上で iSeries システムを FTP サーバーとして使用する場合、FTP サーバーには全世界からアクセスできます。そのため、iSeries サーバーに保管されている重要なビジネス・データが危険にさらされないようにするためには、FTP のセキュリティに対する注意が必要です。FTP クライアントを保護するためのステップも用意されています。

以下のトピックに、FTP サーバーおよび FTP クライアントを保護する方法についての情報が記載されています。

FTP サーバーへのアクセスの防止

FTP 使用していないときには、FTP ポートから iSeries サーバーに侵入できないように FTP を稼働させないようにする必要があります。このトピックでは、FTP ポートをブロックする方法について説明します。

FTP のアクセス制御

FTP を使用している場合、データおよびネットワークを保護するために、ユーザーを常に制御する必要があります。このトピックには、ヒントとセキュリティについての考慮事項が記載されています。

SSL を使用した FTP の保護

SSL のサポートを使用することによって、ユーザーは、OS/400 FTP サーバーを (SSL をサポートする) FTP クライアントと共に使用する際に、ネットワーク上でパスワードとデータを「プレーン・テキスト」で送信する危険を回避できます。

FTP 出口プログラムを使用したアクセス管理

このトピックでは、FTP 出口点を使用して iSeries を保護する方法を説明しています。

iSeries ナビゲーターを使用したアクセス管理

iSeries ナビゲーターにある「アプリケーション管理によるアクセス制限 (Application Administration Limit Access)」を使用して、iSeries FTP サーバーや FTP クライアントを保護できます。

着信 FTP ユーザーのモニター

FTP サーバーにログインする人をモニターします。

FTP サーバーへのアクセスの防止

▶ FTP を使用した iSeries サーバーへのアクセスを防止する場合、FTP サーバーの稼働を防止する必要があります。FTP から iSeries にアクセスするのを防止するには、以下のステップに従います。

FTP サーバーが自動的に始動するのを防止する

TCP/IP を開始したときに FTP サーバー・ジョブが自動的に開始するのを防止するには、以下のステップに従います。

1. iSeries ナビゲーターで、「ユーザーの iSeries サーバー」 → 「ネットワーク」 → 「サーバー」 → 「TCP/IP」と展開します。

2. 「FTP」を右クリックし、「プロパティ」を選択します。
3. 「TCP/IP の開始時に開始 (Start when TCP/IP starts)」を選択解除します。


FTP ポートへのアクセスの防止

FTP が始動したり、FTP 用に iSeries が通常使用するポートに誰かがユーザー・アプリケーション (ソケット・アプリケーションなど) を関連付けるのを防止するには、以下を行います。

1. iSeries ナビゲーターで、「ユーザーの iSeries サーバー」→「ネットワーク」→「サーバー」→「TCP/IP」と展開します。
2. 「TCP/IP 構成」を右クリックし、「プロパティ」を選択します。
3. 「TCP/IP 構成プロパティ (TCP/IP Configuration Properties)」ウィンドウで、「ポート制限」タブをクリックします。
4. 「ポート制限」ページで、「追加」をクリックします。
5. 「ポート制限の追加」ページで、以下を指定します。
 - 「ユーザー名」: iSeries 上で保護するユーザー・プロファイル名を指定します。(保護するユーザー・プロファイルは、権限を認可するプログラムを所有せず、他のユーザーに知られているパスワードを持たないユーザー・プロファイルです。) ポートを特定ユーザーに制限することで、他のすべてのユーザーを自動的に除外することになります。
 - 「開始ポート」: 20
 - 「終了ポート」: 21
 - 「プロトコル」: TCP
6. 「OK」をクリックして、制限を追加します。
7. 「ポート制限」ページで、「追加」をクリックして、UDP プロトコルについてもこの手順を繰り返します。
8. 「OK」をクリックして、ポート制限を保管し、「TCP/IP 構成プロパティ (TCP/IP Configuration Properties)」ウィンドウをクローズします。
9. 次回に TCP/IP を開始するときに、ポート制限が有効になります。ポート制限を設定したときに TCP/IP が活動状態である場合には、TCP/IP を終了して、再度開始する必要があります。

注:

- 次回に TCP/IP を開始するときに、ポート制限が有効になります。ポート制限を設定したときに TCP/IP が活動状態である場合には、TCP/IP を終了して、再度開始する必要があります。

- Internet Assigned Numbers Authority (IANA) の Web サイト <http://www.iana.org>  に、割り当てられているポート番号についての情報が記載されています。
- ポート 20 または 21 が QTCP 以外のユーザー・プロファイルに制限されている場合に FTP サーバーを始動しようとする、エラーが発生して即座に終了します。
- この方法は、FTP サーバーなどのアプリケーションを完全に制限する場合にのみ、正しく機能します。特定ユーザーを制限する場合には、正しく機能しません。ユーザーが FTP サーバーに接続するときには、その接続要求はまず、QTCP プロファイルを使用します。正常に接続された後に、システムは個々のユーザー・プロファイルに変更します。FTP サーバーのすべてのユーザーは、ポートに対して QTCP の権限を使用します。



FTP のアクセス制御

➤ FTP クライアントがシステムにアクセスできるようにする場合は、以下のセキュリティー上の問題に注意してください。


- FTP をシステムで使用できるようにする場合、ご使用のオブジェクト権限体系では十分に行き届いた保護が行うことができない可能性があります。たとえば、ユーザーにファイルを表示する権限 (*USE 権限) がある場合、ユーザーはそのファイルを PC または他のシステムにコピーすることもできます。一部のファイルについては、他のシステムへのコピーを防止する必要があるかもしれません。
- FTP 出口プログラムを使用して、ユーザーが実行できる FTP の操作を制限することができます。FTP 要求妥当性検査出口を使用して、許可する操作を制御できます。たとえば、特定のデータベースのファイルに対して、GET 要求を拒否できます。
- サーバー・ログオン出口点を使用して、FTP サーバーにログオンするユーザーを認証することができます。『匿名 FTP の構成』では、出口プログラムを使用してシステムで匿名 FTP をセットアップする方法を説明しています。
- TLS/SSL を使用しない場合、FTP パスワードはクライアント・システムとサーバー・システム間で送信されるときに暗号化されません。接続方式によっては、システムが回線の盗聴によるパスワードの盗用に対して脆弱である可能性があります。
- QMAXSGNACN システム値が 1 に設定されている場合、QMAXSIGN システム値は TELNET に適用されますが、FTP には適用されません。QMAXSGNACN が 2 または 3 (サインオン回数が最大値に達した場合にプロファイルを使用できないようにする値) に設定されている場合は、FTP に対しログオンを試みた数も回数に入ります。この場合、ハッカーはユーザー・プロファイルが使用できないようになるまで、誤ったパスワードでログオンを繰り返し試みることによって、FTP 経由でサービス妨害攻撃を行うことができます。
- ログオンが失敗するごとに、システムはメッセージ CPF2234 を QHST ログに書き込みます。QHST ログをモニターして、このメッセージがあるか調べるプログラムを作成することができます。ログオンの試みが繰り返し行われていることをプログラムが検出した場合、プログラムは FTP サーバーを終了することができます。
- FTP 構成の非活動タイムアウト (INACTTIMO) パラメーターを使用して、ユーザーが FTP セッションを放置しているときにセッションが危険にされされる可能性を減らすことができます。資料やオンライン・ヘルプを読んで、INACTTIMO パラメーターと接続タイマー (サーバーの始動用) が連携する方法を必ず理解してください。

注: QINACTIV システム値は、FTP セッションに影響を与えません。

- FTP バッチのサポートを使用する場合、プログラムはユーザー ID とパスワードの両方をサーバー・システムに送信する必要があります。ユーザー ID とパスワードの両方をプログラムでコード化する必要があります。コード化しないと、プログラムはこれらをファイルから検索します。パスワードとユーザー ID を保管するこれら両方のオプションを使用すると、セキュリティを危険にさらす可能性があります。FTP バッチを使用する場合、オブジェクト・セキュリティを使用して、確実にユーザー ID とパスワード情報を保護する必要があります。また、ターゲット・システム上での権限が制限される単一のユーザー ID を使用する必要があります。そのようなユーザー ID には、ファイル転送など、必要な機能を実行するのに十分な権限だけを与える必要があります。
- FTP では、拡張プログラム間通信機能 (APPC) や iSeries Access と同様に、リモート・コマンド機能が提供されます。RCMD (リモート・コマンド) FTP サーバー・サブコマンドは、システムのコマンド行と同等です。FTP を使用可能にする前に、オブジェクト・セキュリティ機構が適切であることを確認してください。FTP 出口プログラムを使用して、RCMD サブコマンドを使用する試みを制限または拒否することができます。FTP 出口プログラムには、この出口点の説明とサンプル・プログラムが記載されています。
- ユーザーは、統合ファイル・システムのオブジェクトに FTP 経由でアクセスできます。この結果、システムで FTP サーバーを稼働するときには、統合ファイル・システムの権限体系が適切であることを確認する必要があります。

- ハッカーがよく行う活動の 1 つに、攻撃されるという疑いを持たないサイトを情報のリポジトリとしてセットアップすることがあります。違法な情報やポルノが掲載されることもあります。ハッカーが FTP 経由でユーザーのサイトへアクセスすると、この不適切な情報をユーザーの iSeries にアップロードします。その後、このハッカーは他のハッカーにユーザーの FTP アドレスを通知します。今度は、これらのハッカーが FTP からユーザーの iSeries にアクセスし、この不適切な情報をダウンロードします。

FTP 出口プログラムを使用して、このタイプの攻撃の防止を支援することができます。たとえば、情報のアップロード要求はすべて、書き込み専用のディレクトリーに送信します。このため、ハッカーの友人たちはそのディレクトリーの情報をダウンロードできないため、ハッカーは目標を達成できません。

「AS/400^R Internet Security: Protecting Your AS/400 from HARM on the Internet」 には、FTP 経由でのアップロードを許可した場合のリスクおよび解決策が記載されています。



SSL を使用した FTP サーバーの保護

▶ FTP サーバーを使用することで、非トラステッド・ネットワークにおけるファイルの送受信中のセキュリティが強化されます。FTP サーバーでは、Secure Sockets Layer (SSL) を使用して、情報の交換時にパスワードやその他の機密データを保護します。FTP サーバーでは、クライアント認証や自動サインオンなど、SSL または TLS で保護されたセッションがサポートされます (TLS と SSL プロトコルの詳細は、『SSL の概念』を参照)。

SSL が使用できるアプリケーションのほとんどは、1 つのポートは「保護されない」セッション、もう 1 つのポートはセキュアなセッションといったように、クライアントを個別の TCP ポートに接続します。しかし、FTP を保護する方が多少柔軟性があります。クライアントは、暗号化されていない TCP ポート (通常は TCP ポート 21) に接続し、認証オプションおよび暗号化オプションを折衝することができます。クライアントは、SSL 使用の接続が想定されるセキュアな FTP ポート (通常は TCP ポート 990) も選択できます。iSeries FTP サーバーでは、この両方のオプションが提供されます。

SSL を使用するように FTP サーバーを構成するにはまず、前提プログラムを導入し、iSeries サーバーにデジタル証明書をセットアップする必要があります。

FTP を保護するように SSL を構成するには、以下のタスクを完了させます。

1. ローカル認証局を作成するか、または DCM を使用して SSL 用に公開証明書を使用するように FTP サーバーを構成する。
2. 証明書を FTP サーバーに関連付ける
3. FTP サーバーのクライアント認証を必須にする (オプション)
4. FTP サーバーで SSL を使用可能にする

関連情報については、『TLS/SSL を使用した FTP クライアントの保護』を参照してください。◀

ローカル認証局の作成

▶ IBM デジタル証明書マネージャー (DCM) を使用して、ローカル認証局 (CA) を iSeries サーバー上に作成し、運用することができます。ローカル CA を使用することで、iSeries サーバーで実行するアプリケーション専用の証明書を発行することができます。

DCM を使用してローカル CA を iSeries サーバー上に作成し運用するには、以下のステップに従います。

1. DCM を開始します。

2. DCM のナビゲーション・フレームで、「**認証局 (CA) の作成 (Create a Certificate Authority (CA))**」を選択し、一連のフォームを表示します。これらのフォームは、ローカル CA を作成し、SSL 用のデジタル証明書の使用、オブジェクトの署名、および署名の検証の開始に必要なその他のタスクを完了させるプロセス全体のガイドとなります。
3. 表示されるすべてのフォームを完了させます。iSeries サーバー上でローカル CA を作成し、運用するために実行する必要があるタスクそれぞれについて、フォームが用意されています。これらのフォームを完了させると以下のことを行うことができます。
 - a. ローカル CA の証明書の秘密鍵を保管する方法を選択します。このステップは、IBM 4758-023 PCI Cryptographic Coprocessor が iSeries に導入されている場合にのみ行います。システムに暗号化コプロセッサが導入されていない場合は、DCM は証明書およびその秘密鍵を自動的にローカル CA の証明書ストアに保管します。
 - b. 本人確認情報をローカル CA に提供します。
 - c. ローカル CA 証明書をご使用の PC またはブラウザーに導入します。これにより、ソフトウェアがローカル CA を認識し、CA が発行する証明書を検証できます。
 - d. ローカル CA のポリシーのデータを選択します。
 - e. 新規のローカル CA を使用して、アプリケーションが SSL 接続に使用できるサーバー証明書またはクライアント証明書を発行します。IBM 4758-023 PCI Cryptographic Coprocessor が iSeries サーバーに導入されている場合は、このステップを行うことでサーバー証明書やクライアント証明書の秘密鍵の保管方法を選択できます。システムにコプロセッサが導入されていない場合は、DCM は証明書およびその秘密鍵を自動的に *SYSTEM 証明書ストアに保管します。DCM は、このタスクの一部として *SYSTEM 証明書ストアを作成します。
 - f. SSL 接続用のサーバー証明書またはクライアント証明書を使用するアプリケーションを選択します。注: OS/400 FTP サーバーのアプリケーション ID (QIBM_QTMF_FTP_SERVER) を必ず選択してください。
 - g. 新規のローカル CA を使用して、アプリケーションがオブジェクトにデジタル署名するのに使用するオブジェクト署名の証明書を発行します。これにより、オブジェクト署名の証明書の管理に使用する *OBJECTSIGNING 証明書ストアが作成されます。注: このシナリオでは、オブジェクト署名の証明書を使用しませんが、必ずこのステップは完了させてください。タスクをこの時点で取り消すとタスクは終了し、SSL 証明書の構成を完了させるのに、別のタスクを実行する必要があります。
 - h. ローカル CA を信頼するアプリケーションを選択します。注: OS/400 FTP サーバーのアプリケーション ID (QIBM_QTMF_FTP_SERVER) を必ず選択してください。

証明書の詳細について、以下の関連ページを参照してください。

ユーザー証明書の管理

ユーザーが DCM を使用して証明書を取得したり、既存の証明書をユーザーの iSeries ユーザー・プロファイルと関連付ける方法を学びます。

API を使用して証明書を非 iSeries ユーザーへプログラマチックに発行する

ローカル CA を使用して、証明書を iSeries ユーザー・プロファイルと関連付けずに専用の証明書をユーザーに発行する方法を学びます。

専用 CA 証明書のコピーの取得

専用 CA が発行するサーバー証明書を認証するために、専用 CA の証明書のコピーを取得して PC に導入する方法を学びます。

次の操作:



証明書の FTP サーバーへの関連付け

▶ ローカル認証局 (CA) の作成中に、証明書を FTP サーバー・アプリケーションに割り当てるタスクを実行しなかった場合、または公衆の CA から証明書を要求するようにシステムを構成した場合、このタスクを実行します。

1. IBM デジタル証明書マネージャーを開始します。証明書を取得または作成する必要がある場合、あるいは証明書システムをセットアップまたは変更する必要がある場合は、この時点で行います。証明書システムのセットアップ方法については、『DCM の構成』を参照してください。
2. 「証明書ストアの選択 (Select a Certificate Store)」ボタンをクリックします。
3. 「*SYSTEM」を選択します。「続行」をクリックします。
4. *SYSTEM 証明書ストア用の適切なパスワードを入力します。「続行」をクリックします。
5. 左のナビゲーション・メニューが再ロードされたら、「アプリケーションの管理 (Manage Applications)」を展開します。
6. 「証明書割り当ての更新 (Update certificate assignment)」をクリックします。
7. 次の画面で、「サーバー」アプリケーションを選択します。「続行」をクリックします。
8. 「OS/400 TCP/IP FTP サーバー (OS/400 TCP/IP FTP Server)」を選択します。
9. 「証明書割り当ての更新 (Update Certificate Assignment)」をクリックして、証明書を OS/400 TCP/IP FTP サーバーに割り当てます。
10. リストからサーバーに割り当てる証明書を選択します。
11. 「新規証明書の割り当て (Assign New Certificate)」をクリックします。
12. DCM は「証明書割り当ての更新 (Update Certificate Assignment)」ページを確認メッセージと共に再ロードします。FTP サーバーの証明書のセットアップが終了したら、「終了 (Done)」をクリックします。

次の操作:

FTP サーバーのクライアント認証を必須にする (オプション)
または
FTP サーバーで SSL を使用可能にする



FTP サーバーのクライアント認証を必須にする (オプション)

▶ FTP サーバーを使用してクライアントを認証する必要がある場合、IBM デジタル証明書マネージャーでアプリケーションの指定を変更できます。

注:FTP サーバーではクライアント認証がサポートされますが、OS/400 FTP クライアントではクライアント認証がサポートされません。一部のユーザーはそれでもクライアント認証を必要とすることがありますが、この場合、OS/400 FTP クライアントを SSL 接続に使用できません。

FTP クライアントが接続し、サーバーのクライアント認証が使用可能になる場合でも、クライアントは依然として USER サブコマンドを送信する必要があります。USER サブコマンド情報が送信されると、FTP サーバーは、SSL ハンドシェイクの一部としてクライアントがサーバーに送信したクライアント証明書に関連するプロファイルとユーザーが一致することを確認します。ユーザーがクライアント証明書と一致する

場合、パスワードは必要なくなり、FTP サーバーはユーザーをシステムにログオンします。USER サブコマンドが必要なのは、コマンドを実行せずにクライアントがログオンしたことをクライアントに「通知する」メカニズムが、FTP プロトコルにないためです。

1. IBM デジタル証明書マネージャーを開始します。証明書を取得または作成する必要がある場合、あるいは証明書システムをセットアップまたは変更する必要がある場合は、この時点で行います。証明書システムのセットアップ方法については、『DCM の構成』を参照してください。
2. 「証明書ストアの選択 (Select a Certificate Store)」ボタンをクリックします。
3. 「*SYSTEM」を選択します。「続行」をクリックします。
4. *SYSTEM 証明書ストア用の適切なパスワードを入力します。「続行」をクリックします。
5. 左のナビゲーション・メニューが再ロードされたら、「アプリケーションの管理 (Manage Applications)」を展開します。
6. 「アプリケーション定義の更新 (Update application definition)」をクリックします。
7. 次の画面で、「サーバー」アプリケーションを選択します。「続行」をクリックします。
8. 「OS/400 TCP/IP FTP サーバー (OS/400 TCP/IP FTP Server)」を選択します。
9. 「アプリケーション定義の更新 (Update Application Definition)」をクリックします。
10. 表示される表で「はい」を選択し、クライアント認証を必須にします。
11. 「適用」をクリックします。
12. DCM は「アプリケーション定義の更新 (Update Application Definition)」ページを確認メッセージと共に再ロードします。FTP サーバーのアプリケーション定義の更新が終了したら、「終了 (Done)」をクリックします。

次の操作:

FTP サーバーで SSL を使用可能にする



FTP サーバーで SSL を使用可能にする

▶ 以下のステップを実行して、FTP サーバーで SSL を使用可能にします。

1. iSeries ナビゲーターで、「ユーザーの iSeries サーバー」 → 「ネットワーク」 → 「サーバー」 → 「TCP/IP」と展開します。
2. 「FTP」を右クリックします。
3. 「プロパティ」を選択します。
4. 「一般」タブを選択します。
5. 以下の SSL サポートのオプションを 1 つ選択します。
 - 「セキュアのみ (Secure only)」
これを選択することで、FTP サーバーでは SSL セッションのみが行われます。セキュアでない FTP ポートに接続されることもありますが、ユーザーがログインできるようになるにはまず、FTP クライアントが SSL セッションを折衝する必要があります。
 - 「非セキュアのみ (Non-secure only)」
これを選択することで、FTP サーバーとのセキュアなセッションが禁止されます。SSL ポートと接続しようとしても、接続できません。

- 「セキュアと非セキュアの両方 (Both secure and non-secure)」

これを選択することで、FTP サーバーとはセキュアなセッションおよびセキュアでないセッションの両方を行うことができます。

注: FTP サーバーを再始動する必要はありません。FTP サーバーは、証明書がサーバー自身に割り当てられたことを動的に検出します。この変更を動的に検出しない場合は、iSeries サーバーに最新の PTF が適用されていることを確認してください。



TLS/SSL を使用した FTP クライアントの保護

▶ トランスポート層セキュリティ (TLS) /Secure Sockets Layer (SSL) 接続を使用して、FTP の制御接続とデータ接続を使用して転送されるデータを暗号化することができます。制御接続において暗号化を使用する主な理由は、FTP サーバーにログオンするときにパスワードを隠すためです。

FTP クライアントを使用してサーバーへの接続を保護するにはまず、DCM を使用して、FTP クライアント向けにトラステッド認証局を構成する必要があります。接続するサーバーに割り当てる証明書を作成するために使用した認証局を追加する必要があります。使用する CA によって、認証局 (CA) の証明書のエクスポートやインポートが必要な場合があります。トラステッド CA の詳細は、DCM トピックの『アプリケーションの CA トラスト・リストの定義』を参照してください。

制御接続に TLS/SSL 暗号方式を選択した場合、FTP クライアントは省略時で FTP データ接続で送信されるデータも暗号化します。FTP プロトコルでは、セキュアな制御接続を行わないとセキュアなデータ接続を確立できません。

暗号化はパフォーマンス・コストが非常に高いため、データ接続では回避することができます。回避することで、パフォーマンスを悪化させずに機密ではないファイルを転送でき、パスワードが明らかにされないの、依然としてシステムのセキュリティが保護されます。

FTP クライアントには STRTCPFTP CL コマンドおよびサブコマンドのパラメーターがあり、TLS/SSL サポート (SECOpen と SECData) の一部として使用されます。

iSeries FTP クライアントへの TLS/SSL 保護の指定

制御接続

TLS/SSL 保護は、STRTCPFTP コマンドおよび SECOPEN サブコマンドで指定できます。

STRTCPFTP (FTP) コマンドの場合、SECCNN セキュア接続パラメーターに *SSL を指定して、セキュアな制御接続を要求します。また、*IMPLICIT を指定して、事前定義されたサーバーのポート番号でセキュアな接続を取得することもできます。(詳細は、以下の IMPLICIT SSL 接続を参照。)

FTP クライアントのセッションでは、SECOPEN サブコマンドを使用して、セキュアな制御接続を取得できます。

データ接続

STRTCPFTP (FTP) コマンドの場合、セキュアなデータ接続を指定するには、DTAPROT データ保護パラメーターに *PRIVATE を入力します。データを暗号化しないで送信することを指定するには、DTAPROT データ保護パラメーターに *CLEAR を入力します。

セキュアな制御接続を確立したら、SECData サブコマンドを使用してデータ接続の保護レベルを変更できます。

暗黙 SSL 接続

一部の FTP サーバーでは、暗黙 SSL 接続と呼ばれるものがサポートされます。この接続では、*SSL オプションと同じ暗号化保護が提供されますが、この接続は、事前定義されたサーバー・ポート (通常は 990) でのみ可能です。ここでは、SSL/TLS 接続の折衝が行われることを考慮してサーバーを構成する必要があります。

この方法は、TLS/SSL 保護を行う標準プロトコルをサポートしない可能性がある FTP インプリメンテーションへの接続を保護するために使用されます。

SSL サポートの初期のインプリメンテーションの多くで暗黙の接続方法が使用されましたが、現在は推奨されておらず、IETF からも反対されています。

注: TLS/SSL 接続をセットアップするための標準プロトコルでは、サーバーへの接続を確立するときに、AUTH (許可) サーバー・サブコマンドの使用を必要とします。また、サーバー・サブコマンドの PBSZ と PROT は、データ保護レベルを指定するときに使用されます。

しかし、暗黙 SSL 接続の場合、AUTH、PBSZ、および PROT サーバー・サブコマンドは、サーバーに送信されません。代わりに、サーバーは、クライアントが以下のパラメーターを指定してこれらサブコマンドを送信したかのように動作します。

- AUTH SSL
- PBSZ 0
- PROT P



FTP 出口プログラムを使用したアクセス管理

FTP は、OS/400 のオブジェクト・セキュリティに基づいたセキュリティ・レベルを提供します。つまり、有効なユーザー・プロファイルとパスワードを持っていないリモート・ユーザーは、iSeries FTP サーバーにログオンできません。

FTP サーバー出口点、および FTP クライアント出口点に、FTP 出口プログラムを追加することによって、セキュリティを追加して、ユーザーのシステムに対する FTP アクセスをさらに制限できます。これを利用することにより、たとえば、ライブラリー、オブジェクト、コマンド使用などへのアクセス権に対するのと同様に FTP ログオンを制限できます。

FTP サーバー要求の妥当性確認出口プログラムを作成して、ユーザーがアクセスする可能性がある CL コマンドおよび FTP サブコマンドを制限することができます。説明と例については、『要求妥当性検査出口点: クライアントおよびサーバー』のトピックを参照してください。

サーバー・ログオン出口点の出口プログラムを利用すれば、TCP/IP アプリケーション・サーバーへのユーザー認証を制御できます

ユーザーは、クライアント出口点: 要求妥当性検査用の、FTP クライアント要求妥当性検査用出口プログラムを作成できます。このプログラムは、ユーザーが実行することのできる FTP クライアント機能を制御します。

ユーザーの状況によっては、FTP サブコマンドへのアクセス制限を考慮することもできます。これは、FTP サーバー要求妥当性検査出口点および FTP クライアント要求妥当性検査出口点の出口プログラムを作成する代わりに、アプリケーション管理によるアクセス制限を使用することで行います。

出口プログラムを正常に動作させるには、出口プログラムの導入および登録を行う必要があります。不要になったユーザーのプログラムは、出口プログラムの削除を正しく実行して、今後そのプログラムが機能しないようにする必要があります。

iSeries ナビゲーターを使用したアクセス管理

iSeries ナビゲーターを使用して FTP サーバーおよびクライアントの機能へのユーザー・アクセスを制限できるようにしました。個々のユーザーあるいはユーザーのグループに対して、機能へのアクセスを認可、および拒否する場合は、アプリケーション管理を使用します。あるいは、FTP 要求妥当性検査出口点用の FTP 出口プログラムを作成すると、FTP 機能へのアクセスを管理できます。

iSeries ナビゲーターを介して機能へのユーザー・アクセスを管理するには、次のステップを完了します。

1. iSeries ナビゲーターで、「ユーザーの iSeries サーバー」を右クリックし、「アプリケーション管理」を選択します。
2. 「ホスト・アプリケーション」タブを選択します。
3. 「iSeries の TCP/IP ユーティリティ (TCP/IP Utilities for iSeries)」を展開します。
4. 「ファイル転送プロトコル (FTP) (File Transfer Protocol (FTP))」を展開します。
5. 「FTP クライアント (FTP Client)」または「FTP サーバー (FTP Server)」を展開します。
6. アクセスを許可する機能あるいはアクセスを拒否する機能を選択します。
7. 「カスタマイズ」をクリックします。
8. 「使用のカスタマイズ (Customize Usage)」ダイアログを使用して、選択した機能へのアクセスを許可する、あるいは拒否するユーザーとグループのリストを変更します。
9. 「OK」をクリックして、「アクセスのカスタマイズ」ページの変更を保管します。
10. 「OK」をクリックして、「アプリケーション管理」ページを終了します。

あるいは、特定のユーザーまたはグループに許可する、登録済みの FTP 機能へのアクセスを、iSeries ナビゲーターのユーザーおよびグループ管理ツールを使って管理することができます。それには、次のステップに従います。

1. iSeries ナビゲーターで、「ユーザーの iSeries サーバー」→「ユーザーおよびグループ」と展開します。
2. 「すべてのユーザー」または「グループ」を選択します。
3. ユーザーまたはグループを右クリックし、「プロパティ」を選択します。
4. 「機能」をクリックします。
5. 「アプリケーション」をクリックします。

ここから、リストされている機能に対するユーザーまたはグループの設定を変更することができます。また、「親」の機能の設定を変更することにより、階層グループ内のすべての機能の設定を変更することもできます。

iSeries FTP サーバーの保護の詳細については、トピック『FTP の保護』を参照してください。

着信 FTP ユーザーのモニター

FTP の使用についてログを記録し、検討することによって、活動をモニターしたり、外部からの攻撃について調べることができます。着信 FTP ユーザーをモニターする場合、以下のステップに従います。

1. iSeries ナビゲーターで、「ユーザーのサーバー」→「ネットワーク」→「サーバー」→「TCP/IP」と展開します。

2. 右側のペインで、「FTP」を右クリックし、「サーバー・ジョブ」を選択します。
3. 「FTP サーバー・ジョブ」パネルが表示されます。「現行ユーザー」欄に、サーバー・ジョブにログインしたユーザーが表示されます。ログインしているユーザーがいない場合は、Qtcp が表示されます。
F5 を押すか、または「表示」→「最新表示」と選択し、ディスプレイを更新します。

これらのジョブ名の形式は、*QTFTPnnnnn* です。nnnnn は、ランダムに生成された番号です。

FTP サーバー・ジョブを開始するには、『FTP サーバーの始動および停止』のステップに従います。

FTP サーバーの管理

ファイル転送プロトコル (FTP) を利用すると、ネットワーク全体でファイルの送受信や共有を行うよう iSeries サーバーを設定することができます。FTP は、FTP クライアントと FTP サーバーの 2 つの部分で構成されます。ユーザーは FTP クライアントと対話します。FTP クライアントは FTP サーバーと対話します。通常、ユーザーと FTP サーバーが直接対話することはありません。以下のトピックは、FTP サーバーを管理する際に役立ちます。

- FTP サーバーの始動および停止
- 使用可能な FTP サーバー数の設定
- 構成可能なサブシステムのサポートによる FTP サーバーのパフォーマンスの改善

FTP サーバーの始動および停止

FTP サーバーの始動と停止は、iSeries ナビゲーターを使用して行うことができます。FTP のアクセス方法についての説明は、『iSeries ナビゲーターを使用した FTP サーバー』を参照してください。

FTP サーバーを始動するには、以下のステップを実行します。

1. iSeries ナビゲーターで、「ユーザーの iSeries サーバー」→「ネットワーク」→「サーバー」→「TCP/IP」と展開します。
2. 右側のペインで、「FTP」を右クリックし、「開始」を選択します。

FTP サーバーを停止するには、以下のステップを実行します。

1. iSeries ナビゲーターで、「ユーザーの iSeries サーバー」→「ネットワーク」→「サーバー」→「TCP/IP」と展開します。
2. 右側のペインで、「FTP」を右クリックし、「停止」を選択します。

使用可能な FTP サーバー数の設定

▶ 今後のクライアント接続に備えて、使用可能なサーバーの最小数を指定できます。この値を設定するには、「FTP プロパティ (FTP Properties)」ページに移動し、1 から 20 までの数を「始動するサーバーの初期数 (Initial number of servers to start)」に指定します。値に 1 を指定すると、FTP サーバーに着信する接続が遅れます。推奨値は、3 です。

クライアントが iSeries FTP サーバーに接続すると、サーバーはクライアントに接続していないサーバーで活動状態にあるものの数および「始動するサーバーの初期数 (Initial number of servers to start)」に指定されている値を調べます。この初期サーバー値が使用可能なサーバー数よりも大きい場合は、この 2 つの値が等しくなるように追加のサーバーが始動されます。この初期サーバー値が使用可能なサーバー数よりも小さい場合は、処置は取られません。この初期サーバー値への変更は、次のクライアント接続時、上記のプロセスが活動化されたときに有効になります。

たとえば、同時に 5 つの FTP クライアント・セッションが確立されていて、初期サーバー値が 10 に設定されている場合、15 台のサーバーが稼働することになります。15 台のサーバーのうち、5 台のサーバーは活動状態にある 5 つのクライアント・セッション向けに使用され、10 台が使用可能なサーバーです。使用可能なサーバーの数は、初期サーバー値よりも多くなることがあります。これと同じ例で、5 つのクライアントがセッションを終了し、開始されているセッションが他にない場合、使用可能なサーバーが 15 台になります。 <<

構成可能なサブシステムのサポートによる FTP サーバーのパフォーマンスの改善

省略時のサブシステム (QSYS/QSYSWRK) は、IBM 提供のサーバー・ジョブの多くに使用されます。省略時のサブシステムとは異なるサブシステムを使用すると、リソースの共有の必要性がなくなるため、FTP のパフォーマンスが改善されることがあります。

FTP サーバーにサブシステムを構成するには、これらのステップを行います。

1. iSeries ナビゲーターで、「ユーザーの iSeries サーバー」 → 「ネットワーク」 → 「サーバー」 → 「TCP/IP」と展開します。
2. 「FTP」を右クリックし、「プロパティ」を選択します。
3. 「FTP プロパティ (FTP Properties)」ページで、「サブシステム記述 (Subsystem description)」を選択します。
4. サブシステム記述および事前定義ライブラリーを指定します。

指定されたサブファイルが存在しない場合は、FTP はそのサブファイルを、経路指定テーブル項目とジョブ記述と共に作成します。サーバーの開始ジョブを実行すると、その開始ジョブは新規に作成されたサブシステムに対してパラメーターを指定し、そのサブシステムにおけるバッチ始動のためのサーバー・ジョブを投入します。

iSeries での FTP クライアントの使用

FTP クライアントを使用すると、Root、QSYS.Lib、QOpenSys、QOPT、および QFileSvr.400 ファイル・システムにおけるファイルも含め、iSeries サーバーで検出したファイルを転送することができます。また FTP クライアントを使用すると、文書ライブラリー・サービス (QDLS) ファイル・システム内のフォルダーと文書を転送することもできます。クライアントのサブコマンドがファイルから読み取られ、これらのサブコマンドに対する応答はファイルに書き込まれる無人バッチ・モードでは、FTP クライアントを対話式に実行することができます。また、FTP クライアントには、使用しているシステム上のファイルを操作するためのその他のフィーチャーも含まれています。

クライアントのユーザー・インターフェースからは、FTP サーバーに要求を出すためのクライアントのサブコマンドを入力することができます。これらの要求の結果は、このユーザー・インターフェースに表示されます。

クライアントとサーバー間でファイルを転送するために、2 つの接続が確立されます。FTP サーバー・コマンドを使ってサーバーにサービスを要求する場合は、制御接続が使用されます。サーバーは、その要求をどのように処理したかを示すために、応答をクライアントに送り返します。2 つ目の接続はデータ接続と呼ばれ、ファイルのリストと実際のファイル・データの転送に使用されます。

クライアントおよびサーバーはどちらも、常駐ファイル・システムとインターフェースするデータ転送機能を持っています。この機能により、データをローカル・ファイル・システムとの間で読み取りまたは書き込みしたり、データ接続との間でデータを読み書きします。

クライアント・セッションの開始と終了

クライアント・セッションの開始と終了の方法を説明しています。

サーバーのタイムアウトにおける考慮事項

接続がタイムアウトにならないようにする方法を説明しています。

FTP を使用したファイル転送

FTP を使用したファイルの送受信の方法について説明しています。

FTP のバッチ・ジョブ

無人モードで FTP を実行する方法の例を示しています。

クライアント・セッションの開始と終了

このトピックには、iSeries サーバーでの FTP クライアントの使用についての詳細が記載されています。

FTP クライアント・セッションの開始

FTP クライアント・セッションの終了

FTP クライアント・セッションの開始

FTP クライアント機能を開始する前に、以下の情報を用意する必要があります。

- ファイルを送信または取得するシステムの名前や IP アドレス。
- ファイル転送を行うリモート・システムのログオン ID およびパスワード (必要な場合)。
- 作業 (送受信など) を行うファイルの名前。

TCP/IP ファイル転送の開始 (STRTCPFTP “remotesystem”) は、ローカルの iSeries サーバーでクライアント・セッションを開始し、指定されているリモート・システム上の FTP サーバーとの接続をオープンします。たとえば、コマンド FTP myserver.com を入力すると、使用している iSeries サーバー・システムでクライアント・セッションが開始され、リモートの myserver.com システムの FTP サーバーとの接続がオープンします。追加のパラメーターを指定することもできますし、リモート・システムを指定せずに STRTCPFTP と入力することで、パラメーターの入力の指示を出すこともできます。

TCP/IP ファイル転送の開始 (STRTCPFTP)

選択項目を入力して、実行キーを押してください。

リモート・システム > MYSERVER.COM

コード化文字セット ID	*DFT	1-65533, *DFT
ポート	> *SECURE	1-65535, *DFT, *SECURE
セキュア接続	*DFT	*DFT, *NONE, *SSL, *IMPLICIT
データ保護	*DFT	*DFT, *CLEAR, *PRIVATE

リモート・システム名を指定すると、追加情報を指定するように指示が出されます。以下に使用可能なオプションの要約が記載されており、ヘルプ・フィールドから追加の詳細を入手できます。

リモート・システム (RMTSYS)

ファイルの転送を行うリモート・システム名を指定します。可能値は以下のとおりです。

***INTNETADR**

IP アドレス (INTNETADR) パラメーターを入力するよう指示が出されます。IP アドレスは、nnn.nnn.nnn.nnn という形式で指定されます。nnn は、0 から 255 までの 10 進数です。

remote-system

ファイルの転送を行うリモート・システム名を指定します。

コード化文字セット識別 (CCSID)

FTP TYPE モードを ASCII に設定している場合に 1 バイト文字セット (SBCS) ASCII ファイル転送に使用される、ASCII コード化文字セット識別コード (CCSID) を指定します。可能値は以下のとおりです。

***DFT**

CCSID 値 00819 (ISO 8859-1 8 ビット ASCII) が使用されます。

CCSID-value

要求された CCSID 値が使用されます。有効な ASCII SBCS CCSID が要求されたことを確認するために、この値の妥当性検査が行われます。

ポート (PORT)

FTP サーバーへの接続に使用されるポート番号を指定します。通常は、「事前に割り当て済みの」ポート番号 21 を使用して、FTP サーバーに接続します。状況によっては、21 以外のポートを使用して FTP サーバーに連絡することもできます。そのような状況では、ポート・パラメーターを使用して、接続するサーバーのポートを指定することができます。可能値は以下のとおりです。

***DFT**

値 00021 が使用されます。

***SECURE**

値 00990 が使用されます。ポート 990 は、トランスポート層セキュリティー (TLS) または Secure Sockets Layer (SSL) プロトコルを即座に使用してデータを暗号化するセキュアな FTP サーバー向けに予約されています。

port_value

要求したポート値が使用されます。この値が適切な範囲内にあることを確認するために、この値には妥当性検査が行われます。

注: 990 が指定された場合、FTP クライアントは、*SECURE が指定されたときと同様の機能を実行します。

▶ セキュア接続 (SECNN)

FTP 制御接続で転送される情報 (FTP サーバーとのセッションの認証に使用されるパスワードなど) の保護に使用されるセキュリティー機構のタイプを指定します。トランスポート層セキュリティー (TLS) と Secure Sockets Layer (SSL) は互換性があるプロトコルであり、データの伝送中にデータが調べられるのを防止し、データの損失や破壊が起こらなかったことを検証するために暗号を使用します。

注: FTP クライアント・サブコマンド SECOPEN を使用して、FTP クライアント・セッション中に保護されている FTP 接続をオープンすることができます。

可能値は以下のとおりです。

***DFT**

PORT パラメーターに *SECURE または 990 が指定されている場合、*IMPLICIT が使用されます。指定されていない場合、*NONE が使用されます。

***IMPLICIT**

FTP クライアントは指定されている FTP に接続するときに、(AUTH サブコマンドをサーバーに送信せずに) 即座に TLS/SSL を使用しようとしています。サーバーが指定されているポートで暗黙 TLS/SSL をサポートしない場合、または TLS/SSL の折衝が何らかの理由で失敗する場合、接続は終了します。

***SSL**

指定されている FTP サーバーに接続した後、FTP クライアントは TLS/SSL で保護されるセッションを要求する AUTH (許可) サブコマンドを送信します。サーバーが TLS/SSL をサポートする場合、TLS/SSL 折衝が行われます。サーバーが TLS/SSL をサポートしない場合、または TLS/SSL の折衝が何らかの理由で失敗する場合、接続は終了します。

***NONE**

FTP クライアントは、指定されている FTP サーバーへの制御接続に暗号を使用しません。 <<

>> データ保護 (DTAPROT)

FTP データ接続で転送される情報に使用するデータ保護のタイプを指定します。この接続は、ファイル・データおよびディレクトリー・リストを転送するために使用されます。制御接続が保護されていない場合、FTP プロトコルはデータ接続の保護を許可しません。

注: これに続き、FTP クライアント・サブコマンド SECDATA を使用してデータ保護のレベルを変更できます。セキュアな制御接続が確立された後に、FTP クライアントは、FTP サーバー・サブコマンド PROT を使用して、指定されているデータ保護を要求します。

可能値は以下のとおりです。

***DFT**

SECCNN パラメーターに制御接続の保護が指定されている場合、*PRIVATE が使用されます。指定されていない場合、*CLEAR が使用されます。

***PRIVATE**

FTP データ接続で送信される情報は暗号化されます。SECCNN パラメーターに FTP 制御接続が暗号化されないことが指定されている場合は、*PRIVATE を指定できません。

***CLEAR** FTP データ接続で送信される情報は暗号化されません。 <<

発信する ASCII/EBCDIC テーブル (TBLFTPOUT)

FTP クライアントに発信するすべてのデータをマップするために使用するテーブル・オブジェクトを指定します。発信するデータは EBCDIC から ASCII にマップされます。TBLFTPOUT に指定されているテーブル・オブジェクトがない場合は、CCSID パラメーターを使用して発信するマッピングを判別します。可能値は以下のとおりです。

***CCSID**

CCSID パラメーターを使用して発信するマッピングを判別します。

***DFT**

CCSID パラメーターを使用して発信するマッピングを判別します。

発信するマッピング・テーブルの名前は、以下のライブラリー値のうち 1 つを使用して修飾できます。

***LIBL**

最初に一致するものが見つかるまで、ジョブ・ライブラリー・リストのユーザー部分とシステム部分にあるすべてのライブラリーが検索されます。

***CURLIB**

ジョブの現行ライブラリーが検索されます。ジョブの現行ライブラリーが指定されていない場合は、QGPL ライブラリーが使用されます。

ライブラリー名

検索するライブラリー名を指定します。

outgoing-mapping-table

FTP クライアントが、発信するデータをマップするために使用するテーブル・オブジェクトを指定します。

着信する ASCII/EBCDIC テーブル (TBLFTPIN)

FTP クライアントに着信するすべてのデータをマップするために使用するテーブル・オブジェクトを指定します。着信するデータは、ASCII から EBCDIC にマップされます。TBLFTPIN にテーブル・オブジェクトが指定されていない場合は、CCSID パラメーターを使用して着信するマッピングを判別します。可能値は以下のとおりです。

***CCSID**

CCSID パラメーターを使用して着信するマッピングを判別します。

***DFT**

CCSID パラメーターを使用して着信するマッピングを判別します。

着信するマッピング・テーブルの名前は、以下のライブラリー値のうち 1 つを使用して修飾できます。

***LIBL**

最初に一致するものが見つかるまで、ジョブ・ライブラリー・リストのユーザー部分とシステム部分にあるすべてのライブラリーが検索されます。

***CURLIB**

ジョブの現行ライブラリーが検索されます。ジョブの現行ライブラリーが指定されていない場合は、QGPL ライブラリーが使用されます。

ライブラリー名

検索するライブラリー名を指定します。

incoming-mapping-table

FTP クライアントが、着信するデータをマップするために使用するテーブル・オブジェクトを指定します。

システム間でファイルを転送するステップについては、『FTP を使用したファイルの転送』のトピックを参照してください。

FTP クライアント・セッションの終了

FTP セッションは QUIT サブコマンドで終了します。QUIT サブコマンドは、リモート・ホストとの接続をクローズし、iSeries サーバーでの FTP セッションを終了します。あるいは、F3 (終了) を押しても FTP クライアント・セッションの終了を確認できます。

サーバーのタイムアウトにおける考慮事項

▶ 非活動タイムアウト値については、多少考慮する必要があります。この値は、FTP サーバーが非活動状態になってから、セッションをクローズするまでの時間 (秒単位) です。一部のリモート・サーバーでは、クライアントがこの値を変更できます。たとえば、iSeries では、FTP サーバーの TIME サブコマンドがサポートされており、『QUOTE (FTP サーバーへのサブコマンドの送信)』で説明しているように、FTP クライアントの QUOTE サブコマンドで TIME サブコマンドをサーバーに送信することができます。UNIX サーバーでは、多くの場合 SITE IDLE サブコマンドがサポートされています。

ローカルの iSeries サブコマンドを SYSCMD サブコマンドまたは F21 で使用する場合、クライアントとサーバー間の対話は起こりません。そのため、これらのローカルの iSeries コマンドの実行時間が、サーバーの非活動タイムアウト時間よりも長くなると、サーバーは接続をクローズします。接続がクローズされた場合、リモート・システム (サーバー) へのログオンに関する注意で説明しているように、OPEN コマンド (OPEN <remote system name>) および USER コマンドを使用して、サーバーに再度ログオンする必要があります。◀

FTP を使用したファイル転送

次のステップに従って、FTP を使用してファイルを転送します。

1. 次の情報を収集します。
 - リモート・コンピューターの TCP/IP 名または IP アドレス
 - リモート・コンピューターのログオン名とパスワード (リモート・コンピューターが匿名 FTP をサポートしていない場合)
 - 転送したいファイルの名前と場所
 - 送信先の場所
 - 使用する ファイル転送タイプです。ASCII、EBCDIC、または BINARY があります。
 - トランスポート層セキュリティ (TLS) または Secure Sockets Layer (SSL) で保護した接続を使用するかどうか。
2. コマンド行で、**FTP** と入力し、**Enter** キーを押します。
3. プロンプトで、リモート・コンピューター・システムの TCP/IP 名または IP アドレスを入力し、**Enter** キーを押します。使用できる名前または IP アドレスの例は次のとおりです。

```
remote.systemname.com  
または  
110.25.9.13
```
4. コード化文字セット識別 (CCSID) を入力します。特定の CCSID が必要なのかどうか分からないときは、省略時の値 (*DFT) をそのまま使用します。
5. ▶ セキュアな接続を使用してパスワードやデータを保護する場合は、ポート値を *SECURE に指定します。◀

6. **Enter** キーを押して、接続を開始します。FTP クライアントにより、リモート・システムに正常に接続できたことを示すメッセージが表示されます。




注: ポートに *SECURE を指定したが、サーバーが指定されたポートで暗黙 TLS/SSL をサポートしない場合、または TLS/SSL の折衝が何らかの理由で失敗する場合、接続は終了します。



7. ファイル転送のタイプを変更するには、以下のようにします。
 - a. EBCDIC に切り替えるには、**EBCDIC** と入力し、**Enter** キーを押してからファイルを転送します。
 - b. BINARY に切り替えるには、**BINARY** と入力し、**Enter** キーを押してからファイルを転送します。
 - c. 省略時のタイプである ASCII に戻るには、**ASCII** と入力し、**Enter** キーを押してからファイルを転送します。
8. これでファイル転送の準備が整いました。次の手順に従ってください。
 - a. **CD** に続けてディレクトリーの名前を入力します。 **Enter** キーを押します。
 - b. 次のいずれかを実行します。
 - サーバー・システムからクライアント・システムにファイルを転送するには、次のように **GET** の後にファイル名を入力します。
GET myfile.txt
 - クライアント・システムにあるファイルをサーバー・システムに転送するには、次のように **PUT** の後にファイル名を入力します。
PUT myfile.txt
9. FTP クライアント・セッションを終了して、iSeries のコマンド行に戻るには、FTP サブコマンド **QUIT** を入力します。

FTP のバッチ・ジョブ

FTP クライアントは、対話式に実行するだけでなく、不在モードで実行することができます。このトピックでは、不在モードの例を 2 つ (単純な例と複雑な例) 示します。その他の例については、「V4 TCP/IP

for AS/400: More Cool Things Than Ever」  (約 744 ページ) のセクション 6.7 の『Batch FTP』を参照してください。

バッチ FTP: 単純な例

次の例は、一つのファイルをリモート・システムから正常に転送する、単純なバッチ・ファイル転送の例です。

コンポーネントは以下のとおりです。

- CL プログラム
- FTP コマンドの入力ファイル
- FTP メッセージの出力ファイル

CL プログラム

```
*****  
ITSOLIB1/QCLSRC BATCHFTP:  
-----
```

```

PGM
OVRDBF FILE(INPUT) TOFILE(ITSOLIB1/QCLSRC) MBR(FTPCMDS)
OVRDBF FILE(OUTPUT) TOFILE(ITSOLIB1/QCLSRC) MBR(OUT)
FTP RMTSYS(SYSxxx)
ENDPGM

```

注: このサンプルを ILECL で作成した場合に正常に実行させるには、OVRSCOPE(*CALLLVL) を OVRDBF コマンドに追加する必要があります。

BATCHFTP プログラムは、INPUT パラメーターを、ソース物理ファイル ITSOLIB1/QCLSRC MBR(FTPCMDS) に変更します。出力は MBR(OUT) に送信されます。

入カコマンド・ファイル

```

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

```

必要な FTP サブコマンドは、FTPCMDS ファイルに示してあります。

出力メッセージ・ファイル

```

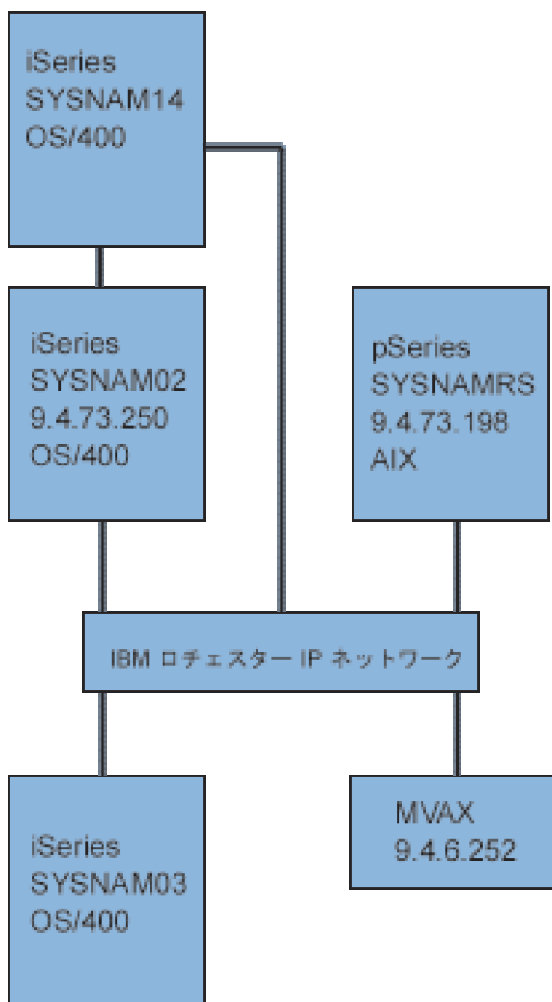
*****
FTP Output Redirected to a File
FTP Input from Overridden File
Connecting to host name SYSxxx
at address x.xxx.xx.xxx using port 21.
220-QTCP at SYSxxx.sysnam123.ibm.com.
220 Connection will close if idle more than 5 minutes.
Enter login ID (itso):
> ITSO ITSO
331 Enter password.
230 ITSO logged on.
OS/400 is the remote operating system. The TCP/IP version is "V3R1M0".
250 Now using naming format "0".
257 "QGPL" is current library.
Enter an FTP subcommand.
> CD ITSOLIB1
Enter an FTP subcommand.
250 Current library changed to ITSOLIB1.
> SYSCMD CHGCURLIB ITSOLIB2
Enter an FTP subcommand.
> GET QCLSRC.BATCHFTP QCLSRC.BATCHFTP (REPLACE
200 PORT subcommand request successful.
150 Retrieving member BATCHFTP in file QCLSRC in library ITSOLIB1.
250 File transfer completed successfully.
147 bytes transferred in 0.487 seconds. Transfer rate 0.302 KB/sec.
Enter an FTP subcommand.
> QUIT
221 QUIT subcommand received.
*****

```

出力ファイルは表示されています。このファイルを処理して、エラーがあれば QSYSOPR にエラー・メッセージを表示するプログラムを作成するのは簡単です。FTP エラー・メッセージの番号は、4 または 5 で始まります。

バッチ FTP: 複雑な例

次の例は、バッチ・モードで、複数のリモート・ホストから中央側の iSeries にファイルを検索する方法を示しています。



*

iSeries SYSNAM03 上のユーザー GWIL は、以下を行うものとしてします。

1. ホスト SYSNAMRS (RS/6000^R) および MVAX (VAX) からファイルを検索する。
2. SYSNAMRS から検索されたファイルは、FTP を使用して SYSNAM02 (他の iSeries) に転送する。
3. SYSNAM02 から、TCP/IP を使用して、そのファイルを iSeries SYSNAM14 に送信する。

CL プログラムの作成と FTP の開始

1. 直前の例のように、FTP はコマンド INPUT およびメッセージ OUTPUT にディスプレイ装置を使用します。バッチ・モードで使用するにはこれを変更する必要があります。この変更は、OVRDBF コマンドを使用して、これらのファイルをバッチで使用するファイルで上書きすることによって行います。

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

2. CL プログラム・ファイルに含まれている STRTCPFTP コマンドでは、ホスト名または IP アドレスが必須パラメーターです。ただし、CL プログラム・ファイルの代わりに、入力コマンド・ファイルにリ

モート・システムを指定する場合には、必須の構文を完結させるために、STRTCPFFTP コマンドにダミー・ホスト名を指定しておく必要があります。このダミー名は、実在しないホスト名でも実在するホスト名でもかまいません。実名の場合には、入力コマンドの最初の項目はユーザー ID およびパスワードで、2 番目の項目は CLOSE サブコマンドでなければなりません。実在する名前でない場合には、これらの項目は不要です。最初の項目は、相手先サーバー・システムに接続するための OPEN サブコマンドにします。

```
FTP RMTSYS(LOOPBACK)
```

FTP は入力ファイル进行处理し、出力ファイル (FTPLOG) にメッセージを書き込みます。

- FTP アプリケーションの終了後に、上書き情報を次のように削除します。

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

バッチ FTP 用の CL プログラムは、以下の SYSNAM01 システム上の例を参考にしてください。

```

桁 . . . . . : 1 71          走査検索          GERRYLIB/QCLSRC
SEU==>          FTPBATCH
FMT **  ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
***** データの始め *****
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
***** データの終わり *****

F3=終了  F5=最新表示  F9=コマンドの複写  F10=カーソル  F11=切り替え  F12=取り消し
F16=検索の反復  F24=キーの続き

(C) COPYRIGHT IBM CORP. 1981, 1999.

```

図 1. バッチ FTP 用 FTPBATCH CL プログラム

FTP 入力ファイル (FTPCMDS) の作成

このファイルは、必要なすべての FTP クライアント・サブコマンドを含んでいなければなりません。これによって、サーバーへの接続とログオン、ファイル転送用の設定と実行、サーバー接続のクローズ、およびクライアント・セッションの終了を行います。次の例は、2 つの異なるリモート・システムへのファイル転送で使用されるサブコマンドです。

```

桁 . . . . . :   1 71           走査検索           GERRYLIB/QCLSRC
SEU==>                               FTPCMDS
FMT **  ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
          ***** データの始め *****
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
          ***** データの終わり *****
F3=終了  F5=最新表示  F9=コマンドの複写  F10=カーソル  F11=切り替え  F12=取り消し
F16=検索の反復  F24=キーの続き

```

図 2. 2 つのリモート・システムへのファイルの転送

以下で、図 2 に示されている FTP クライアント・サブコマンドについて説明します。ディスプレイ上の行番号は、以下の番号に対応します。

0001

クライアント iSeries SYSNAM03 内のダミー接続用ユーザー ID およびパスワード。

0002

iSeries SYSNAM03 内のダミー接続のクローズ。

0003

RISC System/6000^R SYSNAMRS への制御接続のオープン。

0004

USER サブコマンドと SYSNAMRS のユーザー ID およびパスワード。

注: バッチ・モードで FTP を実行中である場合、USER サブコマンドは OPEN サブコマンドの次に指定する必要があります。USER サブコマンドには、ログオン・ユーザー ID およびパスワード・パラメーターを両方指定する必要があります。これは FTP のオンラインでの対話式操作とは異なります。FTP がオンラインで対話実行中は、クライアントは自動的に USER サブコマンドを開始して、ユーザーに対してログオン ID のプロンプトが出されます。FTP をバッチ・モードで実行中は、自動 USER サブコマンドはありません。

0005

ASCII データの転送 (iSeries 上で EBCDIC に、または EBCDIC へ変換される)。

0006

クライアント iSeries で実行される CL コマンド: ファイルの削除。このコマンドの代わりに、次のステートメントでパラメーター (REPLACE を使用することも可能です)。

0007

RISC System/6000 システムからのファイルの検索。

0008

RISC System/6000 SYSNAMRS への制御接続のクローズ。

0009

VAX MVAX への接続のオープン。

0010

USER サブコマンドと、MVAX のユーザー ID およびパスワード。

0011

VAX からのファイル検索と既存 iSeries ファイルの取り替え。

0012

VAX MVAX への制御接続のクローズ。

0013

リモート iSeries SYSNAM02 への制御接続のオープン。

0014

USER サブコマンドと、SYSNAM02 のユーザー ID およびパスワード。

0015

EBCDIC データ転送 (iSeries から iSeries への転送として)。

0016

TCP/IP を使用した iSeries ファイルの iSeries SYSNAM02 への送信。

0017

TCP/IP ネットワークでの、サーバー iSeries SYSNAM03 からリモート iSeries SYSNAM14 へのファイルの転送。

0018

iSeries SYSNAM02 への制御接続のクローズ。

0019

FTP アプリケーションの終了。

FTPbatch ジョブを投入するための CL プログラムの作成

ファイル転送をスケジュールし、その転送を不在で実行し、FTPbatch ジョブの投入をする CL プログラムを作成します。次の例では、ファイル転送は翌金曜日の 17:00 に不在モードで実行するように予定されています。

```

桁 . . . . . :   1 71           走査検索                       GERRYLIB/QCLSRC
SEU==>                                           FTPSUBMIT
FMT **  ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
***** データの始め *****
0001.00 PGM
0002.00          SBMJOB      CMD(CALL PGM(GERRYLIB/FTPbatch)) +
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
***** データの終わり *****

F3=終了  F5=最新表示  F9=コマンドの複写  F10=カーソル  F11=切り替え  F12=取り消し
F16=検索の反復  F24=キーの続き

(C) COPYRIGHT IBM CORP. 1981, 1999.

```

図 3. バッチ FTP ジョブの投入をする CL プログラム

FTP 出力ファイルのエラー検査

スケジュールされて実行中は、FTP は上で示すように、ファイル・メンバー FTPLOG にデータを作成します。オリジナルのステートメントに対応する、ファイル・メンバー FTPLOG 内のデータは、両方の例で確認できます。

```

          ポート 21 を使用してアドレス 127.0.0.1 でホスト LOOPBACK へ接続中。
220-QTCP at localhost.
220 CONNECTION WILL CLOSE IF IDLE MORE THAN 5 MINUTES.
ログイン ID (gwil) を入力してください。

>>>GWIL ****
331 ENTER PASSWORD.
230 GWIL LOGGED ON.
   OS/400 IS THE REMOTE OPERATING SYSTEM.  THE TCP/IP VERSION IS "V4R2M0".
250 NOW USING NAMING FORMAT "0".
257 "QGPL" IS CURRENT LIBRARY.
FTP サブコマンドを入力してください。

> CLOSE
221 QUIT SUBCOMMAND RECEIVED.
FTP サブコマンドを入力してください。

> OPEN SYSNAMRS
ポート 21 を使用してアドレス 9.4.73.198 でホスト SYSNAMRS へ接続中。
220 sysnamrs.sysnam123.ibm.com FTP server (Version 4.9 Thu Sep 2 20:35:07 CDT
    1993) ready.
FTP サブコマンドを入力してください。

```

図 4. FTPbatch プログラム実行後の FTP 出力 (FTPLOG) (パート 1)

```
> USER root ****
331 Password required for root.
230 User root logged in.
UNIX Type: L8 Version: BSD-44
FTP サブコマンドを入力してください。

> ASCII
200 Type set to A; form set to N.
FTP サブコマンドを入力してください。

> SYSCMD DLTF FILE(GERRYLIB/RS6)
FTP サブコマンドを入力してください。

> GET /Itsotest GERRYLIB/RS6/RS7
200 PORT command successful.
150 Opening data connection for /Itsotest (467 bytes).
226 Transfer complete.
467 bytes transferred in 2.845 seconds. Transfer rate 0.167 KB/sec.
FTP サブコマンドを入力してください。
```

図 5. FTPBATCH プログラム実行後の FTP 出力 (FTPLOG) (パート 2)

```
> CLOSE
221 Goodbye.
FTP サブコマンドを入力してください。

> OPEN MVAX
ポート 21 を使用してアドレス 9.4.6.252 でホスト mvax へ接続中。
220 FTP Service Ready
FTP サブコマンドを入力してください。

> USER TESTER *****
331 User name TESTER received, please send password
230 TESTER logged in, directory $DISK1:[TESTER]
FTP サブコマンドを入力してください。

GET SCREEN1.FILE GERRYLIB/VAX.VAX (REPLACE
200 PORT Command OK.
125 ASCII transfer started for $DISK1:[TESTER SCREEN1.FILE;1(266586 bytes)
226 File transfer completed ok.
265037 bytes transferred in 8.635 seconds. Transfer rate 30.694 KB/sec.
FTP サブコマンドを入力してください。

> CLOSE
221 Goodbye.
FTP サブコマンドを入力してください。

OPEN SYSNAM02
ポート 21 を使用してアドレス 9.4.73.250 でホスト SYSNAM02 へ接続中。
220-QTCP at SYSNAM02.sysnam123.ibm.com.
220 CONNECTION WILL CLOSE IF IDLE MORE THAN 5 MINUTES.
FTP サブコマンドを入力してください。
```

図 6. FTPBATCH プログラム実行後の FTP 出力 (FTPLOG) (パート 3)


```

> USER GWIL ****
331 ENTER PASSWORD.
230 GWIL LOGGED ON.
   OS/400 IS THE REMOTE OPERATING SYSTEM.  THE TCP/IP VERSION IS "V4R2M0".
250 NOW USING NAMING FORMAT "0".
257 "QGPL" IS CURRENT LIBRARY.
FTP サブコマンドを入力してください。

> EBCDIC
200 REPRESENTATION TYPE IS EBCDIC NONPRINT.
FTP サブコマンドを入力してください。

> PUT GERRYLIB/RS6.RS6 GERRYLIB/RS6.RS6
200 PORT SUBCOMMAND REQUEST SUCCESSFUL.
150 SENDING FILE TO MEMBER RS6 IN FILE RS6 IN LIBRARY GERRYLIB.
250 FILE TRANSFER COMPLETED SUCCESSFULLY.
   0.148 秒に 467 バイトが転送された。転送速度は 3.146 KB/秒です。
FTP サブコマンドを入力してください。

> RCMD SNDNETF FILE(GERRYLIB/RS6) TOUSRID((GERRYLIB SYSNAM14))
250 COMMAND SNDNETF FILE(GERRYLIB/RS6) TOUSRID((GWIL SYSNAM14))
   SUCCESSFUL.
FTP サブコマンドを入力してください。

```

図 7. FTPBATCH プログラム実行後の FTP 出力 (FTPLOG) (パート 4)

```

> CLOSE
221 QUIT SUBCOMMAND RECEIVED.
FTP サブコマンドを入力してください。
> QUIT
(このコマンドが FTP アプリケーションを終了させます)

```

図 8. FTPBATCH プログラム実行後の FTP 出力 (FTPLOG) (パート 5)

この出力を使って、FTP 処理で発生する可能性があるエラーの検査を行います。この検査は、ユーザー自身の目で行うこともできますが、エラー応答コードを検査するプログラムを実行して行うこともできます。3 桁の FTP エラー応答コードは、4 または 5 で始まります。「467 bytes transferred...」などのメッセージはエラーではありませんので注意が必要です。

サンプル・プロシージャ: サンプル REXX プロシージャおよびサンプル物理ファイル・メンバーが、TCP/IP 製品の一部として提供されています。ライブラリー QTCP 内のファイル QATMPINC は、以下の 2 つのメンバーを含んでいます。

- 入出力バッチ・ファイルを指定する REXX ソース・コード、および開始 FTP を含む BATCHFTP。
- FTP へのログオンと、FTP の実行に必要なサブコマンドおよびデータを含む、BFTPFILE。

FTP 参照情報

➤ 以下のトピックには、FTP サーバーおよび FTP クライアントで作業する際に役立つ情報が記載されています。

FTP サーバー・サブコマンド

これらのコマンドは、クライアントとサーバー間の通信を表します。このトピックでは、iSeries CL と同等の機能を持つ、iSeries FTP サーバーに固有のサブコマンドを説明しています。

FTP クライアント・サブコマンド

FTP クライアント・サブコマンドを使用すると、リモート FTP サーバーとの接続の確立、ライブラリーおよびディレクトリーのナビゲート、ファイルの作成、削除、転送を行うことができます。

FTP 出口プログラム

FTP を保護するために、FTP 出口プログラムを利用する。FTP サーバーは、特定の出口点を通じて各出口プログラムとの通信を行います。このトピックには、パラメーターの説明およびコードの例が記載されています。

その他の参照情報

- データの転送方法
- ファイル・システムと命名規則
- FTP サーバーの応答状況メッセージ
- FTP サーバーの構文規則
- FTP クライアントの構文規則



FTP サーバー・サブコマンド

このトピックでは、FTP サーバー・サブコマンドについて説明します。FTP クライアントは、サーバー・サブコマンドを使用するサーバーと通信します。通常、ユーザーは FTP サーバーと通信しないため、このトピックではサーバー・サブコマンド (その効果と構文規則を含む) と FTP 応答状況メッセージについて説明します。

iSeries FTP サーバーは、以下のサブコマンドをサポートしています。

サブコマンド	用途
ABOR	直前のサブコマンドを取り消します。
ADDM	物理ファイルにメンバーを追加します。
ADDV	物理ファイルに可変長メンバーを追加します。
APPE	指定されたファイルにデータを追加します。
AUTH	現行の FTP セッションに使用される認証メカニズムを定義します。
CDUP	ディレクトリーを親ディレクトリーに変更します。
CRTL	ライブラリーを作成します。
CRTP	物理ファイルを作成します。
CRTS	ソース物理ファイルを作成します。
CWD	作業ディレクトリーまたは作業ライブラリーを変更します。
DEBUG	サーバー・トレースを開始または終了します。
DELE	ファイル、メンバー、または文書を削除します。
DLTF	ファイルを削除します。

サブコマンド	用途
DLTL	ライブラリーを削除します。
HELP	FTP サーバー・サブコマンドに関する情報を取得します。
LIST	ファイルまたはディレクトリー項目をリストします。
MKD	ディレクトリーを作成します。
MODE	データ伝送の形式を指定します。
NLST	ファイルまたはディレクトリーの名前をリストします。
NOOP	サーバーが応答しているかどうかを検査します。
PASS	サーバーへパスワードを送信します。
PASV	次回のデータ接続を受動的に開始するようサーバーに指示します。
PBSZ	データ接続時に送受信され、アプリケーション・レベルでエンコードされたデータに使用される、バッファ保護のためのバッファ・サイズの最大値を定義します。
PORT	クライアントが接続を listen するデータ・ポートを識別します。
PROT	FTP データ接続に使用される保護を定義します。
PWD	現行作業ディレクトリーを表示します。
QUIT	ユーザーをログオフし、接続をクローズします。
RCMD	FTP サーバーに CL コマンドを送信します。
REIN	サーバー上のセッションを再始動します。
RETR	サーバーからデータを検索します。
RMD	ディレクトリーを除去します。
RNFR	変更前のファイル名を指定します。
RNTO	新しいファイル名を指定します。
SITE	使用するサーバーに関する情報を送信します。
STAT	サーバーから状況情報を取得します。
STOR	サーバー上のデータを保管し、既存のファイルを置き換えます。
STOU	サーバー上のデータを保管しますが、既存のファイルは置き換えません。
STRU	ファイルの構造を指定します。
SYST	サーバー上の OS の名前を印刷します。
TIME	FTP サーバーのタイムアウト値を設定します。
TYPE	ファイル転送のタイプを指定します。
USER	サーバーにユーザー・ログオン ID を送信します。
XCUP	親ディレクトリーに変更します。
XCWD	作業ディレクトリーを変更します。
XMKD	ディレクトリーを作成します。
XPWD	現行ディレクトリーまたは現行ライブラリーを表示します。
XRMD	ディレクトリーを除去します。

iSeries FTP サーバーに固有のサブコマンド

iSeries FTP サーバーのサブコマンドには、特殊なコマンド・セットが含まれています。これらのコマンドは、実際には iSeries CL コマンド名を短縮したもので、機能的には iSeries CL コマンドと等価です。これらの特殊なサーバー・サブコマンドは、FTP のアーキテクチャーの制限に合わせて、4 文字にする必要があります。iSeries サーバーは、これらのサブコマンドを受け取ると、次のように解釈します。

- ADDM = ADDPFM (物理ファイル・メンバーの追加)
- ADDV = ADDPVLV (物理ファイル可変長メンバーの追加)
- CRTL = CRTLIB (ライブラリーの作成)
- CRTP = CRTPF (物理ファイルの作成)
- CRTS = CRTSRCPF (ソース物理ファイルの作成)
- DLTF = DLTF (ファイルの削除)
- DLTL = DLTLIB (ライブラリーの削除)

これら特定のサブコマンド以外にも、FTP サーバーのサブコマンド RCMD を使用して、任意の CL コマンドをサーバーに送信することができます。

関連トピック:

- FTP サーバーの構文規則
- FTP サーバーの応答状況メッセージ
- FTP クライアント・サブコマンド: これらのサブコマンドを使用すると、リモート FTP サーバーとの接続の確立、ライブラリーおよびディレクトリーのナビゲート、ファイルの作成、削除、転送を行うことができます。

ADDM (物理ファイル・メンバーの追加)

FTP サーバー・サブコマンド

ADDM *parameters*

parameters

このサブコマンドのパラメーターは、ADDPFM CL コマンドのパラメーターと同じです。

たとえば、iSeries 上のライブラリー RLKAYS の中の物理ファイル GEORGE にメンバー BANANA を追加するには、次のように入力します。

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

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

ADDV (物理ファイル可変長メンバーの追加)

FTP サーバー・サブコマンド

ADDV *parameters*

parameters

このサブコマンドのパラメーターは、ADDPVLM CL コマンドのパラメーターと同じです。

たとえば、iSeries 上のライブラリー RLKAYS の中の物理ファイル GEORGE にメンバー POLEBEAN を追加するには、次のように入力します。

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

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

APPE (既存ファイルへの追加)

FTP サーバー・サブコマンド

APPE FTP サーバー・サブコマンドは、転送されたデータを受け入れて、サーバー・システム上のファイルに保管します。指定したファイルが存在している場合、データはそのファイルに追加されます。ファイルが存在していない場合は、指定したファイルが作成されます。

APPE <i>filename</i>

filename

サーバー・システムでデータを受信するファイル。

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

AUTH (許可)

FTP サーバー・サブコマンド

AUTH FTP サーバー・サブコマンドは、現在の FTP セッションに使用する認証/セキュリティー機構を定義します。このサブコマンドの形式は次のとおりです。

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

パラメーター値:

TLS-C	セキュリティー機構としてトランスポート層セキュリティー (TLS) プロトコルを使用します。データ接続のためのセキュリティー設定は、RFC2228 省略時を使用します。このことは、データ接続には暗黙的保護がないことを意味しています。
TLS-P	セキュリティー機構として TLS プロトコルを使用します。さらに、データ接続も暗黙的に保護します (これは AUTH TLC-C、PBSZ 0、PROT P という順序のコマンドと同等です)
TLS	TLS-C と同義語。
SSL	TLS-P と同義語。

注: TLS プロトコルは、Secure Sockets Layer (SSL) プロトコルと互換性があります。

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

CRTL (ライブラリーの作成)

FTP サーバー・サブコマンド

CRTL *parameters*

parameters

このサブコマンドのパラメーターは、CRTLIB CL コマンドのパラメーターと同じです。

たとえば、iSeries サーバー上に TESTTCP という名前のライブラリーを作成するには、次のように入力します。

```
CRTL TESTTCP
```

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

CRTP (物理ファイルの作成)

FTP サーバー・サブコマンド

CRTP *parameters*

parameters

このサブコマンドのパラメーターは、CRTPF CL コマンドのパラメーターと同じです。

たとえば、レコード長が 80 でメンバー数制限のない、MYFILE という物理ファイルを作成するには、次のように入力します。

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

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

CRTS (ソース物理ファイルの作成)

FTP サーバー・サブコマンド

CRTS *parameters*

parameters

このサブコマンドのパラメーターは、CRTSRCPF CL コマンドのパラメーターと同じです。

たとえば、ライブラリー RLKAYS の中に GEORGE という名前のソース物理ファイルを作成するには、次のように入力します。

```
CRTS FILE(RLKAYS/GEORGE)
```

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

CWD (作業ディレクトリーまたはライブラリーの変更)

FTP サーバー・サブコマンド

作業ディレクトリー、ライブラリー、またはファイル・グループを変更するには、次の CWD FTP サーバー・サブコマンドを使用します。

CWD <i>directory</i>

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

DEBUG (FTP サーバー・トレースをオンに設定)

注: FTP サーバー・トレースは、ソフトウェア問題を IBM へ報告する場合にのみ使用します。この機能によって、システム・パフォーマンスが低下する場合があります。

FTP サーバー・サブコマンド

DEBUG

FTP サーバー・トレースが活動状態でない場合、サーバーがトレースを開始します。サーバーは、別の DEBUG サブコマンドまたは QUIT サブコマンドを受信するまで、引き続きトレースを実行します。サーバーがトレースを終了させる際に、トレース・データをフォーマットする間しばらく時間がかかる場合があります。

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

DELE (ファイルまたは文書の削除)

FTP サーバー・サブコマンド

ファイル、メンバー、または文書を削除するには、次の CWD FTP サーバー・サブコマンドを使用します。

DELE *remotefile*

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

DLTF (ファイルの削除)

FTP サーバー・サブコマンド

DLTF *parameters*

parameters

このサブコマンドのパラメーターは、DLTF CL コマンドのパラメーターと同じです。

たとえば、ライブラリー RLKAYS から MYFILE という名前のファイルを削除するには、次のように入力します。

DLTF FILE(RLKAYS/MYFILE)

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

DLTL (ライブラリーの削除)

FTP サーバー・サブコマンド

DLTL *parameters*

parameters

このサブコマンドのパラメーターは、DLTLIB CL コマンドのパラメーターと同じです。

たとえば、あるライブラリーを削除するには、次のように入力します。

DLTL libname

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

HELP (iSeries リモート・サーバーからのヘルプの取得)

FTP サーバー・サブコマンドに関する情報を取得するには、次の形式の HELP サブコマンドを使用します。

HELP [*subcommand*]

subcommand

情報が必要なサーバー・サブコマンドの名前。たとえば、HELP ADDM と入力すると、iSeries 上の物理ファイルメンバーを追加する方法についてのヘルプ情報が提供されます。

iSeries サーバーが使用する ADDV サブコマンドの構文を調べるには、次のサーバー・サブコマンドを使用します。

HELP ADDV

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

LIST (ファイルのリスト)

FTP サーバー・サブコマンド

ディレクトリー項目、ライブラリーの内容、またはファイル・グループ内のファイルのリストを取得するには、次の LIST FTP サーバー・サブコマンドを使用します。

```
LIST [directory | name]
```

FTP によって転送可能なファイルのみがリストされます。

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則
- SITE (サーバー・システムが使用する情報の送信): このサブコマンドを使用して、LIST サブコマンドによって戻されるデータを変更します。
- UNIX スタイル形式の LIST 情報: このサブコマンドを使用して、LIST サブコマンドによって戻されるデータを表示します。
- iSeries 形式の LIST 情報: このサブコマンドを使用して、LIST サブコマンドによって戻されるデータを表示します。

MKD (ディレクトリーの作成)

FTP サーバー・サブコマンド

ディレクトリーを作成するには、次の MKD FTP サーバー・サブコマンドを使用します。

```
MKD directoryname
```

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

MODE (転送モードの設定)

FTP サーバー・サブコマンド

データ・ビットの伝送方法を指定するには、次の MODE FTP サーバー・サブコマンドを使用して、モードまたはデータ形式を指定します。

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

B

ブロック・モードを指定します。このモードでは、データは 1 つまたは複数の見出しバイトが先行する一連のデータ・ブロックです。

S

ストリーム・モードを指定します。このモードでは、データはバイトのストリームです。ストリーム・モードでは、すべての表示タイプを使用できます。この転送モードは、サーバーがデータ・ブロック情報を転送しないため、より効率的です。

- 注:
1. ストリーム・モードは、iSeries サーバーで使用される省略時の転送モードであり、望ましいモードです。
 2. パラメーターがない場合、サーバーは、MODE の現行設定値を示す応答を戻します。

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

NLST (名前のリスト)

FTP サーバー・サブコマンド

複数のファイル、ファイル・グループ、ディレクトリー、またはライブラリーの名前のみが表示されたリストを取得するには、NLST FTP サーバー・サブコマンドを使用します。

```
NLST [directory | name]
```

FTP によって転送可能なファイルのみがリストされます。

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

NOOP (サーバー応答の獲得)

FTP サーバー・サブコマンド

NOOP FTP サーバー・サブコマンドは、「OK」応答をクライアントに送信します。これ以外には、サーバー処理に影響を与えません。このコマンドは、サーバーが接続され、応答しているかどうかを判別するために、クライアントによって使用されます。NOOP サブコマンドは次のように使用します。

```
NOOP
```

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

PASS (パスワード)

FTP サーバー・サブコマンド

```
PASS password
```

password

サーバー・システムで使用するパスワードを指定するストリング。

注: USER サーバー・サブコマンドは、PASS サーバー・サブコマンドの直前に発行する必要があります。

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

PASV (受動データ接続の使用)

FTP サーバー・サブコマンド

このサーバーに次回のデータ接続を受動的に開始するよう指示するには、次の形式の PASV FTP サーバー・サブコマンドを使用します。

```
PASV
```

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

PBSZ (保護バッファ・サイズ)

FTP サーバー・サブコマンド

PBSZ サブコマンドは、データ接続で送受信されるアプリケーション・レベルでエンコードされたデータで使用される、最大のバッファ・サイズを定義します。このサブコマンドの形式は次のとおりです。

PBSZ *value*

ここで *value* は、10 進整数を表す ASCII 文字ストリングです。

注: RFC2228 では、PBSZ サブコマンドが PROT サブコマンドより前に発行される必要があります。ただし、TLS/SSL はデータ・ブロッキングを処理します。このため、「0」が唯一の有効な値です。

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

PORT (データ・ポート)

FTP サーバー・サブコマンド

クライアントがデータ接続を listen するポートを識別するには、次の形式の PORT FTP サーバー・サブコマンドを使用します。

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

hn

システムの IP アドレスを表します。0 ~ 255 の 10 進数値を表す文字列です。

pn

TCP ポート番号を表します。0 ~ 255 の 10 進数値を表す文字列です。

p1 と p2 の値を TCP ポート番号に変換するには、次の式を使用します。

$$\text{port} = (\text{p1} * 256) + \text{p2}$$

たとえば、次の PORT サブコマンドの場合、

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

ポート番号は 1032、IP アドレスは 9.180.128.180 です。

注: TCP/IP RFC 1122 に指定されているように、接続のクローズ後、2 分間経過しなければ、サーバーは同じクライアント IP アドレスとポート番号に接続できません。クライアント IP アドレスが同じであっても異なるポート番号に接続する場合は、この制限はありません。

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

PROT (データ・チャネル保護レベル)

FTP サーバー・サブコマンド

PROT サブコマンドは、FTP データ接続で使用される保護を定義します (この定義は、ディレクトリー・リストおよびファイル・データの転送で使用されます)。このサブコマンドの形式は次のとおりです。

PROT [C | P]

パラメーター値:

C	クリア。データ接続は、ファイル転送で「生データ」を運び、適応されるセキュリティーはありません。
P	プライベート。データ接続は、TLS/SSL を使用します。これによって、保全性保護と機密性保護が提供されます。

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

PWD (作業ディレクトリーまたはライブラリーの表示)

FTP サーバー・サブコマンド

次の PWD FTP サーバー・サブコマンドを使用すると、サーバーは、現行ディレクトリーまたはライブラリーの名前をクライアントに返します。

```
PWD
```

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

QUIT (FTP サーバー・セッションの終了)

FTP サーバー・サブコマンド

FTP サーバー・サブコマンド QUIT は、クライアント・ユーザーをログオフし、制御接続をクローズします。ファイル転送が進行中の場合、ファイル転送が完了するまでは接続はオープンのままであり、ファイル転送の完了後、サーバーによってクローズされます。

```
QUIT
```

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

RCMD (FTP サーバー・システムへの CL コマンドの送信)

FTP サーバー・サブコマンド

サーバー・サブコマンド RCMD は、FTP サーバー・システム上で iSeries の制御言語 (CL) コマンドを実行するために使用します。RCMD サブコマンド文字列の長さは、1000 文字までです。RCMD サブコマンドには使用可能なプロンプトがないため、RCMD サブコマンド文字列には CL コマンドの実行に必要なすべてのパラメーターを指定する必要があります。

RCMD を通して呼び出された CL コマンドが正常に実行されると、サブコマンドが正常に実行されたことを示すメッセージが表示されます。エラーが発生した場合は、エラーが起きたことを示すメッセージが表示されます。ライブラリー名、ファイル名、またはメンバー名が無効なためにエラーが発生した場合を除き、このメッセージはどのようなエラーであるかは伝えません。

RCMD を使用してファイル削除 (DLTF) コマンドを実行する例を次に示します。

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

mylib は、ファイルを削除するライブラリーの名前です。myfile は、削除するファイルの名前です。

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

リモート・システムで CL コマンドを実行することの代替方法については、『REXEC』にも記載されています。

REIN (システム間のセッションの再初期化)

FTP サーバー・サブコマンド

```
REIN
```

REINITIALIZE サブコマンドは次の操作を行います。

1. 進行中のすべての転送を完了させる。
2. USER セッションを終了し、すべての入出力情報およびアカウント情報を除去する。
3. すべてのサーバー・パラメーターを省略時設定値にリセットする。
4. 制御接続をオープンのままにする。

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

RETR (ファイルの検索)

FTP サーバー・サブコマンド

サーバー・システム上のデータを検索するには、次の RETR FTP サーバー・サブコマンドを使用します。

```
RETR remotefile
```

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

RMD (ディレクトリーの除去)

FTP サーバー・サブコマンド

ディレクトリーを除去するには、次の RMD FTP サーバー・サブコマンドを使用します。

```
RMD directoryname
```

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

RNFR (変更前の名前)

FTP サーバー・サブコマンド

RNFR FTP サーバー・サブコマンドは、ファイルの名前を変更します。このサブコマンドのすぐ後に、RNTO (変更後の名前) サーバー・サブコマンドを発行する必要があります。

RNFR *filename*

filename

変更前のファイル名。

注: iSeries サーバーでは、異なるファイル・システムへファイルを名前変更することはできません。

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

RNTO (変更後の名前)

FTP サーバー・サブコマンド

RNTO FTP サーバー・サブコマンドは、サーバー・システム上のファイル名を変更する場合に、新しいファイル名を指定します。このサブコマンドは、ファイル名を指定した RNFR サブコマンドのすぐ後に発行する必要があります。

RNTO *filename*

filename

変更後のファイル名。

注: iSeries サーバーでは、異なるファイル・システムへファイルを名前変更することはできません。

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

SITE (サーバー・システムが使用する情報の送信)

FTP サーバー・サブコマンド

サーバー・システムが使用する情報を送信する、またはサーバー・システムに固有のサービスを提供するには、次の形式の SITE FTP サーバー・サブコマンドを使用します。

SITE [*parameters*]

iSeries FTP サーバーは、SITE サブコマンドの以下のパラメーターをサポートしています。

LISTFMT 0

サーバーは、V3R1M0 で導入された iSeries 形式の LIST 情報で LIST サブコマンドに関する情報を戻します。iSeries クライアントは、iSeries 形式と UNIX スタイル形式の両方をサポートしています。

LISTFMT 1

サーバーは、UNIX スタイル形式の LIST 情報で LIST サブコマンドに関する情報を戻します。戻される各行の最後の項目はファイルの名前です。iSeries クライアントは、iSeries 形式と UNIX スタイル形式の両方をサポートしています。

LISTFMT

FTP サーバー LISTFMT の現行設定値を示すメッセージを戻します。

- 注: サーバー上の LISTFMT 省略時を変更したい場合は、CHGFTPA コマンドの LISTFMT オプションを使用します。また、iSeries ナビゲーターを使用して、この FTP サーバーのプロパティを設定することもできます。
1. iSeries ナビゲーターで、「ユーザーの iSeries サーバー」 → 「ネットワーク」 → 「サーバー」 → 「TCP/IP」と展開します。
 2. 右側のペインで、「FTP」を右クリックし、「プロパティ」を選択します。
 3. 「初期形式 (Initial Formats)」タブをクリックします。
 4. 「ファイル・リスト」の見出しの下で、サーバーの LISTFMT の省略時として、iSeries または UNIX を使用可能にします。
 5. 「OK」をクリックして、変更内容を受け入れます。

NAMEFMT 0

LIBRARY/FILE.MEMBER 命名形式を使用します。これは、ライブラリー・ファイル・システムのデータベース・ファイル専用の命名形式です。

NAMEFMT 1

パス命名形式を使用します。これは、FTP がサポートするすべてのファイル・システム (ライブラリー・ファイル・システムを含む) の命名形式です。ライブラリー・ファイル・システムを除くすべての iSeries のファイル・システムを処理するには、命名形式 1 を使用する必要があります。

NAMEFMT

サーバー・ファイルの現行の命名形式を含むメッセージを戻します。

- 注: CHGFTPA コマンドの NAMEFMT オプションで、iSeries FTP サーバーの NAMEFMT の省略時の設定を構成することができます。

CRTCCSID *CALC

ASCII ファイル転送時に作成された新しいデータベース・ファイルは、ASCII ファイル転送 CCSID の関連省略時 EBCDIC CCSID を使用します。

CRTCCSID *USER

ASCII ファイルの転送時に作成された新しいデータベース・ファイルは、現行ジョブ CCSID を使用します。この CCSID が 65535 の場合、省略時の CCSID は現行ジョブに対して指定された言語 ID によって決まります。

CRTCCSID *SYSVAL

ASCII ファイルの転送時に作成された新しいデータベース・ファイルは、QCCSID システム値で指定された CCSID を使用します。

CRTCCSID [CCSID-number]

ASCII ファイルの転送時にクライアント上でデータベース・ファイルを作成する際に、CCSID を指定します。この値はサーバーによって妥当性検査されます。

CRTCCSID

現行の FTP クライアント CRTCCSID 設定値が入ったメッセージを表示します。

NULLFLDS 0

サーバーは、NULL フィールドを含んでいるデータベース・ファイルの転送を許可しません。これが省略時です。

NULLFLDS 1

サーバーは、NULL フィールドを含んでいるデータベース・ファイルの転送を許可します。

注: NULL フィールドを含むファイルを転送するには、クライアントとサーバーの両方でこの設定を使用可能にしておく必要があります。NULL フィールドを含んでいるファイルをサーバーが iSeries 以外のサーバーに転送する場合、またはデータのコード・ページ変換が行われる転送タイプの場合、結果は予測できません。

NULLFLDS

FTP サーバー NULLFLDS の現行設定値を示すメッセージを戻します。

TRIM 0

トリム・オプションをオフに設定します。サーバーは、データベース・レコードの末尾ブランクを送信します。

TRIM 1

トリム・オプションをオンに設定します。ファイル構造とストリーム・モードを使用するデータベース・ファイルを転送する場合、サーバーはデータベース・レコードの末尾ブランクを送信しません。これが省略時です。

TRIM 2

レコード構造およびブロック・モードを含むすべての転送について、サーバーはデータベース・レコードの末尾ブランクを送信しません。

TRIM

FTP サーバーのトリム・オプションの現行の設定値を示すメッセージを表示します。

注:

1. このサブコマンドが使用可能になる前は、QSYS.LIB ファイル・システム・レコードの末尾ブランクは、サーバー・システムへファイルが転送される前に必ず除去されていました。
2. TRIM の設定は、TYPE I (バイナリー) ファイルの転送には適用されません。TYPE I ファイルの転送では、TRIM の設定に関係なく、ブランクはトリムされません。

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

STOR (ファイルの保管)

FTP サーバー・サブコマンド

サーバー・システム上でデータを保管して、既存のファイルをオーバーレイするには、次の形式の STOR FTP サーバー・サブコマンドを使用します。

```
STOR remotefile
```

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

STOU (固有の保管)

FTP サーバー・サブコマンド

既存のファイルをオーバーレイしないで、サーバー・システム上でデータを保管するには、次の STOU FTP サーバー・サブコマンドを使用します。

```
STOU remotefile
```

サーバーは固有のファイル名を生成します。ファイルに割り当てられた名前は、クライアントに送り返された応答に表示されます。

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

STRU (ファイル構造の指定)

FTP サーバー・サブコマンド

ファイルの構造を連続した一連のデータ・バイトとして指定するには、次の形式の STRU FTP クライアント・サブコマンドを使用します。

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

F

ファイル構造。ファイルの構造は、連続した一連のデータ・バイトです。

R

レコード構造。ファイルは、一連の順次レコードです。

注:

1. ファイルの構造は、転送モードと、ファイルの解釈および保存に影響します。
2. パラメーターがない場合、サーバーは、ファイル構造の現行設定値を示す応答を戻します。

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

SYST (オペレーティング・システム・システムの名前の識別)

FTP サーバー・サブコマンド

サーバー・システムのオペレーティング・システムの名前を調べるには、次の SYST FTP サーバー・サブコマンドを使用します。

```
SYST
```

戻される情報は、システムによって異なります。

iSeries サーバーには TCP/IP バージョンが組み込まれています。

サーバー応答の例を次に示します。

```
OS/400 is the remote operating system. The TCP/IP version is "V4R4M0".
```

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

TIME (FTP サーバーのタイムアウト値の設定)

FTP サーバー・サブコマンド

FTP クライアントと FTP サーバー間に FTP 制御接続が確立された後、FTP サーバーはこの接続のタイムアウトを制御します。これが非活動タイムアウト値です。

転送タイムアウトと呼ばれるデータ接続のタイムアウト値もあります。

TIME FTP サーバー・サブコマンドの形式は次のとおりです。

```
TIME inactivity [transfer]
```

inactivity

クライアントとの接続を終了するまでサーバーが待機する秒数。非活動タイムアウト値の範囲は、1 ~ 9,999,999 秒です。省略時の非活動タイムアウト値は 300 秒です。

transfer

ファイル転送タイムアウト (秒数)。このパラメーターはオプションです。このパラメーターを指定しないと、サーバーは現行値を変更しません。転送タイムアウト値の範囲は、1 ~ 9,999,999 秒です。省略時の転送タイムアウト値は 420 秒です。

たとえば、FTP サーバーの非活動タイムアウト値を 1000 秒に設定し、転送タイムアウトは現行値のままにするには、次のように入力します。

```
QUOTE TIME 1000
```

TIME サブコマンドは標準 FTP サブコマンドではありません。このコマンドは、iSeries FTP サーバーに固有のものです。

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

TYPE (表示タイプの指定)

FTP サーバー・サブコマンド

ファイル転送を行うときのファイル転送のタイプまたは表示方法を指定するには、次の形式の TYPE FTP サーバー・サブコマンドを使用します。

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

A

転送タイプとして省略時の (ASCII) 転送タイプを指定します。サーバーは、どの垂直方向の書式制御もファイルに関連付けません。サーバーが ASCII に対してサポートしているのは、省略時形式の NON PRINT のみです。ASCII 転送タイプは、両方のシステムが EBCDIC タイプを使用している場合を除き、テキスト・ファイルの転送に使用されます。

注: TYPE A の CCSID は、FTP サーバー構成属性の CCSID 値です。CHGFTPA コマンドでこれらの属性を変更できます。

B

シフト JIS 漢字 (CCSID 932)

B 1

シフト JIS 漢字 (CCSID 932)

B 2

拡張 UNIX コード漢字 (CCSID 5050)

B 3

ASCII シフトイン・エスケープ・シーケンスを使用する JIS 1983 (CCSID 5054)

B 3 A

ASCII シフトイン・エスケープ・シーケンスを使用する JIS 1983 (CCSID 5054)

B 3 R

JISROMAN シフトイン・エスケープ・シーケンスを使用する JIS 1983 (CCSID 5052)

B 4

ASCII シフトイン・エスケープ・シーケンスを使用する JIS 1978 (CCSID 5055)

B 4 A

ASCII シフトイン・エスケープ・シーケンスを使用する JIS 1978 (CCSID 5055)

B 4 R

JISROMAN シフトイン・エスケープ・シーケンスを使用する JIS 1978 (CCSID 5053)

B 5

ハングル (CCSID 934)

B 6

韓国語標準コード KSC-5601、1989 年版 (CCSID 949)

B 7

繁体字中国語 (5550) (CCSID 938)

C

転送タイプとして、システムに導入されている任意の CCSID (コード化文字セット識別コード) を指定します。CCSID 番号の前には、必ず C を付けます。

E

転送タイプとして EBCDIC を指定します。サーバーは、どの垂直方向の書式制御もファイルに関連付けません。サーバーが EBCDIC に対してサポートしているのは、省略時形式の NON PRINT のみです。EBCDIC 転送タイプは、内部文字表示として EBCDIC を使用するシステム間で効率的に転送を行うためのものです。

F

IBM EBCDIC 漢字 (CCSID 5035)

F 1

IBM EBCDIC 漢字 (CCSID 5035)

I

転送タイプとしてイメージを指定します。イメージ転送タイプでは、データは 8 ビット・バイトにパックされたビット・ストリングです。イメージ転送タイプは、ファイルの保管と検索、およびオブジェクト・コードなどの 2 進データの転送を、効率的に行います。

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

USER (サーバーへのユーザー・ログオン ID の送信)

FTP サーバー・サブコマンド

```
USER username
```

username

iSeries サーバー上のユーザー・プロファイルです。

注: USER サブコマンドが正常に実行され、iSeries サーバーがパスワード・セキュリティ対応で構成されている場合、サーバーはパスワードを要求する応答をクライアントに送信します。クライアントは、PASS サーバー・サブコマンドを使用して、パスワードをサーバーに送信します。サーバーがセキュリティ・レベル 10 で稼働している場合は、パスワード・プロンプトは表示されません。

参照:

- FTP サーバー・サブコマンド
- FTP サーバーの構文規則

FTP クライアント・サブコマンド

ファイル転送プロトコル (FTP) サブコマンドは、あるコンピューターから別のコンピューターにファイルを転送するよう、FTP クライアントに指示します。FTP クライアント・サブコマンドでは、リモート FTP サーバーとの接続を確立し、ライブラリーおよびディレクトリーをナビゲートし、ファイルの作成、削除、転送を行うことができます。

クライアント・サブコマンドとその構文の説明は、次のトピックからアクセスできます。

iSeries FTP クライアントは次のサブコマンドをサポートします。次の表は、クライアント・サブコマンドと、使用可能な省略形、および各サブコマンドの機能を示しています。

サブコマンド	用途
?	FTP の使用方法を説明します。
ACCT	ユーザーのアカウント情報をリモート・システムに送信します。
APPEND	リモート・システム上のファイルにローカル・ファイル・メンバーを追加します。
ASCII	ファイル転送のタイプを ASCII 形式に設定します。
BINARY	ファイル転送のタイプをバイナリー形式に設定します。
CD	リモート・システム上の作業ディレクトリーを変更します。
CDUP	リモート・システム上の親ディレクトリーを変更します。
CLOSE	リモート・システムとのセッションを終了します。
DEBUG	デバッグのオン / オフを切り替えます。
DEBUG	クライアントのタイムアウト値を変更します。
DELETE	リモート・システム上のファイルを削除します。
DIR	リモート・システム上のディレクトリーとファイルを表示します。
EBCDIC	ファイル転送のタイプを EBCDIC 形式に設定します。
GET	リモート・システムからローカル・システムへファイルをコピーします。
HELP	FTP クライアント・サブコマンドに関する情報を取得します。
LCD	ローカル・システム上の作業ディレクトリーを変更します。
LOCSITE	ローカル・サイト情報を指定します。

サブコマンド	用途
LOCSTAT	ローカル状況情報を表示します。
LPWD	ローカル・システム上の作業ディレクトリーを表示します。
LS	リモート・システム上のファイル・セットの中のファイルの名前をリストします。
LTYPE	ローカル・システム上のファイル転送のタイプを指定します。
MDELETE	サーバー・システム上の複数のファイルを削除します。
MGET	リモート・システムから 1 つまたは複数のファイルをコピーします。
MKDIR	ディレクトリーまたはサブディレクトリーを作成します。
MODE	ファイル転送のデータ形式を指定します。
MPUT	1 つまたは複数のローカル・ファイルをリモート・システムに送信します。
NAMEFMT	使用するファイル命名形式を指定します。
NOOP	応答を検査します。
NULLFLDS	NULL フィールドを許可します。
OPEN	FTP サーバーに接続します。
PASS	ユーザーのパスワードを送信します。
PUT	ローカル・ファイル・メンバーをリモート・システムへコピーします。
PWD	リモート・システムの現行ディレクトリーを表示します。
QUIT	FTP セッションを終了します。
QUOTE	サブコマンドを FTP サーバーへ送信します。
REINITIALIZE	リモート・システム上のセッションを再始動します。
RENAME	リモート・システム上のファイルの名前を変更します。
RESET	サーバー応答待ち行列をクリアします。
RMDIR	リモート・システム上のディレクトリーを削除します。
SECDATA	FTP サーバーとセキュアな接続を確立している場合に、データ接続に使用する保護レベルを指定します。
SECOPEN	指定されているセキュリティー・プロトコルを使用して FTP サーバーへのセキュアな制御接続をオープンします。
SENDPASV	PASV サブコマンドを送信するかどうかを指定します。
SENDPORT	PORT サブコマンドを送信するかどうかを指定します。
SENDSITE	SITE サブコマンドを送信するかどうかを指定します。
SITE	リモート・システムが使用する情報を送信します。
STATUS	リモート・システムから状況情報を取得します。
STRUCT	送信するデータのファイル構造を指定します。
SUNIQUE	ファイル置換を制御します。
SYSCMD	FTP を終了せずにローカル・システム上で CL コマンドを実行します。

サブコマンド	用途
SYSTEM	リモート・システムの OS を表示します。
TYPE	ファイル転送のタイプを指定します。
USER	リモート・システムへユーザー ID を送信します。
VERBOSE	FTP サーバー応答の表示を制御します。

関連トピック:

- FTP クライアントの構文規則
- FTP サーバー・サブコマンド
- FTP サーバーの応答状況メッセージ: 一般的な応答コードおよびその説明へのアクセス。

ACCT (アカウント情報の送信)

FTP クライアント・サブコマンド

システムによっては、特定のシステム機能を使用可能にするために、アカウント情報を必要とするものがあります。リモート・システムは、それらの情報を求めるプロンプトを表示します。アカウント情報を送信するには、次のように、アカウント (ACCT) FTP クライアント・サブコマンドを使用します。

```
ACCT account-information
```

account-information

ユーザーのアカウントを識別する文字列。アカウント情報は、ホスト・システムが特権を認可するために使用するパスワードの形を取ることがあります。このパスワードは、ユーザーのユーザー・パスワードではなく、リモート・システム上のパスワードです。

たとえば、IBM VM (仮想計算機) オペレーティング・システムの下での TCP/IP では、ミニディスクへの読み取りアクセスと書き込みアクセスのために、パスワードが必要な場合があります。現行ディレクトリーのミニディスクのためのパスワードを提供するには、ACCT サブコマンドを使用します。リモート・システムが iSeries の場合、ACCT サブコマンドは何も実行しません。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

APPEND (リモート・ファイルへのローカル・ファイル・メンバーの追加)

FTP クライアント・サブコマンド

ローカル・ファイル・メンバー、文書、または他のファイル・システムのファイルをリモート・ファイルに追加するには、次の形式の APPEND FTP クライアント・サブコマンドを使用します。

```
APpend localfile [remotefile]
```

localfile

ローカル・ファイル・メンバー、文書、または他の iSeries ファイルの名前。リモート・システム上

のディレクトリーに追加する、階層ファイル・システム (HFS) ファイルの名前。ファイル命名方法の詳細については、NAMEFMT (ファイル命名形式の選択) を参照してください。

remotefile

リモート・システム上のファイル。リモート・ファイルを入力しない場合は、FTP クライアントが省略時名を作成します。FTP による省略時名の作成方法については、『クライアント転送サブコマンドのファイル名』を参照してください。

サーバー上にリモート・ファイルが存在しない場合は、FTP サーバーが作成します。

リモート・システムへファイルを追加するには、そのリモート・システムに対する書き込み特権が必要です。ユーザーは、ACCT サブコマンドを使用して、適切なアカウント情報を指定する必要がある場合があります (ACCT (アカウント情報の送信) を参照してください)。

省略時のファイル・コピー・モードはストリームです。このモードは、MODE サブコマンドを使用して変更しなければならない場合があります。リモート・ファイルが固定レコード形式の場合は、サーバーが、リモート・ファイルのファイル形式とレコード長を保存します。ローカル・ファイル・メンバーからのレコードは、必要に応じて、切り縮められるかブランクを埋め込まれます。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

ASCII (ファイル・タイプの ASCII への変更)

FTP クライアント・サブコマンド

AScii

FTP でファイルを転送するときには、ASCII および BINARY という 2 つの基本的なファイル・タイプを使用できます。ASCII ファイルは、プレーン・テキストのファイルです。txt といった拡張子がある場合もありますし、拡張子がない場合もあります。BINARY ファイルは、プログラムまたはその他のテキスト以外のファイルです。テキスト以外のファイルは、ファイルを作成したアプリケーションのファイル形式またはアーカイブや圧縮したファイル形式で保管されています。

ASCII 転送タイプは、EBCDIC 表記をサポートしない ASCII システムとの間で、テキスト・ファイルを転送するときには使用します。ASCII は省略時の転送タイプです。サーバーは、垂直方向の書式制御をファイルに関連付けません。ASCII は、省略時形式の NON PRINT のみをサポートします。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

BINARY (転送タイプのイメージへの設定)

FTP クライアント・サブコマンド

Binary

FTP でファイル転送するときには、ASCII および BINARY という 2 つの基本的なファイル・タイプを使用できます。ASCII ファイルは、プレーン・テキストのファイルです。.txt といった拡張子がある場合もありますし、拡張子がない場合もあります。BINARY ファイルは、プログラムまたはその他のテキスト以外のファイルです。テキスト以外のファイルは、ファイルを作成したアプリケーションのファイル形式またはアーカイブや圧縮したファイル形式で保管されています。

バイナリー・データを既存の iSeries ファイルへ転送する場合、レコード長は、その既存の iSeries ファイルのレコード長となります。たとえば、新しいデータは既存のファイル・サイズに収まる必要があります。ファイルが iSeries サーバーに存在していない場合には、FTP がユーザーに代わってレコード長を選択します。

保管ファイルなどの特定のファイルには、バイナリー・イメージ転送が必要です。TYPE をバイナリーに設定せずにその種のファイルを転送しようとする、バイナリーにする必要があることを示すメッセージが表示されます。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

CD (作業ディレクトリーまたはライブラリーの変更)

FTP クライアント・サブコマンド

リモート・システム上の作業ディレクトリー、ライブラリー、またはファイル・グループを変更するには、次のディレクトリー変更 (CD) FTP クライアント・サブコマンドを使用します。

CD <i>directory</i>

directory

リモート・システム上のファイル・ディレクトリー、ライブラリー、またはシステムに依存するその他のファイル・グループ指定機能の名前。

リモート・システムが iSeries の場合、このサブコマンドは現行のライブラリーまたはディレクトリーを変更します。リモート・システム上にどのようなディレクトリーがあるか調べるには、ディレクトリー (DIR) サブコマンドを使用してリストを表示します。

DIR サブコマンドを使用する際は、注意が必要です。詳細は、DIR (ディレクトリー項目、ライブラリー、またはファイルのリスト) を参照してください。

注: サブコマンド CD (または LCD) を使用して、ある iSeries ファイル・システムから別のファイル・システムへ変更するには、変更後の現行ディレクトリーが含まれるファイル・システムのルート・ディレクトリーを指定する必要があります。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

CLOSE (リモート・システムとの FTP セッションの終了)

FTP クライアント・サブコマンド

ローカル iSeries 上で FTP を活動状態にしたまま、リモート・システムとのセッションを終了するには、次の CLOSE FTP クライアント・サブコマンドを使用します。

```
CLOSE
```

CLOSE サブコマンドを使用すると、FTP 環境にとどまったまま、別のシステム上に別の FTP セッションをオープンすることができます。OPEN サブコマンドを使用して、同じリモート・システムまたは別のリモート・システムとの新しい接続を確立します。QUIT サブコマンドを使用して FTP サービスを終了し、FTP を開始したときの iSeries 環境へ戻ります。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

DEBUG (クライアント・トレースの作成およびリモート・システムへ送信されたサーバー・サブコマンドの表示の制御)

注: IBM にソフトウェアの問題を報告する場合には、FTP クライアント・トレースのみを使用してください。この機能によって、システム・パフォーマンスに悪い影響が出ることがあります。

この機能は、OS/400 のリリース V4R4 およびそれ以降のリリースで使用可能です。

FTP クライアント・サブコマンド

FTP クライアント・トレースを作成または表示するには、DEBUG FTP クライアント・サブコマンドを使用します。DEBUG サブコマンドは、デバッグ・モードを切り替えます。クライアントがオプションのデバッグ値を指定している場合、クライアントはその値を使用してデバッグ・レベルを設定します。デバッグがオンに設定されている場合、クライアントは文字列「>>>」を表示します。FTP クライアント・トレースを作成するには、デバッグ値を 100 に設定する必要があります。

```
DEBbug [debug value]
```

debug value

DEBUG 値が 0 の場合、DEBUG 機能はオフです。DEBUG 値が正の整数の場合、DEBUG 機能はオンです。値を指定しない場合、デバッグ値はゼロから 1 に切り替えられるか、または正の整数からゼロに切り替えられます。

100

FTP クライアント・トレースを開始します。クライアントは、DEBUG がオフになるまで、またはサーバーが FTP クライアントを終了させるまで、トレースを実行し続けます。サーバーがトレースを終了させる際に、トレース・データをフォーマットする間しばらく時間がかかる場合があります。

FTP クライアントの起動時に、即時にトレースを開始するには、次のコマンドを使用して、QTEMP ライブラリー内に QTMFTPD100 データ域を作成する必要があります。

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

QTMFTPD100 データ域が存在する場合は、DEBUG 値が 100 に設定され、FTP クライアント・トレースが開始されます。この機能の目的は、DEBUG 100 サブコマンドを発行しても FTP クライアント・トレースを開始できない場合に、FTP クライアントのデバッグ・トレースを使用可能にすることです。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

DEBUG (クライアント・タイムアウト制限値の変更)

FTP クライアント・サブコマンド

省略時のタイムアウト値がデータ転送を正常に行うのに十分でない場合、DEBUG サブコマンドを使用してクライアント・タイムアウト制限値を変更します。ネットワーク通信量または他の条件のために転送時間が非常に長くなる場合に限り、これらの値を変更してください。

FTP クライアント・タイムアウト値を変更するには、次の DEBUG FTP クライアント・サブコマンドを使用します。

```
DEBUg T1 | T2 [ value ]
```

T1

サーバー応答の読み取りのための FTP クライアント・タイムアウト制限を変更または表示します。FTP クライアントがこの制限時間内に予期されたサーバー応答を受け取らない場合、クライアントはサーバーとの制御接続をクローズします。

T2

データ転送のための FTP クライアント・タイムアウト限度を変更または表示します。FTP クライアントがこの制限時間内に予期されたデータ接続応答を受け取らない場合、クライアントはサーバーとのデータ接続をクローズします。

value

タイムアウト制限値 (秒数)。この値はゼロより大きい正数でなければなりません。この値を省略すると、クライアントは現行のタイムアウト制限値を表示します。

たとえば、次のようになります。

```
DEBUG T1 900
```

この値は、サーバー応答に対するクライアントのタイムアウト値を 900 秒に設定します。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

DELETE (リモート・システムのファイルの削除)

FTP クライアント・サブコマンド

リモート・システム上のファイル・メンバーまたはデータベース・ファイル・メンバーを削除するには、DELETE サブコマンドを使用します。DELETE FTP クライアント・サブコマンドの形式は次のとおりです。

DELEte remotefile

remotefile

リモート・システム上の削除したいファイル。リモート・システムが iSeries サーバーの場合のファイルの指定方法については、NAMEFMT (ファイル命名形式の選択) を参照してください。

リモート・システムから、ファイル削除権限について尋ねられることがあります。ACCT (アカウント情報の送信) サブコマンドを使用して、その要求に回答してください。

参照:

- MDELETE (リモート・システムの複数ファイルの削除)
- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

DIR (ディレクトリー項目、ライブラリー、またはファイルのリスト)

FTP クライアント・サブコマンド

DIR FTP クライアント・サブコマンドは、ライブラリーとその内容、またはリモート・システムのディレクトリーとディレクトリー項目のリストを表示します。ディレクトリー (DIR) サブコマンドは次の形式で使用します。

Dir [name] [(Disk)]

name

ディレクトリーまたはライブラリーの名前。省略時は、現行ディレクトリーまたは現行ライブラリーです (この場合、ディレクトリーまたはライブラリーの全体がリストされます)。現行のライブラリーまたはディレクトリーを変更するには、作業ディレクトリー変更 (CD) サブコマンドを使用します。1 セットのリモート・ファイルを指定する方法は、システムによって異なります。ほとんどのシステムでは、総称アスタリスク * が使用できます。リモート・システムが iSeries の場合、たとえば、DIR MYLIB/MYFILE.* と指定すると、ライブラリー MYLIB 中の MYFILE のすべてのメンバーがリストされます。

使用できるファイル命名形式は 2 つあります。ここでの例は NAMEFMT 0 を使用しています。FTP ファイルの命名方法の詳細については、NAMEFMT (ファイル命名形式の選択) を参照してください。

(Disk

DIR サブコマンドの結果を画面に表示する代わりに、*CURLIB/DIROUTPUT.DIROUTPUT というファイルに保管します。

リモート・システムが iSeries の場合、次の情報が含まれます。

- データベース・ファイルの場合は、*FILE オブジェクトとメンバー
- 階層ファイル・システム (HFS) のファイルの場合
 - すべての文書ライブラリー・サービス (QDLS) のフォルダーとその内容 (他のフォルダーまたは文書の場合もあります。)

- すべての光ディスク・ボリューム (QOPT) とその内容 (ディレクトリーまたはファイルの場合もあります。)

DIR サブコマンドを使用する際は、注意が必要です。パラメーターを指定せずに DIR サブコマンドを入力した場合、サーバーは、現行ディレクトリーにあるすべてのファイルのリストを生成します。このリストは予想以上に長くなることがあります。

ディレクトリーの中のファイルの名前をリストするには、リスト (LS) サブコマンドを使用します (LS (リモート・ファイル名のリスト) を参照)。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

EBCDIC (ファイル・タイプの EBCDIC への変更)

FTP クライアント・サブコマンド

```
EbCdic
```

EBCDIC 転送タイプは、別の EBCDIC システムとの間でファイルを転送する場合に便利です。それは、どちらのシステムでも、ASCII と EBCDIC 間の変換が不要なためです。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

GET (リモート・システムからローカル・システムへのファイルのコピー)

FTP クライアント・サブコマンド

```
Get remotefile [localfile]  
[(Replace)]
```

remotefile

リモート・システムから検索したいファイル。

localfile

作成したいローカル・ファイル・メンバー、文書、またはその他のファイル。ローカル・ファイル名を指定しない場合、FTP クライアントにより省略時名が提供されます。省略時名については、『クライアント転送サブコマンドのファイル名』を参照してください。

(Replace

localfile がすでに存在する場合は、それを上書きします。(Replace を指定しないと、サーバーは localfile を上書きしません。

GET サブコマンドで使用するファイル命名形式は、ファイルの存在するファイル・システムによって決まります。

- ファイルがライブラリー・ファイル・システム (QSYS.LIB) に存在しない場合、次の名前形式 (NAMEFMT) 1 で GET サブコマンドを使用する必要があります。

```
GET /QDLS/QIWSOS2/PCSMENU.EXE
```

- ファイルがライブラリー・ファイル・システムに存在しない場合、GET サブコマンドを使用して、名前形式 (NAMEFMT) を 0 に設定します。

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

リモート・サーバーが iSeries の場合、このコマンドは、YOURLIB の中の YOURFILE の YOURMBR を取得し、ローカル・システム上の現行ディレクトリーの中の YOURFILE の YOURMBR にコピーします。現行ディレクトリーの定義の詳細については、『LCD (ローカル・システムの作業ライブラリーまたは作業ディレクトリーの変更)』を参照してください。

- 注:** リモート・ファイル名にアポストロフィを使用する必要がある場合は、そのファイル名をさらに 2 組のアポストロフィで囲みます。次の例では、リモート・ホストから 'MEMBER.ONE' が取得されます。詳細は、『サブコマンド・パラメーターの囲み方』を参照してください。

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

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則
- MGET (リモート・システムからローカル・システムへの複数ファイルのコピー)
- PUT (ローカル・システムのファイル・メンバーをリモート・システムのファイルへコピー)
- MPUT (ローカル・システムからリモート・システムへの複数ファイル・メンバーの送信)

HELP (FTP サブコマンドのヘルプの取得)

HELP サブコマンドは、ローカル・システムとリモート・システムで使用する FTP サブコマンドについての情報を提供します。

FTP クライアント・サブコマンドのヘルプ

ローカル・システムで使用する FTP サブコマンドについての情報を取得するには、次の形式の HELP サブコマンドを使用します。

```
Help [* | ALL | subcommand ]
```

* または ALL

FTP クライアント・サブコマンドのリストを表示します。

subcommand

指定したクライアント・サブコマンドの詳細なヘルプを提供します。たとえば、HELP GET は、リモート・システムからローカル・システムへファイルを転送する方法を示します。サブコマンドは、意味を失わない程度に省略した形で入力できます。

パラメーターなしで HELP サブコマンドを使用した場合は、サブコマンドのリストと、使用可能なヘルプ情報の一般的な説明が表示されます。ヘルプ画面でコマンドの上にカーソルを置き、**Enter** キーを押すと、そのコマンドについての適切なヘルプが表示されます。

iSeries サーバーでローカル・サブコマンドのリストを表示するには、次のように入力します。

HELP

ヘルプ情報は、? サブコマンドを使用しても取得できます。

FTP サーバー・サブコマンドのヘルプ

リモート・システムの FTP サブコマンドについてのヘルプを取得するには、次の形式の HELP サブコマンドを使用します。

```
Help SERVER [subcommand]
```

SERVER

FTP サーバー・サブコマンドについてリモート・システムが提供しているヘルプが表示されます。これは、HELP パラメーターを付けて QUOTE を使用するのと似ています。QUOTE HELP は、リモート・システムがサポートする FTP サブコマンドをリストします。

subcommand

情報が必要なサーバー・サブコマンドの名前。たとえば、HELP SERVER STOR は、STOR サブコマンドについてのヘルプをサーバーに要求します。

注: RHELP は HELP SERVER と同義です。たとえば、HELP SERVER SITE と RHELP SITE は同等です。

詳細については、QUOTE (FTP サーバーへのサブコマンドの送信) を参照してください。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

LCD (ローカル・システムの作業ライブラリーまたは作業ディレクトリーの変更)

FTP クライアント・サブコマンド

```
LCd pathname
```

pathname

ローカル・システム上のライブラリー、フォルダー、またはディレクトリーの名前。

- 注:
1. LCD サブコマンドは、ライブラリー・リストの現行ライブラリー・エントリーを変更しません。
 2. サブコマンド CD (または LCD) を使用して、あるファイル・システムから別のファイル・システムへ変更するには、“root” ディレクトリー (たとえば、/QDLS または /QOPT) を指定する必要があります。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

LOCSITE (ローカル・サイト情報の指定)

FTP クライアント・サブコマンド

クライアント・システムに固有のサービスを提供するために FTP クライアントが使用する情報を指定するには、次の LOCSITE FTP クライアント・サブコマンドを使用します。

```
LOCSITE [ parameters ]
```

iSeries FTP クライアントがサポートする LOCSITE サブコマンドのパラメーターとパラメーター・オプションは次のとおりです。

CRTCCSID *CALC

ASCII ファイル転送時に作成された新しいデータベース・ファイルは、ASCII ファイル転送 CCSID の関連省略時 EBCDIC CCSID を使用します。これが省略時値です。

CRTCCSID *USER

ASCII ファイルの転送時に作成された新しいデータベース・ファイルは、現行ジョブ CCSID を使用します。ただし、この CCSID が 65535 の場合は、現行ジョブに対して指定された言語 ID で決定される、省略時の CCSID を使用します。

CRTCCSID *SYSVAL

ASCII ファイルの転送時に作成された新しいデータベース・ファイルは、QCCSID システム値で指定された CCSID を使用します。

CRTCCSID [CCSID-number]

ASCII ファイル転送時にクライアント上でデータベース・ファイルを作成する際に使用する CCSID を指定します。この値はサーバーによって妥当性検査されます。

CRTCCSID

現行の FTP クライアント CRTCCSID 設定値が入ったメッセージを表示します。

TRIM 0

トリム・オプションをオフに設定します。サーバーは、データベース・レコードの末尾ブランクを送信します。

TRIM 1

トリム・オプションをオンに設定します。ファイル構造とストリーム・モードを使用するデータベース・ファイルを転送する場合、サーバーはデータベース・レコードの末尾ブランクを送信しません。これが省略時です。

TRIM 2

レコード構造およびブロック・モードを含むすべての転送について、サーバーがデータベース・レコードの末尾ブランクを送信ないように、トリム・オプションを設定します。

TRIM

FTP クライアントの TRIM オプションの現行の設定値が入ったメッセージを表示します。

注:

1. このサブコマンドが使用可能になる前は、QSYS.LIB ファイル・システム・レコードの末尾ブランクは、サーバー・システムへファイルが転送される前に必ず除去されていました。
2. TRIM の設定は、TYPE I (バイナリー) ファイルの転送には適用されません。TYPE I ファイルの転送では、TRIM の設定に関係なく、ブランクはトリムされません。

▶ DTAPROT C

データ保護変数を C (消去) に設定します。この変数は、セキュアな制御接続をオープンするときにデータ保護レベルを設定するために使用します。データ保護セキュリティの設定の詳細は、コマンド SECDATA および SECOPEN を参照してください。

DTAPROT P

データ保護変数を P (プライベート) に設定します。この変数は、セキュアな制御接続をオープンするときにデータ保護レベルを設定するために使用します。

DTAPROT

データ保護変数の現行値が入ったメッセージを表示します。 ◀

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

LOCSTAT (ローカル状況情報の表示)

FTP クライアント・サブコマンド

```
LOCSTat
```

以下のようなローカル状況情報を表示します。

- SENDSITE サブコマンドの現行設定値
- SENDPORT サブコマンドの現行設定値
- リモート・システム名、ポート番号、およびログオン状況
- データ・タイプと転送モード
- クライアントとサーバーの名前形式の値
- VERBOSE モードの設定値
- DEBUG モードの設定値

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

LS (リモート・ファイル名のリスト)

FTP クライアント・サブコマンド

```
LS [name] [(Disk)]
```

name

リストしたいリモート・ディレクトリー、リモート・ファイル、またはリモート・ライブラリー。リモート・システムが iSeries サーバーの場合、サーバーはファイル名とファイルのメンバーをリストします。省略時では、現行ディレクトリー、現行ライブラリー、または現行フォルダーの全体がリストされます。現行ディレクトリー、現行ライブラリー、または現行フォルダーを変更するには、CD サブコマンド (CD (作業ディレクトリーまたはライブラリーの変更) を参照してください) を使用します。リモート・ファイルの様子はシステムによって異なります。

(Disk

LS サブコマンドの結果を、画面に表示する代わりに、*CURLIB/LSOUTPUT.LSOUTPUT ファイルに保管します。同じ *CURLIB に (Disk パラメーターを指定するたびに、サーバーは LSOUTPUT.LSOUTPUT メンバー・ファイルの内容を変更します。

注: FTP サーバーが否定の応答コード (550) を戻す場合、LSOUTPUT メンバーは作成されません。FTP サーバーがファイル名の指定されていない肯定の応答コード (150) を戻す場合、レコードのない LSOUTPUT メンバーが作成されます。

LS サブコマンドはファイル名だけをリストします。ファイルに関する補足情報を含む完全なディレクトリー項目のリストを必要とする場合は、DIR (ディレクトリー項目、ライブラリー、またはファイルのリスト) を参照してください。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

LTYPE (ローカル・タイプ)**FTP クライアント・サブコマンド**

ローカル・システムでファイル転送を行うときのファイル転送のタイプまたは表示方法を指定するには、次の形式の LTYPE FTP クライアント・サブコマンドを使用します。

LType C ccsid#

C

CCSID タイプ。この値を C としてコード化します。

ccsid#

CCSID 値。この値を CCSID 番号 1-65533 としてコード化します。

注: LTYPE サブコマンドは TYPE サブコマンドに似ています (TYPE (ファイル転送タイプの指定) を参照してください)。LTYPE サブコマンドは、クライアント側の表示タイプだけを変更します。TYPE サブコマンドは、クライアント側とサーバー側の両方の表示タイプを変更します。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

MDELETE (リモート・システムの複数ファイルの削除)

FTP クライアント・サブコマンド

```
MDelete {remotefile [remotefile...]}
```

remotefile

そのサーバー・システムにある、1 つまたは複数の削除したいファイル。

注: リモート・ファイルが iSeries の QSYS.LIB ファイルの場合、サーバーはこの iSeries 物理ファイルのすべてのメンバーを削除します。そのファイル自身は残されます。

次は、NAMEFMT 0 のライブラリー・ファイル・システムの例です。

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

この例では、iSeries サーバーのライブラリー MYLIB のファイル FILE1 のメンバー MBRA と、ライブラリー YOURLIB のファイル FILE2 のメンバー MBRB が削除されます。次は、NAMEFMT 1 の場合の同じ条件の例です。

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

次は、NAMEFMT 1 の文書ライブラリー・システムの例です。

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

この例では、文書ライブラリー・サービス・ライブラリーの中のフォルダー QIWSOS2 から、文書 PCSMENU.EXE が削除され、さらに、iSeries 上の QDLS ライブラリーの中のフォルダー PCSDIR から、PCSFIL.EXE が削除されます。

アスタリスク (*) を使用すると、削除するファイルを総称的に指定できます。たとえば、リモート・システムが iSeries の場合、NAMEFMT 0 では次のように入力します。

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

この例では、ライブラリー MYLIB のファイル MYFILE のすべてのメンバーが削除されます。アスタリスクの使用は、ストリングの終わりでのみ有効です。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

MGET (リモート・システムからローカル・システムへの複数ファイルのコピー)

FTP クライアント・サブコマンド

MGET によるファイル転送方法:

1 つまたは複数のリモート・ファイルをコピーするには、次の形式の MGET FTP クライアント・サブコマンドを使用します。転送するリモート・ファイルごとに、GET サブコマンドが個別に実行されます。対応するローカル・ファイルの名前は、省略時の命名規則に従って、サーバーが自動的に作成します。

MGET FTP クライアント・サブコマンドは、次の処理により、ファイルの配置場所を決定します。

- MGET サブコマンドは、常に、現行ライブラリーまたは現行ディレクトリーにファイルを配置します。

- ユーザーが LCD サブコマンドを発行した場合、サーバーはこのライブラリーまたはディレクトリーを使用します。
- ユーザーが LCD サブコマンドを発行しなかった場合、サーバーは現行ディレクトリーを次のように設定します。
 - ユーザーのジョブが現行ライブラリーを設定している場合、このライブラリーが FTP の現行ディレクトリーとなります。
 - ユーザーのジョブに現行のライブラリー・セットがない場合、サーバーは QGPL を現行ディレクトリーとして使用します。

```
MGet {remotefile
[remotefile...]}[(Replace]
```

remotefile

リモート・システムから検索したい 1 つまたは複数のファイル。

(Replace

ローカル・システム上の既存ファイルを上書きします。 Replace オプションを使用しない場合は、ローカル・システム上に同じ名前のファイルが存在していても、その既存ファイルは上書きされません。リモート・ファイルがコピーされるローカル・ファイルの名前は、自動的に作成されます。詳細については、GET (リモート・システムからローカル・システムへのファイルのコピー) の GET サブコマンドの解説を参照してください。

アスタリスク (*) を使用すると、あるファイルのすべてのメンバーを、現行ライブラリーまたは現行ディレクトリーにコピーできます。たとえば、リモート・システムが iSeries の場合、

- MGET MYLIB/MYFILE.* は、リモート・システムのライブラリー MYLIB のファイル MYFILE のすべてのメンバーを、ローカル・システム上の現行ライブラリーへコピーします。
- MGET /QSYS.LIB/MYLIB.LIB/MYFILE.FILE/*.MBR は、NAMEFMT 1 の場合のこのコマンドの例です。
- MGET /QOPT/PICTURES/IMAGES/*. * は、光ディスク・ボリューム PICTURES からディレクトリー IMAGES のすべてのファイルを、ローカル・システム上の現行ライブラリー (または現行ディレクトリー) へコピーします。
- MGET TESTFILE.A* は、ファイル TESTFILE の中の文字 A で始まるすべてのメンバーをコピーします。
- MGET /QDLS/QISSOS2/A* は、フォルダー QISSOS2 の中の文字 A で始まるすべての文書をコピーします。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

MKDIR (ディレクトリーの作成)

FTP クライアント・サブコマンド

```
MKdir pathname
```

pathname

リモート・システム上のファイル・ディレクトリー、ライブラリー、またはシステムに依存するその他のファイル・グループ指定機能の名前。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

MODE (データの伝送モードの指定)

FTP クライアント・サブコマンド

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

B

ブロック・モードを指定します。このモードでは、サーバーは、1 つまたは複数の見出しバイトが先行する一連のデータ・ブロックとして、データを伝送します。データをブロック・モードで転送する場合、そのタイプは EBCDIC でなければなりません。

S

ストリーム・モードを指定します。このモードでは、サーバーはバイトのストリームとしてデータを伝送します。ストリーム・モードでは、すべての表示タイプを使用できます。

注:

1. ストリーム・モードは FTP で使用される省略時の転送モードです。システムによっては、ブロック・モードをサポートしていないものもあります。
2. このオプション・パラメーターを省略すると、クライアントは現在の MODE 値を表示します。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

MPUT (ローカル・システムからリモート・システムへの複数ファイル・メンバーの送信)

FTP クライアント・サブコマンド

1 つまたは複数のローカル・ファイルをリモート・システムへ送信するには、MPUT FTP クライアント・サブコマンドを使用します。クライアントは、転送する各ローカル・ファイルごとに、別個の PUT サブコマンドを実行します。対応するリモート・ファイルの名前は、省略時の命名規則に従って作成されます。

```
Mput {localfile [localfile...]}
```

localfile

リモート・システムへ転送したい、ローカル・ライブラリー・ファイル・システムのファイルのメンバー、または FTP がサポートする他のファイル・システムのファイルを、1 つまたは複数指定します。リモート・システム上でファイルに与えられる名前は、クライアントによって自動的に生成されます。

注: リモート・ファイルがすでに存在する場合、その内容は *localfile* の内容で置き換えられます。ただし、固有保管 (SUNIQUE) がオンの場合は置き換えられません (SUNIQUE (ファイルの上書きの制御) を参照)。

リモート・ファイルが iSeries の場合のファイルの指定方法については、NAMEFMT (ファイル命名形式の選択) を参照してください。次は、NAMEFMT 0 の場合の例です。

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

この例では、ライブラリー MYLIB のファイル FILE1 から、メンバー MBR1 および MBR2 がリモート・システムへ送信されます。

次は、NAMEFMT 1 の場合の例です。

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

この例では、フォルダー QIWSOS2 から、文書 PCSMENU.EXE および PCSMENU2.EXE がリモート・システムへ送信されます。

アスタリスク (*) を使用すると、あるファイルのすべてのメンバーを送信できます。たとえば、MPUT MYLIB/MYFILE.* では、ライブラリー MYLIB のファイル MYFILE のすべてのメンバーが転送されます。詳細は、『MGET (リモート・システムからローカル・システムへの複数ファイルのコピー)』を参照してください。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則
- PUT (ローカル・システムのファイル・メンバーをリモート・システムのファイルへコピー)

NAMEFMT (ファイル命名形式の選択)

FTP クライアント・サブコマンド

ローカル・システムとリモート・システム (iSeries の場合) で使用するファイル命名形式を選択するには、次の NAMEFMT FTP クライアント・サブコマンドを使用します。

```
Namefmt [ 0 | 1 ]
```

0

ライブラリー・ファイル・システムのデータベース・ファイル専用の命名形式。一般的な形式は次のとおりです。

```
[libname/]filename[.mbrname]
```

1

FTP がサポートするすべてのファイル・システムの命名形式 (ライブラリー・ファイル・システムを含む)。すべての iSeries ファイル・システムで動作するようにするには、命名形式を「1」に設定する必要があります。

次は、この命名形式のライブラリー・ファイル・システムのファイルです。

```
[/QSYS.LIB/][libname.LIB/]filename.FILE[/mbrname.MBR]
```

保管ファイルには、次の形式も使用できます。

/QSYS.LIB/libname.LIB/filename.SAVF

文書ライブラリー・サービスのファイル・システムのファイルは、次の形式で命名されます。

[/QDLS/][{foldername[.ext]/}]filename[.ext]

光ファイル・システムの場合は、形式は次のようになります。

/QOPT/volname/dirname/filename.ext

- 注:
1. 作業ディレクトリーがデータベース・ライブラリーの場合のみ、命名形式を 0 に設定できます。
 2. パラメーターなしで NAMEFMT サブコマンドを指定すると、クライアントは現行の命名形式を表示します。

以下も参照してください。

- ファイル・システムと命名規則
- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

NULLFLDS (NULL フィールドのあるファイルの転送の許可)

FTP クライアント・サブコマンド

次のコマンドを使用して、ローカル・システムおよびリモート・システム (iSeries の場合) で NULL フィールド値を含むデータベース・ファイルの転送を許可するかどうかを選択します。

```
Nullflds [ 0 | 1 ]
```

パラメーターを入力する場合、有効な値は次のとおりです。

0

NULL フィールドを含むデータベース・ファイルの転送を許可しません。これが省略時です。

1

NULL フィールドを含むデータベース・ファイルの転送を許可します。

- 注:
1. NULL フィールドを含むファイルを転送するには、クライアントとサーバーの両方でこの設定を使用可能にしておく必要があります。ターゲット・ファイルはファイル転送前に存在していなければなりません。また、ターゲット・ファイルはソース・ファイルと同じファイル定義を持っていなければなりません。
 2. NULL フィールドを含むファイルを iSeries 以外のサーバーに転送する場合、またはデータのコード・ページ変換が行われる転送タイプの場合、結果は予測できません。
 3. パラメーターなしで NULLFLDS サブコマンドを指定すると、クライアントは現行の設定を表示します。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

OPEN (リモート・システムの FTP サーバーへの接続)

FTP クライアント・サブコマンド

```
Open systemname [portnumber]
```

systemname

リモート・システムの名前または IP アドレス。

portnumber

サーバーが接続を終了するまで、このセッションに使用するポート番号。これはオプションです。ポート番号を指定しない場合は、サーバーが 1 を選択します。

いったんリモート・システムへの接続をオープンすると、現行セッションをクローズしないかぎり、別のシステムへ接続することはできません。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

PASS (パスワードの送信)

FTP クライアント・サブコマンド

```
PAss password
```

password

パスワードを指定する文字列。

このサブコマンドより前に OPEN サブコマンドおよび USER サブコマンドを指定する必要があります。システムによっては、アクセス制御のためのユーザー識別にはこれで十分です。サーバーに接続またはログオンする際にサーバーがパスワードの入力を要求した場合は、このサブコマンドは必要ありません。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

PUT (ローカル・システムのファイル・メンバーをリモート・システムのファイルへコピー)

FTP クライアント・サブコマンド

```
PUt localfile [remotefile]
```

localfile

ローカル・ライブラリー・システム・ファイルのメンバー、保管ファイル、文書、またはその他のファイルの名前。

remotefile

送達されたファイルのリモート・システムでの名前。リモート・ファイル名を指定しない場合は、FTP サーバーによって省略時名が提供されます。省略時名については、『クライアント転送サブコマンドのファイル名』を参照してください。同じ名前のリモート・ファイルがすでに存在する場合、固有保管 (SUNIQUE) がオンになっていない限り、サーバーはその内容をローカル・ファイルの内容で置き換えます。(SUNIQUE (ファイルの上書きの制御) を参照)。

リモート・システムへファイルを送信するには、定義済みの現行作業ディレクトリーと、それに対する書き込み特権がなければなりません。

次の例では、PUT サブコマンドを使用してファイル・メンバーを転送しています。

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

この例では、ライブラリー MYLIB のファイル MYFILE のメンバー MYMBR が、リモート・システムへ送信されます。

次の例では、文書ライブラリー・サービス・ファイル・システムのフォルダー QIWSOS2 から文書 PCSMENU.EXE がリモート・システムへ送信されます。

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

注: リモート・ファイル名にアポストロフィを使用する必要がある場合は、そのファイル名をさらに 2 組のアポストロフィで囲まなければなりません。次の例では、'MEMBER.ONE' が、ファイル名としてリモート・ホストへ送信されます。詳細については、『サブコマンド・パラメーターの囲み方』を参照してください。

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

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則
- MPUT (ローカル・システムからリモート・システムへの複数ファイル・メンバーの送信)
- GET (リモート・システムからローカル・システムへのファイルのコピー)
- MGET (リモート・システムからローカル・システムへの複数ファイルのコピー)

PWD (現行ディレクトリー、現行フォルダー、または現行ライブラリーの表示)

FTP クライアント・サブコマンド

リモート・システムの現行ディレクトリーまたは現行ライブラリーを表示するには、次の PWD FTP サブコマンドを使用します。

```
PWd
```

リモート・サーバーが iSeries の場合、サーバーは、リモート・システム上の現行ライブラリーまたはファイル・システム・ディレクトリーを表示します。また、サーバーは作業ディレクトリーも、引用符で囲んで表示します。リモート・システムの現行ライブラリーまたは現行ディレクトリーを変更するには、作業ディレクトリー変更 (CD) サブコマンドを使用します。

参照:

- FTP クライアント・サブコマンド

- FTP クライアントの構文規則

QUOTE (FTP サーバーへのサブコマンドの送信)

FTP クライアント・サブコマンド

```
QUOte string
```

string

リモート FTP サーバーへ送信してそこで解釈されるサーバー・サブコマンド。FTP サーバーは、文字列 `verbatim` をリモート FTP サーバーに送信します。

注:

1. クライアントが、特殊な iSeries FTP サーバーのサブコマンド `RCMD` (FTP サーバー・システムへの `CL` コマンドの送信) を実行するには、`QUOTE` サブコマンドが必要です。たとえば、サーバー・ジョブ・ログをスプール・ファイルに書き込む場合、次のように入力します。

```
QUOTE RCMD DSPJOBLOG
```

ジョブ・ログにアクセスするには、`WRKSPLF` を使用できます。`WRKSPLF` を異なるユーザー・プロファイルから実行する場合には、FTP サーバーにログインしたユーザーのユーザー・プロファイルを指定する必要があることに注意してください。

2. iSeries FTP サーバーでは、ストリングは 1000 文字に制限されています。
3. `QUOTE` サブコマンドは、入力されたものをすべてサーバーに渡します。たとえば、

```
QUOTE CWD 'SYS1'
```

サーバーは次を受け取ります。

```
CWD 'SYS1'
```

サーバーからのヘルプを表示するには、次のとおり入力します。

```
QUOTE HELP
```

サーバーは `HELP` サブコマンドをリモート・ホストへ送信し、リモート・ホストはサポートするすべてのサブコマンドの表示を戻します。表示される情報は、リモート・ホストのタイプによって異なります。

`QUOTE` サブコマンドと共に入力したサーバー・サブコマンドは、サーバーのみに影響しますが、同様のクライアント・サブコマンドはクライアントとサーバーの両方に影響することがあるため、注意が必要です。たとえば、`REIN` クライアント・サブコマンドは、サーバーへ `REIN` サーバー・サブコマンドを送信し、特定のクライアント状態変数を再初期化します。`QUOTE REIN` は、`REIN` のみをサーバーに送信し、クライアント状態変数は一切変更しません。

注:

`QUOTE` サブコマンドを使用してサーバー・サブコマンドを直接入力するときは、意図と異なる結果にならないよう、注意が必要です。通常、`QUOTE` サブコマンドは、他のクライアント・サブコマンドを使用できないような、特殊な状況に使用します。`CRTL` のような特殊な iSeries サーバーのサブコマンドの 1 つを使用する必要がある場合が、その一例です。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

REINITIALIZE (システム間のセッションの再初期化)

FTP クライアント・サブコマンド

```
REInitialize
```

サーバーが REINITIALIZE サブコマンドをサポートしている場合、そのサーバーとの USER セッションは終了します。サーバーは、接続が確立されたときと同じ状態になります。ユーザーは、継続するために再度ログオンする必要があります。

既に進行中のファイル転送は、USER セッションを終了する前にすべて完了させることができます。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

RENAME (リモート・システムのファイルの名前変更)

FTP クライアント・サブコマンド

```
REname originalname newname
```

originalname

リモート・ファイルの現在の名前。

newname

リモート・ファイルの新しい名前。 *newname* で指定したファイルがすでに存在する場合、新しいファイルで置き換えられます。

次の例では、光ディスク・ボリューム PICTURES 上のディレクトリー IMAGES にあるファイル SPORTSCAR.BMP が、CAR.BMP に名前変更されます。

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

注: iSeries サーバーでは、異なるファイル・システムへファイルを名前変更することはできません。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

RESET (リセット)

FTP クライアント・サブコマンド

サーバー応答待ち行列をクリアするには、次の RESET FTP クライアント・サブコマンドを使用します。

```
REset
```

このサブコマンドは、サーバー・サブコマンドと応答の順序付けを、リモート FTP サーバーと再同期させます。再同期は、リモート・サーバーによる FTP プロトコル違反が発生したときに、必要になることがあります。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

RMDIR (ディレクトリーの除去)

FTP クライアント・サブコマンド

```
RMdir pathname
```

pathname

リモート・システム上のファイル・ディレクトリー、ライブラリー、またはシステムに依存するその他のファイル・グループ指定機能の名前。階層ファイル・システム (HFS) ディレクトリーの場合、削除できるのは空のディレクトリーのみです。サーバーは無条件にライブラリーを削除します。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

SECData (データ・セキュリティ保護の設定)

FTP クライアント・サブコマンド

リモート・システムとのセキュアな制御接続を確立済みの場合に、データ接続に使用する保護レベルを指定するには、SECData サブコマンドを以下のように使用します。

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

注: SData はこのサブコマンドの同義語です。

C

データ・チャネル保護レベルは「clear」に設定されます。この接続は、セキュアではありません。この接続は、事前に暗号化されたデータまたは機密ではないデータに使用することができます。

P

データ・チャネル保護レベルは「private」に設定されます。この接続はセキュアです。クライアントとサーバー間の TLS 折衝は、データが接続を通して送信される前に行われる必要があります。

1. パラメーターが指定されていない場合、SECData はデータ・セキュリティ保護の設定に現在使用中の値を表示します。
2. FTP サーバーとのセキュアな制御接続が確立されたときには、データ保護レベルは、まず STRTCPFTP CL コマンドの DTAPROT パラメーターで指定された値に設定されます。
3. セキュアな制御接続は、SECData サブコマンドを使用する際に必要です。

4. SECData サブコマンドによって正常にデータ保護レベルが設定されるときに、PROT サーバー・サブコマンドはサーバーに対して実行されます。
5. SECData サブコマンドは、データ保護レベルを設定するときに PBSZ サブコマンドと PROT サブコマンドをサーバーに送信します。また、SECData サブコマンドは正常に実行された PROT サブコマンドそれぞれに対して、クライアント変数を設定します。この変数は、サーバーが受け入れた最新のデータ保護レベル (C または P) を表します。この変数は、SECOpen サブコマンドがセキュアな制御接続をオープンするときに、データ保護レベルを設定するために使用されます。この変数は、LOCSITE DTAPROT オプションを使用して変更することができます。
6. SECData サブコマンドのパラメーター「C」と「P」は、PROT サーバー・サブコマンドが使用するパラメーターと同じです。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

SECOpen (データ・セキュリティ保護の設定)

FTP クライアント・サブコマンド

SECOpen FTP クライアント・サブコマンドは、指定されたセキュリティ・オプションを使用して FTP サーバーへのセキュアな制御接続をオープンします。このサブコマンドの形式は次のとおりです。

```
SECOpen systemname [portnumber] [ security_option ]
```

注: SOpen は、SECOpen の同義語です。

systemname

リモート・システムの名前または IP アドレスを入力します。

portnumber

この接続のポート番号を入力します。

- 注:**
- このパラメーターを省略し、(SSL が指定された場合、ポート番号 21 が使用されます。
 - このパラメーターを省略し、(IMPLICIT が指定された場合、ポート番号 990 が使用されます。
 - ポート番号と security_option の両方が省略された場合、ポート番号 21 と (SSL が使用されます。

security_option

使用するセキュリティのタイプを指定します。

(SSL

FTP サーバーへの接続にセキュアな SSL 接続を使用します。接続の確立には、AUTH (許可) サーバー・サブコマンドが使用されます。

(IMPLICIT

FTP サーバーへの接続には、「暗黙」SSL/TLS のセキュアな接続を使用します。「暗黙」SSL 接続は、AUTH、PBSZ、および PROT サーバー・サブコマンドをサーバーに送信せずに、確立されます。この場合、指定されたポート番号について SSL/TLS 接続の折衝が行われることを予測して、サーバーを構成する必要があります。

「暗黙」SSL の場合、サーバーは、クライアントが以下のパラメーターを指定してこれらのサブコマンドを送信したかのように動作します。

- AUTH SSL
- PBSZ 0
- PROT P

注: security_options パラメーターが指定されない場合は、(SSL が使用されます。ポート番号に 990 が使用される場合は、(IMPLICIT が使用されます。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

SENDPASV (PASV サブコマンドを送信するかどうかの指定)

FTP クライアント・サブコマンド

データ転送を行うとき、または DIR サブコマンドや LS サブコマンドを発行するときに、FTP サーバーに PASV サブコマンドを送信するかどうかを指定するには、次の SENDPASV FTP クライアント・サブコマンドを入力します。

SENDPASV [0 1]

パラメーターがない場合、SENDPASV はトグル・スイッチのような働きをします。SENDPASV の値は、1 (オン) から 0 (オフ) へ、または 0 から 1 へ切り替えられます。

パラメーターがある場合、有効な値は次のとおりです。

0

PASV サブコマンドを送信しません。

1

PASV サブコマンドを送信します。これが省略時です。

iSeries の省略時 (オン) では、PASV サブコマンドを送信します。 SENDPASV がオフの場合、サーバーは PASV サブコマンドを送信しません。

注:

1. このサブコマンドは、RFC 1579「Firewall-Friendly FTP」をサポートしています。ファイアウォールを通してデータを転送する必要がある場合は、PASV サブコマンドによってデータ接続を確立する方法がより効果的です。事例によっては、PASV を使用しないと、ファイアウォールを通したデータ転送ができない場合があります。
2. FTP サーバーによっては、PASV サブコマンドをサポートしていないものもあります。そのような状況で SENDPASV がオンに設定されていると、FTP クライアントは、サーバーが PASV をサポートしていないというメッセージを表示します。システムは、PASV サブコマンドを送信せずにデータ接続を確立しようとします。
3. SENDPASV がオフまたは使用不可の場合、SENDPORT がオンに設定されていると、サーバーは PORT サブコマンドを送信します。SENDPORT (PORT サブコマンドを送信するかどうかの指定) を参照してください。
4. PASV をサポートしない FTP サーバーは RFC 1123 に準拠していません。

制限事項

SOCKS サーバーを通して FTP サーバーに接続した場合、SENDPASV サブコマンドは、なんらかのデータ転送サブコマンドやディレクトリー・リスト・サブコマンドを発行する前にのみ使用できます。これらのサブコマンドのいずれかを使用した後で SENDPASV を使用すると、クライアントは FTP サーバーへのデータ接続を確立できなくなります。

クライアントがデータ転送サブコマンドまたはディレクトリー・リスト・サブコマンドを発行したら、SENDPASV を再発行する前に、SOCKS サーバーを通して FTP サーバーとの接続をクローズします。

FTP クライアントを FTP サーバーから切断する際に、SENDPASV サブコマンドを使用することができます。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

SENDPORT (PORT サブコマンドを送信するかどうかの指定)

FTP クライアント・サブコマンド

データ転送を行うとき、または DIR サブコマンドや LS サブコマンドを発行するときに、FTP サーバーに PORT サブコマンドを送信するかどうかを指定するには、次の SENDPORT FTP クライアント・サブコマンドを入力します。

SENDPOrt [0 | 1]

パラメーターがない場合、SENDPORT はトグル・スイッチのような働きをします。SENDPORT の値は、1 (オン) から 0 (オフ) に、または 0 から 1 に変わります。

パラメーターがある場合、有効な値は次のとおりです。

0

PORT サブコマンドを送信しません。

1

PORT サブコマンドを送信します。これが省略時です。

注:

1. SENDPORT は、これを使用しないとサーバーとの接続を確立できない場合にのみ使用してください。SENDPORT を無差別に使用すると、エラーが起こることがあります。
2. システムによっては、PORT サブコマンドを無視したにもかかわらず、受け入れたものとして通知するものがありますので、このようなシステムには PORT サブコマンドを送信しないのが適当です。
3. SENDPASV オプションが ON に設定されている場合、サーバーは PORT サブコマンドを送信しません。SENDPASV (PASV サブコマンドを送信するかどうかの指定) を参照してください。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

SENDSITE (SITE サブコマンドを送信するかどうかの指定)

FTP クライアント・サブコマンド

PUT 操作または MPUT 操作の実行時に、レコード様式情報が入った SITE サブコマンドを自動的に送信するかどうかを指定するには、次の形式の SENDSITE FTP クライアント・サブコマンドを入力します。

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

パラメーターがない場合、SENDSITE はトグル・スイッチのような働きをします。SENDSITE の値は、0 (オフ) から 1 (オン) へ、または 1 から 0 へ切り替わります。

パラメーターがある場合、有効な値は次のとおりです。

0

SITE サブコマンドを送信しません。これが省略時です。

1

PUT サブコマンドおよび MPUT サブコマンドを送信する前に、SITE サブコマンド (レコード様式情報が入った) を送信します。SITE サブコマンドと共に送信されるレコード様式情報を使用する IBM 仮想計算機サーバーにファイルを送信する場合は、この設定を使用してください。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

SITE (リモート・システムが使用する情報の送信)

FTP クライアント・サブコマンド

リモート・システムに固有のサービスを提供するためにリモート・システムが使用する情報を送信するには、次の形式の SITE FTP クライアント・サブコマンドを使用します。

```
Site [parameters]
```

parameters

リモート・システムによって異なります。

これらのパラメーターの性質とその構文仕様を調べるには、`HELP SERVER SITE` サブコマンドを発行します。FTP サーバーによっては、`SITE` サブコマンドをサポートしないものがあります。

注: `SITE` サブコマンドは、レコードの様式と長さを示すために `PUT` サブコマンドと `MPUT` サブコマンドが使用します。省略時では、`PUT` サブコマンドは `SITE` サブコマンドを自動的に送信します。`NAMEFMT` サブコマンドは、`SITE` サブコマンドを使用して、名前が `NAMEFMT 0` 形式であるか `NAMEFMT 1` 形式であるかをサーバーに示します。

詳しくは、`SENDSITE` (`SITE` サブコマンドを送信するかどうかの指定) を参照してください。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

STATUS (リモート・システムからの状況情報の検索)

FTP クライアント・サブコマンド

```
STAtus [name]
```

name

状況情報が必要なリモート・ディレクトリーまたはファイルの名前。これは必須パラメーターではありません。

注: `iSeries` FTP サーバー・アプリケーションは、この `name` パラメーターをサポートしていません。

パラメーターがない場合、サーバーは、FTP サーバー・プロセスに関する一般状況情報を戻します。これには、すべての転送パラメーターの現行値と、接続の状況が含まれます。戻される状況情報は、特定のサーバーのインプリメンテーションによって異なります。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

STRUCT (ファイル構造の指定)

FTP クライアント・サブコマンド

あるファイルについて、送信するデータの構造を指定するには、次の形式の `STRUCT` FTP クライアント・サブコマンドを使用します。

```
STRuct [F | R]
```

F ファイル構造。ファイルの構造は、連続した一連のデータ・バイトです。

R レコード構造。ファイルは、一連の順次レコードとして転送されます。

ファイルの構造は、転送モードと、ファイルの解釈および保管に影響します。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

SUNIQUE (ファイルの上書きの制御)

FTP クライアント・サブコマンド

PUT サブコマンドまたは MPUT サブコマンドの実行時に、ファイルを上書きするかどうかを制御するには、次の SUNIQUE FTP クライアント・サブコマンドを入力します。

```
SUnique [ 0 | 1 ]
```

パラメーターがない場合、SUNIQUE はトグル・スイッチのような働きをします。SUNIQUE の値は、0 (オフ) から 1 (オン) へ、または 1 から 0 へ切り替わります。

パラメーターがある場合、有効な値は次のとおりです。

0

ファイルが存在する場合は上書きします。これが省略時です。

1

既存のファイルを上書きする代わりに、リモート・システム上で固有の名前を持つ新しいファイルを作成します。リモート・システム上の FTP サーバーは、作成されたファイルの名前をユーザーに送り返します。

注: リモート・システムが iSeries である場合、サーバーが作成する File.Mbr の名前は、PUT サブコマンドまたは MPUT サブコマンドで指定した *localfile* の後に番号を付けたものです。たとえば、リモート・システムに名前 *NEWFILE.NEWMBR* がすでに存在している場合、リモートの iSeries サーバーは *NEWFILE.NEWMBR1* を作成してそれにデータを書き込みます。

HFS などの他のファイル・システムのファイル名も、同様の方法で作成されます。名前がすでに存在している場合、指定したファイル名と接尾部番号で構成される新しいファイルが作成されます。たとえば、リモート・システムに *xfsname* がすでに存在している場合、リモートの iSeries は *xfsname1* を作成します。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

SYSCMD (iSeries CL コマンドをローカルの iSeries に渡す)

FTP クライアント・サブコマンド

FTP 環境を終了せずにローカルの iSeries で制御言語 (CL) コマンドを実行するには、次の形式で SYSCMD FTP クライアント・サブコマンドを使用します。

```
SYSCmd commandline
```

commandline

iSeries の CL コマンドです。コマンド名の前に ? を付けると、CL コマンドのプロンプトを表示することができます。たとえば、

```
SYSCMD ? SNDBRKMSG
```

と入力すると、中断メッセージ送信 (SNDBRKMSG) コマンドの画面が表示されます。

CL コマンドの結果である低レベルのメッセージを表示したい場合、または FTP 環境へ戻る前に複数の CL コマンドを入力したい場合は、iSeries の CALL QCMD コマンドを使用してください。

たとえば、iSeries の「コマンドの入力」画面を表示するには、次のように入力します。

```
SYSCMD CALL QCMD
```

このコマンド入力画面で、アプリケーション・プログラムを呼び出したり、CL コマンドを入力したりすることができます。アプリケーション・プログラムまたは CL コマンドが完了したら、コマンド入力画面に戻ります。この画面で、メッセージを表示したり、システムに対する追加の作業を開始したりすることができます、あるいは F3 (終了) または F12 (取り消し) を押して FTP に戻ることができます。

FTP の主画面で F21 (CL コマンド行) を押すと、iSeries CL コマンドを入力できます。「FTP クライアント要求妥当性検査」出口点に出口プログラムが追加されている場合、サーバーでは F21 キーを使用できません。

- 注:
1. 大部分のサーバー・システムにはタイムアウト期間が設定されており、一定時間内に何も活動が行われない場合は、セッションは終了します。コマンドの実行がタイムアウト期間内に終了しない場合は、サーバーはクライアントとの接続を終了します。
 2. iSeries サーバーでは感嘆符 (!) をサポートしています。SYSCMD サブコマンドと同義に使用できません。
 3. SYSCMD サブコマンドは、ユーザーが入力したとおりの情報を CL コマンドとして iSeries に渡します。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

TYPE (ファイル転送タイプの指定)

FTP クライアント・サブコマンド

ファイル転送を行うときのファイル転送タイプまたは表示方法を指定するには、次の形式の TYPE FTP クライアント・サブコマンドを使用します。

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

A

転送タイプとして省略時の (ASCII) 転送タイプを指定します。これは、ASCII サブコマンドを発行するのと同じ効果があります。サーバーは、どの垂直方向の書式もファイルに関連付けません。サーバ

ーが ASCII に対してサポートしているのは、省略時形式の NON PRINT のみです。ASCII 転送タイプは、両方のシステムが EBCDIC タイプを使用する場合を除いて、テキスト・ファイルの転送に使用します。

タイプ A (ASCII) の省略時 CCSID は、STRTCPFFTP コマンドまたは FTP サブコマンドの CCSID パラメーターで指定された CCSID です。

B

シフト JIS 漢字 (CCSID 932)

B 1

シフト JIS 漢字 (CCSID 932)

B 2

拡張 UNIX コード漢字 (CCSID 5050)

B 3

ASCII シフトイン・エスケープ・シーケンスを使用する JIS 1983 (CCSID 5054)

B 3 A

ASCII シフトイン・エスケープ・シーケンスを使用する JIS 1983 (CCSID 5054)

B 3 R

JISROMAN シフトイン・エスケープ・シーケンスを使用する JIS 1983 (CCSID 5052)

B 4

ASCII シフトイン・エスケープ・シーケンスを使用する JIS 1978 (CCSID 5055)

B 4 A

ASCII シフトイン・エスケープ・シーケンスを使用する JIS 1978 (CCSID 5055)

B 4 R

JISROMAN シフトイン・エスケープ・シーケンスを使用する JIS 1978 (CCSID 5053)

B 5

ハングル (CCSID 934)

B 6

韓国語標準コード KSC-5601、1989 年版 (CCSID 949)

B 7

繁体字中国語 (5550) (CCSID 938)

C ccsid#

転送タイプとして、システムに導入されている任意の CCSID (コード化文字セット識別コード) を指定します。CCSID 番号の前には、必ず C を付けます。

E

転送タイプとして EBCDIC を指定します。これは、EBCDIC サブコマンドを発行するのと同じ効果があります。サーバーは、どの垂直方向の書式制御もファイルに関連付けません。サーバーが

EBCDIC に対してサポートしているのは、省略時形式の NON PRINT のみです。EBCDIC 転送タイプは、内部文字表示として EBCDIC を使用するシステム間で効率的に転送を行うために使用します。

F

IBM EBCDIC 漢字 (CCSID 5035)

F 1

IBM EBCDIC 漢字 (CCSID 5035)

I

転送タイプとしてイメージを指定します。これは、**BINARY** サブコマンドを発行するのと同じ効果があります。イメージ転送タイプでは、データは 8 ビット・バイトにパックされたビット・ストリングです。イメージ転送タイプは、ファイルの保管と検索を効率的に行うため、オブジェクト・コードなどの 2 進データの転送に使用されます。データはそのまま転送され、変換は行われません。

パラメーターがない場合、サーバーは **TYPE** サブコマンドの現在の設定を表示します。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

USER (リモート・システムへのユーザー ID の送信)

FTP クライアント・サブコマンド

```
User userid [password]
```

userid

リモート・システムで使用するログオン名。

password

リモート・システムで使用するパスワード。パスワードの指定はオプションです。ログオン・パスワードが必要なリモート・システムの場合、**USER** サブコマンドを呼び出すときにパスワードを入力しないと、その入力を求めるプロンプトが出ます。

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

VERBOSE (エラー応答メッセージのテキスト表示の制御)

FTP クライアント・サブコマンド

FTP サーバー応答の表示を制御するには、次の **VERBOSE** FTP クライアント・サブコマンドを使用します。**VERBOSE** サブコマンドは、詳細表示スイッチのオンとオフを切り替えます。詳細表示がオンの場合、すべてのサーバー応答が、応答コードを含めて表示されます。オフに設定すると、一部のサーバー応答と応答コードは廃棄され、表示されません。

```
Verbose
```

参照:

- FTP クライアント・サブコマンド
- FTP クライアントの構文規則

FTP 出口プログラム

▶ FTP クライアントと FTP サーバーは、特定の出口点を通して、各出口プログラムとの通信を行います。パラメーターは、サーバーおよび出口プログラム間で受け渡しされます。交換される情報の形式は、出口点形式で指定されます。

FTP は、以下の出口点を使用します。パラメーター記述やコード例などの詳細については、以下のトピックを参照してください。

- 要求妥当性検査出口点: クライアントおよびサーバー
- サーバー・ログオン出口点

出口プログラムを正常に機能させるには、出口プログラムの導入および登録を行う必要があります。不要になったユーザーのプログラムは、出口プログラムの削除を正しく実行して、今後そのプログラムが機能しないようにする必要があります。

TCP/IP 出口点および出口点形式

以下の表には、さまざまな TCP/IP アプリケーションの出口点およびそれに関連する出口点形式の情報が記載されています。

TCP/IP 出口点	アプリケーション	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 出口点は、複数の形式を持っている場合があります。ただし、一つの出口プログラムの登録は、一つの出口点形式に対してのみ可能です。これらの形式をそれぞれ検査して、ユーザーのシステムに最も適切な形式の一つを選択します。

² この形式は、V5R1 から使用可能になっています。



要求妥当性検査出口点: クライアントおよびサーバー

要求妥当性検査出口点を使用して、FTP ユーザーが実行できる操作を制限できます。要求妥当性検査出口点は、FTP クライアントと FTP サーバーの両方で提供されます。FTP クライアントと FTP サーバー両方へのアクセスを制限するには、出口プログラムを両方の出口点に追加する必要があります。

注: FTP クライアントと FTP サーバーの出口点はどちらも、同じ出口点形式を使用するため、プログラムを 1 つ作成して両方を処理できます。

匿名 FTP をインプリメントする場合、FTP サーバー要求妥当性検査用出口プログラムを作成し、匿名 FTP ユーザーを検索サブコマンドのみに制限し、CL コマンドを実行できないようにする必要があります。

プログラムに組み込む必要のあるもの:

- 例外処理
- デバッグ
- ログ記録

許可されるコマンドおよび拒否されるコマンド

FTP 要求の妥当性を検査する出口プログラムを使用すると、操作を受け入れるか拒否するかを制御できるようになります。出口プログラムが判断した結果は、FTP クライアントまたはサーバー・アプリケーションによって実行されるすべての妥当性検査に反映されます。FTP クライアントまたは FTP サーバーのアプリケーションは、FTP クライアントまたは FTP サーバーが以下の要求のいずれかを処理するたびに、そのアプリケーションに登録された出口プログラムを呼び出します。

1. ディレクトリー / ライブラリーの作成
2. ディレクトリー / ライブラリーの削除
3. 現行ディレクトリーの設定
4. ファイル名の表示
5. ファイルの削除
6. ファイルの送信
7. ファイルの受信
8. ファイルのリネーム
9. CL コマンドの実行

コマンドを常に無条件に拒否するには、VRLQ0100 出口点形式のパラメーター 8 (操作を許可) を -1 に設定することもできます。

出口プログラムのタイムアウト・フィーチャーの有無

FTP 出口プログラムにはタイムアウト機能はありません。出口プログラムに処理できないエラーや例外が検出された場合は、FTP サーバーによりセッションが中止されます。

プログラム例

サーバー上に匿名 FTP をセットアップする際に役に立つプログラム例を利用できます。本書に記載したコードは、例を示すことを目的としています。これらのコードには、実動マシン上で現状のまま実行できるだけの機能は含まれていません。これらのサンプルは、プログラムを構築する際の開始点として使用できます。サンプルのコードの一部をコピーして、それを、独自に作成するプログラムに追加できます。サンプル・プログラムの実行は、実動システム以外のシステムで行うことをお勧めします。

例: CL で作成された FTP クライアントまたはサーバー要求妥当性検査用出口プログラム

例: ILE RPG コードで作成された FTP サーバー要求の妥当性検査出口プログラム

コード例についての特記事項

IBM は、お客様に、このプログラムをサンプルとして使用することができる非独占的な使用権を許諾します。お客様は、このサンプル・コードから、お客様独自の特別のニーズに合わせた類似のプログラムを作成することができます。

このサンプル・コードは、例として示す目的でのみ、IBM により提供されます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。

ここに含まれるすべてのプログラムは、現存するままの状態を提供され、いかなる保証条件も適用されません。不侵害、商品性、特定目的適合性に関する黙示の保証の適用も一切ありません。

例: CL で作成された FTP クライアントまたはサーバー要求妥当性検査用出口プログラム: 次に、単純な FTP 要求の妥当性検査出口プログラムの例を掲載します。プログラムを作成する際に使用した言語は、iSeries コマンド言語 (CL) です。このコードは完全なものではありませんが、クライアントやサーバー出口点用のプログラムを独自に作成する足がかりとなります。

注: 重要な法的情報については、コードについての特記事項をお読みください。

(次の例で示すあらかじめフォーマットされているテキストは、枠の外側に記載されています。)

```

/*****
/*
/* 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)

```



```
ELSE      CMD(CHGVAR VAR(%BINARY(&ALLOWOP)) +
          VALUE(&ALLOW))
```

```
END:      ENDPGM
```

例: ILE RPG コードで作成された FTP サーバー要求の妥当性検査出口プログラム: 次に、単純な FTP サーバー要求の妥当性検査出口プログラムの例を掲載します。プログラムを作する際に使用した言語は、ILE RPG プログラム言語です。このコードは完全なものではありませんが、プログラムを独自に作成する足がかりとなります。

注: 重要な法的情報については、コードについての特記事項をお読みください。

(次の例で示すあらかじめフォーマットされているテキストは、枠の外側に記載されています。)

```
* 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('CLRSVAF 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('ABCDEFGHIJKLMNopqrstuvwxyz')
*
```

```

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 IPLEnds      9     12B 0
D OPLEnds     13     16B 0
D ALLOWOPds   17     20B 0
*
C *LIKE      DEFINE  APPIDds      APPIDIN
C *LIKE      DEFINE  OPIDds       OPIDIN
C *LIKE      DEFINE  IPLEnds      IPLENIN
C *LIKE      DEFINE  OPLEnds      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 exits 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

```

VLRQ0100 出口点形式: FTP サーバー・アプリケーション要求妥当性検査出口点は、以下のとおりです。

QIBM_QTMF_SERVER_REQ

FTP クライアント・アプリケーション要求妥当性検査出口点は、以下のとおりです。

QIBM_QTMF_CLIENT_REQ

この出口点のパラメーター形式は次のインターフェースによって制御されます。

VLRQ0100

次の表は、VLRQ0100 インターフェースのパラメーターとパラメーター形式をまとめたものです。

VLRQ0100 出口点インターフェース用必須パラメーター形式

パラメーター	記述	入力 / 出力	タイプと長さ
1	アプリケーション識別コード	入力	Binary(4)
2	操作識別コード	入力	Binary(4)
3	ユーザー・プロファイル	入力	Char(10)
4	リモート IP アドレス	入力	Char(10)
5	リモート IP アドレスの長さ	入力	Binary(4)
6	操作に関する固有情報	入力	Char(*)
7	操作に関する固有情報の長さ	入力	Binary(4)
8	Allow 操作	出力	Binary(4)

パラメーターの説明

VLRQ0100 パラメーター 1:

アプリケーション識別コード

入力; Binary(4)

要求を発信している TCP/IP アプリケーション・プログラムを指定します。4 つの異なる TCP/IP アプリケーションによって VLRQ0100 インターフェースが共有されます。最初のパラメーターは、出口プログラムを呼び出すアプリケーションを識別します。可能値は以下のとおりです。

0	FTP クライアント・プログラム
1	FTP サーバー・プログラム
2	REXEC サーバー・プログラム
3	TFTP サーバー・プログラム

VLRQ0100 パラメータ 2:

操作識別コード

入力; Binary(4)

FTP ユーザーが実行したい (要求する) 操作 (コマンド) を指定します。

アプリケーション識別コード (パラメーター 1) に FTP クライアントまたは FTP サーバー・プログラムを指定されている場合の可能値は以下のとおりです。

値	操作識別コード	クライアント・サブコマンド	サーバー・サブコマンド
0	セッションの開始	Open、SECOpen	新規接続
1	ディレクトリー / ライブラリーの作成	*	MKD、XMDK
2	ディレクトリー / ライブラリーの削除	*	RMD、XRMD

値	操作識別コード	クライアント・サブコマンド	サーバー・サブコマンド
3	現行ディレクトリー / ライブラリーの設定	LCD	CWD、CDUP、XCWD、XCUP
4	ファイルのリスト	*	LIST、NLIST
5	ファイルの削除	*	DELE
6	ファイルの送信	APPEND、PUT、MPUT	RETR
7	ファイルの受信	GET、MGET	APPE、STOR、STOU
8	ファイルの名前変更	*	RNFR、RNTO
9	CL コマンドの実行	SYSCMD	RCMD、ADDm、ADDV、CRTL、CRTP、CRTS、DLTF、DLTL rzaiqmdelete.htm

注: 記号 * は、FTP クライアントの出口が認識しない制御操作を表します。FTP クライアントのサブコマンド SYSCMD を使って CL コマンドを使用する場合にのみ、クライアントはこれらの操作を使用できません。操作識別コード 9 は、CL コマンドの実行を制御します。

VLRQ0100 パラメーター 3:

ユーザー・プロファイル

入力; Char(10)

FTP セッションのユーザー・プロファイル。

VLRQ0100 パラメーター 4:

リモート IP アドレス

入力; CHAR(*)

リモート・ホスト・システムのインターネット・プロトコル (IP) アドレス。このストリングは左詰め小数点付き 10 進数 (123.45.67.89) 形式で入力します。アプリケーション識別パラメーターの設定によって、リモート・ホストはクライアントの場合とサーバー場合があります。

VLRQ0100 パラメーター 5:

リモート IP アドレス (パラメーター 4) の長さ (バイト単位)

入力; Binary(4)

リモート IP アドレス (パラメーター 4) の長さ。

VLRQ0100 パラメーター 6:

操作に関する固有情報

入力; CHAR(*)

要求された操作についての説明。このフィールドの内容は、操作識別コード (パラメーター 2) とアプリケーション識別コード (パラメーター 1) の値によって変わります。たとえば、次のようになります。

操作識別コードとアプリケーション識別コードが共に 0 の場合

操作に関する固有情報情報はありません。このフィールドは空白です。

操作識別コードが 0 で、アプリケーション識別コードが 1 の場合

操作に固有の情報には、このセッションのローカル・ホスト (FTP サーバー) へ接続する TCP/IP インターフェースの IP アドレスがあります。このストリングは左詰め小数点付き 10 進数 (123.45.67.89) 形式で入力します。

操作識別コードが 1 ~ 3 の場合

操作に関する固有情報には、操作を実行するディレクトリーまたはライブラリーの名前などがあります。ディレクトリーまたはライブラリー名の形式は絶対パス名です。

操作識別コードが 4 ~ 8 の場合

操作に関する固有情報には、操作を実行するファイルの名前などがあります。ファイル名の形式は絶対パス名です。

操作識別コードが 9 の場合

操作に固有の情報には、ユーザーが要求する iSeries の制御言語 (CL) コマンドがあります。

『VLRQ0100 出口点形式の使用上の注意』も参照してください。

VLRQ0100 パラメーター 7:

操作に関する固有情報の長さ

入力; Binary(4)

操作に関する固有情報 (パラメーター 6) の長さを指定します。出口点が操作に固有の情報を提供しない場合の長さは 0 です。

VLRQ0100 パラメーター 8:

Allow 操作

出力; Binary(4)

要求された操作の受諾 / 拒否を指定します。可能値は以下のとおりです。

-1	この操作識別コードの使用を禁止 します。 現行セッションの残りのセッションでは、この操作識別コードを無条件で拒否します。 この操作識別コードでは、再び出口プログラムが呼び出されることはありません。
0	操作を拒否します。
1	操作を許可します。
2	この操作識別コードの使用を常時 許可します。 現行セッションの残りのセッションでは、この操作識別コードを無条件で許可します。 この操作識別コードでは、再び出口プログラムが呼び出されることはありません。

VLRQ0100 出口点形式の使用上の注意: VLRQ0100 は、FTP クライアント要求妥当性検査出口点と FTP サーバー要求妥当検査性出口点の両方に使用される出口点形式です。

無効出力パラメーター

Allow 操作パラメーター (パラメーター 8) に対して戻された出力が無効な場合、FTP サーバーは要求された操作を拒否し、次のメッセージをジョブ・ログに記録します。

Data from exit program for exit point &1 is missing or not valid

例外事項

出口プログラムを呼び出すときになんらかの例外を検出すると、FTP サーバーは次のメッセージをジョブ・ログに記録します。

Exception encountered for FTP exit program &1 in library &2 for exit point &3

まとめ: 操作に関する固有情報

次の表は、各操作識別コード (VLRQ0100 パラメーター 2) で必要とされる、操作に関する固有情報 (VLRQ0100 パラメーター 6) についてまとめたものです。

操作識別コード (VLRQ0100 パラメーター 2)	操作固有の情報 (VLRQ0100 パラメーター 6)
0	アプリケーション ID=0 (パラメーター 1) の場合はなし
0	アプリケーション ID=1 または 2 (パラメーター 1) の場合、クライアント・ホストの IP アドレス (小数点付き 10 進数形式で表示)
1~3	ライブラリーまたはディレクトリーの絶対パス名。例: /QSYS.LIB/QGPL.LIB ^(a) /QOpenSys/DirA/DirAB/DirABC ^(b)
4~8	ファイルの絶対パス名。例: /QSYS.LIB/MYLIB.LIB/MYFILE.FILE/MYMEMB.MBR ^(a) /QOpenSys/DirA/DirAB/DirABC/FileA1 ^(b)

注: ^(a) - QSYS.LIB ファイル・システム・パス名は、必ず大文字で表記します。

^(b) - QOpenSys ファイル・システム・パス名は、大文字と小文字が区別され、両者が混合していても構いません。

サーバー・ログオン出口点

TCP/IP アプリケーション・サーバー・ログオン出口点を利用すれば、TCP/IP アプリケーション・サーバーに対するユーザー認証を制御できます。この出口点を使用すれば、発信元セッションのアドレスに基づいたサーバー・アクセスを実行することができます。また、ユーザー・プロファイルで定義されているものとは違うディレクトリーを初期作業ディレクトリーとして指定することもできます。

出口プログラムが出口点に追加されると、サーバーは、ユーザーがログオンしようとするたびに、ログオン出口プログラムを呼び出します。サーバーがログオン操作を継続するかどうかを指定する戻りコードの出力パラメーターは出口プログラムで設定します。代替戻りコードは、ログオン処理およびディレクトリー情報の初期化用として設定できます。

FTP サーバーにログオンするための iSeries 出口点は次のとおりです。

QIBM_QTMF_SVR_LOGON

利用できる出口点形式には、次の 3 つがあります。

- TCPL0100 出口点形式によって、この基本ログオン制御ができます。
 - ログオンの受諾 / 拒否機能
 - ユーザー・プロファイル、パスワードと現行ライブラリーの制御
- TCPL0200 出口点形式には、以下のような、ログイン処理を制御するために追加されたパラメーターがあります。
 - システムのディレクトリー上であれば場所を問わず、作業ディレクトリーを設定する機能
 - アプリケーション固有の情報を戻す機能
 - FTP クライアントと送受信される FTP データの暗号化を制御する機能
- TCPL0300 出口点形式は、TCPL0200 出口点形式を拡張したものです。これによって、OS/400 拡張パスワード・サポートが利用可能になり、さらに、パスワード・フィールドおよびディレクトリー名フィールドを、CCSID が処理するための、パラメーターが追加されています。また、そのセッションのユーザーがクライアント認証で認証されている場合には、クライアント認証が出口プログラムに提供されます。

注:

1. FTP サーバー・ログイン出口点には、1 つだけ出口プログラムを登録することができます。ユーザーは、使用する出口点形式を、3 つの形式から決定する必要があります。
2. FTP アプリケーションの場合、この出口点によって匿名 FTP を実装する機能が提供されます。また、アクセスを記録し、制御するために必要な情報も、同時に提供されます。
3. 出口点形式 TCPL0100 および TCPL0200 内のすべての文字パラメーター、および、出口点形式 TCPL0200 内の、関連付けされた CCSID を持たないすべて文字パラメーターの、出口プログラムに渡される文字データは、ジョブの CCSID に存在します。ジョブの CCSID が 65535 である場合、文字データはジョブの省略時の CCSID の中にあります。出口プログラムによって戻される、これらのパラメーター内の文字データはすべて、この同じ CCSID の中にあることになっています。

匿名 FTP の場合、以下を実行する FTP サーバー・ログオン出口プログラムを作成してください。

1. ユーザー ID ANONYMOUS からのログオンを受諾する。
2. パスワードとして電子メール・アドレスを要求する。通常、パスワードとして「有効電子メール・アドレス」が必要とされます。出口プログラムは、英数字ストリングの中にある @ 記号を検査するだけなので、語句は間違えられる場合があります。しかも、この間違いは頻繁に起こります。このことから、ユーザーの IP アドレスのログを記録しておくことが大切です。
3. パスワード・ストリングに含まれている @ 記号をチェックする。
4. ANONYMOUS ユーザーを強制的に公用ライブラリーのみアクセスさせる。TCPL0200 形式 については、パラメーター 8 の戻りコード 3 を参照してください。

プログラムに組み込む必要のあるもの:

- 例外処理
- デバッグ
- ログギング
 - FTP 要求の発信元の IP アドレスと (パスワードとして送信された) 電子メール・アドレスのログを記録します。

出口プログラムのタイムアウト・フィーチャーの有無

FTP 出口プログラムにはタイムアウト機能はありません。出口プログラムに処理できないエラーや例外が検出された場合は、FTP サーバーによりセッションが中止されます。

QTCP には権限が必要

アプリケーションによって FTP サーバーのログオン出口プログラムが呼び出されると、FTP サーバーのジョブは QTCP ユーザー・プロファイルに従って実行されます。

ログ・ファイルや出口プログラムに関連付けられているサテライト・ファイルすべてにアクセスし、書き込みを行うのに十分な権限が QTCP にあることを確認してください。

プログラム例

サーバー上に匿名 FTP をセットアップする際に役に立つプログラム例を利用できます。本書に記載するコードは、例を示すことを目的としています。これらのコードには、実動マシン上で現状のまま実行できるだけの機能は含まれていません。これらのサンプルは、プログラムを構築する際の開始点として使用できます。サンプルのコードの一部をコピーして、それを、独自に作成するプログラムに追加できます。サンプル・プログラムの実行は、実動システム以外のシステムで行うことをお勧めします。

例: CL コードで作成された FTP サーバー・ログオン出口プログラム

例: C コードで作成された FTP サーバー・ログオン出口プログラム

例: ILE RPG コードで作成された FTP サーバー・ログオン出口プログラム

コード例についての特記事項

IBM は、お客様に、このプログラムをサンプルとして使用することができる非独占的な使用権を許諾します。お客様は、このサンプル・コードから、お客様独自の特別のニーズに合わせた類似のプログラムを作成することができます。

このサンプル・コードは、例として示す目的でのみ、IBM により提供されます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。

ここに含まれるすべてのプログラムは、現存するままの状態を提供され、いかなる保証条件も適用されません。不侵害、商品性、特定目的適合性に関する黙示の保証の適用も一切ありません。

例: CL コードで作成された FTP サーバー・ログオン出口プログラム: 次に、単純な FTP サーバー・ログオン出口プログラムの例を掲載します。プログラムを作成する際に使用した言語は、iSeries コマンド言語 (CL) です。このコードは完全なものではありませんが、プログラムを独自に作成する足がかりとなります。

注: 重要な法的情報については、コードについての特記事項をお読みください。

(次の例で示すあらかじめフォーマットされているテキストは、枠の外側に記載されています。)

```
/*
/*
/* 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(&RETCDOUT)) VALUE(6)
  CHGVAR   VAR(&USRPRFOUT) VALUE('ANONYMOUS ')
  CHGVAR   VAR(&CURLIBOUT) VALUE('PUBLIC ')
  ENDDO
/* Any other user: proceed with normal logon processing. */
  ELSE    CMD(CHGVAR VAR(%BINARY(&RETCDOUT)) VALUE(1))
  ENDDO
  ELSE    CMD(CHGVAR VAR(%BINARY(&RETCDOUT)) VALUE(1))

END:      ENDPGM

```

例: C コードで作成された FTP サーバー・ログオン出口プログラム: 次に、単純な FTP サーバー・ログオン出口プログラムの例を掲載します。プログラムを作成する際に使用した言語は、C プログラミング言語です。このコードは完全なものではありませんが、プログラムを独自に作成する足がかりとなります。

注: 重要な法的情報については、コードについての特記事項をお読みください。

(次の例で示すあらかじめフォーマットされているテキストは、枠の外側に記載されています。)

```

/* Module Description *****/
/*
/*****
/*
/* Note: This program is a sample only and has NOT undergone any
/      formal review or testing.
/*
/*****
/*
/* Source File Name: qtmfsvr1gn.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.   */
/*                   qtmfsvr1gn - 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 "qsyusri.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. */
    /* Application Identifier */
    qtmfsvrlgn(*(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: qtmfsvrlgn */
/*
/* 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 qtmfsvrln(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 */
}
}
}
}
}

```



```

/* Function Specification *****/
/*
/* Function Name: CheckClientAddress
/*
/* Descriptive Name: Check the IP address of the originating session
/*
/*           from the input argument (ClientIPAddr_p) to
/*           ensure it is in valid dotted-decimal format,
/*           and that the client system is allowed access.
/*           This is an example of an input validity check.
/*
/* Notes:
/*
/* Dependencies:
/*     None
/*
/* Restrictions:
/*     None
/*
/* 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 */

```

```

        (strspn(ClientIPAddr_p, ValidChars) < Lgth_ClientIPAddr)||
        (strspn(ClientIPAddr_p, ".") EQ 1)|| /* Leading '.' in IP */
        (strspn(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

```

例: ILE RPG コードで作成された FTP サーバー・ログオン出口プログラム: 次に、単純な FTP サーバー・ログオン出口プログラムの例を掲載します。プログラムを作する際に使用した言語は、ILE RPG プログラム言語です。このコードは完全なものではありませんが、プログラムを独自に作成する足がかりとなります。

注: 重要な法的情報については、コードについての特記事項をお読みください。

(次の例で示すあらかじめフォーマットされているテキストは、枠の外側に記載されています。)

```

* 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          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          USRPRFOU        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   '0'           Z-ADD     0           i           3 0
C   '0'           SCAN      AUTIN:1    i
*
*
C   i             IFGT      0
C   AUTLENIN      SUBST(P)  AUTIN:1    Email        30
C   Z-ADD         5         RETCDOUT
*
* 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           RETCDOUT
C               ENDIF
*
C               ELSE
C               Z-ADD     0           RETCDOUT
C               ENDIF
*
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           RETCDOUT
C               ELSE
C               Z-ADD     1           RETCDOUT
C               ENDIF
*
C               ENDIF
*
C               EVAL      *INLR = *ON
C               RETURN

```

TCPL0100 出口点形式:

出口点の形式名: TCPL0100

出口点名: QIBM_QTMF_SVR_LOGON

出口点名: QIBM_QTMX_SVR_LOGON

必須のパラメーター・グループを次に示します。

1	アプリケーション識別コード	入力	Binary(4)
2	ユーザー ID	入力	Char(*)

3	ユーザー ID の長さ	入力	Binary(4)
4	認証ストリング	入力	Char(*)
5	認証ストリングの長さ	入力	Binary(4)
6	クライアント IP アドレス	入力	Char(*)
7	クライアント IP アドレス の長さ	入力	Binary(4)
8	戻りコード	出力	Binary(4)
9	ユーザー・プロファイル	出力	Char(10)
10	パスワード	出力	Char(10)
11	初期現行ライブラリー	出力	Char(10)

パラメーターの説明

アプリケーション識別コード

入力; BINARY(4) - 要求されたアプリケーション・サーバーを示します。有効値は以下のとおりです。

1

FTP サーバー・プログラム

2

REXEC サーバー・プログラム

ユーザー ID

入力; CHAR(*) - クライアント・プログラムで割り当てたユーザー ID。FTP サーバーの場合、このパラメーターは USER サブコマンドの実行結果から得たデータ・フィールドの内容を表します。

ユーザー ID の長さ

入力; Binary(4) - ユーザー ID ストリングの長さ (バイト)

認証ストリング

入力; CHAR(*) - クライアント・プログラムで割り当てるストリング (パスワードなど)

FTP サーバーの場合、このパラメーターは PASS (パスワード) サブコマンドの実行結果から得たデータ・フィールドの内容を表します。V5R1 以降は、ユーザーがクライアント認証で認証されている場合には、このパラメーターではデータはなにも提供されません。

認証ストリングの長さ

入力; Binary(4) - 認証ストリングの長さ (バイト)

注: FTP サーバーでは、ユーザーがクライアント認証で認証されている場合には、このパラメーターは 0 に設定されています。

クライアント IP アドレス

入力; CHAR(*) - セッションを開始するインターネット・プロトコル (IP) アドレス。このストリングは小数点付き 10 進数形式で左詰めに入力します。

クライアント IP アドレスの長さ

入力; Binary(4) - クライアント IP アドレスの長さ (バイト) を指定します。

戻りコード

出力; BINARY(4) - パスワードの認証を実行するためにログオン操作を受け入れるか拒否するか、および初期現行ライブラリーを一時変更するかどうかを示します。有効値は以下のとおりです。

0

ログオン操作を拒否します。ユーザー・プロファイル、パスワード、初期現行ライブラリー出力パラメーターは無視されます。

1

指定されたユーザー ID と認証ストリング、およびユーザー指定の初期現行ライブラリーを使用して、ログオン操作を継続します。ユーザー ID がユーザー・プロファイルとなり、認証ストリングがパスワードとなります。ユーザー・プロファイル、パスワード、および初期現行ライブラリーの出力パラメーターは無視されます。

注: ログオンを成功させるためには、認証ストリングとユーザー・プロファイルで指定されているパスワードが一致する必要があります。

2

入力されたユーザー ID と認証ストリングでログオン操作を継続します。このとき、初期現行ライブラリーを、初期現行ライブラリー・パラメーターによって指定されたライブラリーに一時変更します。ユーザー ID はユーザー・プロファイルです。認証ストリングはパスワードです。初期現行ライブラリー出力パラメーターを指定します。ユーザー・プロファイルとパスワードの出力パラメーターは無視されます。

注: ログオンを成功させるためには、認証ストリングとユーザー・プロファイルで指定されているパスワードが一致する必要があります。

3

ログオン操作を継続します。ユーザー・プロファイルとパスワードを、この出口プログラムの出力パラメーターから受け取った値に、一時変更します。出口プログラムが戻す、ユーザー・プロファイルで指定された初期現行ライブラリーを使用します。初期現行ライブラリーの出力パラメーターは無視されます。

注: ログオンを成功させるためには、パスワード出力パラメーターとユーザー・プロファイルで指定されているパスワードが一致する必要があります。**重要:** IBM は、出口プログラムでパスワードを直接コード化しないことを、強くお勧めします。たとえば、暗号化を使用すれば、アルゴリズムによるパスワード決定を実行することができます。

4

ログオン操作を継続します。ユーザー・プロファイル、パスワード、初期現行ライブラリーを、この出口プログラムの出力パラメーターに、一時変更します。

注: ログオンを成功させるためには、パスワード出力パラメーターとユーザー・プロファイルで指定されているパスワードが一致する必要があります。**重要:** IBM は、出口プログラムでパスワードを直接コード化しないことを、強くお勧めします。たとえば、暗号化を使用すれば、アルゴリズムによるパスワード決定を実行することができます。

5

ログオン操作を受諾します。ユーザー・プロファイル、この出口プログラムのユーザー・プロ

ファイル出力パラメーターに戻されたものに一時変更します。この出口プログラムによって戻されるユーザー・プロファイルで指定されている初期現行ライブラリーを使用します。初期現行ライブラリーとパスワード用の出力パラメーターは無視されます。

注: この値を入力すると、通常の OS/400 のパスワード処理が一時変更されます。パスワード認証はこれだけです。

6

ログオン操作を受諾します。ユーザー・プロファイルと初期現行ライブラリーを、この出口プログラムの出力パラメーターに戻されたものに一時変更します。パスワード用の出力パラメーターは無視されます。

注: この値を入力すると、通常の OS/400 のパスワード処理が一時変更されます。パスワード認証はこれだけです。

ユーザー・プロファイル

出力; CHAR(10) - このセッションに使用するユーザー・プロファイル。このパラメーターは左詰めで入力し、未満部分はブランクを入力します。

パスワード

出力; CHAR(10) - このセッションに使用するパスワード。このパラメーターは左詰めで入力し、未満部分はブランクを入力します。

初期現行ライブラリー

出力; CHAR(10) - このセッション用に設定されている初期現行ライブラリー。このパラメーターは左詰めで入力し、未満部分はブランクを入力します。

参照:

- TCPL0100 形式の使用上の注意

必要に応じて以下も参照してください。

- TCPL0200 形式: この出口点には、ログオン処理を制御するためのパラメーターがさらに追加されています。
- TCPL0300 形式: この出口点は、要求の発信元であるアプリケーション・サーバーを識別します。
- サーバー・ログオン出口点: インターフェース: この TCP/IP アプリケーション・サーバー・ログオン出口点を利用すれば、TCP/IP アプリケーション・サーバーに対するユーザー認証を制御できます。

TCPL0100 形式の使用上の注意: FTP の場合、戻された出力パラメーターの中に無効なものがあると、FTP サーバーの操作が実行できなくなります。この場合、FTP サーバーは、次のメッセージをジョブ・ログに対して発行します。「Data from exit program for exit point &1 is missing or not valid」

FTP の場合、出口プログラムを呼び出すときに、なんらかの例外が検出されると、FTP サーバーによって次のメッセージが発行されます。「Exception encountered for FTP exit program &1 in library &2 for exit point &3」

次の表は、出口プログラムが FTP サーバーに戻す戻りコード (パラメーター 8) の値に応じて、FTP サーバーが実行する内容を要約したものです。

注: 「戻り値」という値は、出口プログラムがその出力パラメーターに対して、適切な値を戻さなければならぬことを示しています。FTP サーバーはこの値を使用して、ログオン要求プロセスを完了します。

戻りコード	ユーザー・プロファイル (9)	パスワード (10)	初期ライブラリー (11)
0	無視	無視	無視
1	(ユーザー ID、パラメーター 2)	(パスワード、パラメーター 4)	(ユーザー・プロファイルから取得)
2	(ユーザー ID、パラメーター 2)	(パスワード、パラメーター 4)	戻り値
3	戻り値	戻り値	(ユーザー・プロファイルから取得)
4	戻り値	戻り値	戻り値
5	戻り値	無視	(ユーザー・プロファイルから取得)
6	戻り値	無視	戻り値

上記の表の括弧内の値は、出力値が TCP/IP アプリケーションによって無視された場合に使用される情報を表しています。無視と記載されている部分は、値を使用しなかったことを意味しています。したがって、その戻りコード値に対しては何も戻しません。

FTP サーバー (出口点 QIBM_QTMF_SVR_LOGON、アプリケーション ID 1) の場合、ユーザー ID が ANONYMOUS で出口プログラムがこの出口点に追加されているときは、パスワードを要求する際にサーバーが次の臨時応答メッセージを発行します。「331 Guest logon in process, send complete e-mail address as password.」アプリケーションは、出口プログラムを呼び出す前に、このメッセージを発行します。

サーバー・ログオンがアプリケーションによって受諾されると、FTP サーバーは次の応答メッセージを発行します。「230 Guest logon accepted, access restrictions apply」

REXEC サーバーの場合 (アプリケーション識別コード 2):

- 戻り許可操作の出力パラメーターが無効だと、REXEC サーバーの操作が実行できなくなります。REXEC サーバーは、ジョブ・ログに対して次のメッセージを発行します。「Data from exit program for exit point &1 is missing or not valid」
- REXEC サーバーが出口プログラムを呼び出すときに、なんらかの例外が検出されると、REXEC サーバーの操作は実行できなくなります。ジョブ・ログに対して次のメッセージが発行されます。「Exception encountered for REXEC exit program &1 in library &2 for exit point &3」

TCPL0200 出口点形式:

出口点の形式名: TCPL0200

出口点名: QIBM_QTMF_SVR_LOGON

必須のパラメーター・グループを次に示します。

1	アプリケーション識別コード	入力	Binary (4)
2	ユーザー ID	入力	Char(*)
3	ユーザー ID の長さ	入力	Binary(4)
4	認証ストリング	入力	Char(*)
5	認証ストリングの長さ	入力	Binary(4)
6	クライアント IP アドレス	入力	Char(*)

1	アプリケーション識別コード	入力	Binary (4)
7	クライアント IP アドレスの長さ	入力	Binary(4)
8	ログオン許可	出力	Binary(4)
9	ユーザー・プロファイル	出力	Char(10)
10	パスワード	出力	Char(10)
11	初期現行ライブラリー	入出力	Char(10)
12	初期ホーム・ディレクトリ	出力	Char(*)
13	初期ホーム・ディレクトリの長さ	入出力	Binary(4)
14	アプリケーション固有の情報	入出力	Char(*)
15	アプリケーション固有の情報の長さ	入力	Binary(4)

パラメーターの説明

アプリケーション識別コード

入力; Binary(4) - 要求を発信するアプリケーション・サーバーを指定します。有効値は以下のとおりです。

1

FTP サーバー・プログラム

ユーザー ID

入力; CHAR(*) - クライアント・プログラムで割り当てたユーザー ID。FTP サーバーの場合、このパラメーターは USER サブコマンドの実行結果から得たデータ・フィールドの内容を表します。

ユーザー ID の長さ

入力; Binary(4) - ユーザー ID スtringの長さ (バイト)

認証String

入力; CHAR(*) - クライアント・プログラムで割り当てるString (パスワードなど)

FTP サーバーの場合、このパラメーターは PASS (パスワード) サブコマンドの実行結果から得たデータ・フィールドの内容を表します。V5R1 以降は、ユーザーがクライアント認証で認証されている場合には、このパラメーターではデータはなにも提供されません。

認証Stringの長さ

入力; Binary(4) - 認証Stringの長さ (バイト)

注: FTP サーバーでは、ユーザーがクライアント認証で認証されている場合には、このパラメーターは 0 に設定されています。

クライアント IP アドレス

入力; CHAR(*) - セッションを開始するインターネット・プロトコル (IP) アドレス。このストリングは小数点付き 10 進数形式で左詰めに入力します。

クライアント IP アドレスの長さ

入力; Binary(4) - クライアント IP アドレスの長さ (バイト) を指定します。

ログオン許可

出力; Binary(4) - ログオン操作の受諾 / 拒否、およびパスワード認証の方法を指定します。有効値は以下のとおりです。

0

ログオン操作を拒否します。その他の出力パラメーターはすべて無視します。

1

入力されたユーザー ID と認証ストリングでログオン操作を継続します。ユーザー ID がユーザー・プロファイルとなり、認証ストリングがパスワードとなります。現行ライブラリーと作業ディレクトリーは、出力パラメーターの設定に基づきます。アプリケーションは、ユーザー・プロファイルとパスワードの出力パラメーターを無視します。

注: ログオンを成功させるためには、認証ストリングとユーザー・プロファイルで指定されているパスワードが一致する必要があります。

2

ログオン操作を継続します。ユーザー・プロファイルとパスワードを、この出口プログラムの出力パラメーターで戻された値に、一時変更します。アプリケーションは、現行ライブラリーと作業ディレクトリーを、出力パラメーターの設定に基づいて初期設定します。

注: ログオンを成功させるためには、パスワード出力パラメーターとユーザー・プロファイルで指定されているパスワードが一致する必要があります。**重要:** IBM は、出口プログラムでパスワードを直接コード化しないことを、強くお勧めします。たとえば、暗号化を使用すれば、アルゴリズムによるパスワード決定を実行することができます。

3

ログオン操作を受諾します。ユーザー・プロファイルを、この出口プログラムのユーザー・プロファイル出力パラメーターで戻されたプロファイルに一時変更します。プログラムは、現行ライブラリーと作業ディレクトリーを、出力パラメーターの設定に基づいて初期設定します。パスワード出力パラメーターは無視します。

注: ご使用のシステムがセキュリティー・レベル 20 以上で動作している場合、この値を入力すると、通常の OS/400 パスワード処理が一時変更されます。パスワード認証はこれだけです。

ユーザー・プロファイル

出力; CHAR(10) - このセッションに使用するユーザー・プロファイル。必要な場合、このパラメーターは左詰めを入力し、未満部分はブランクを入力します。

パスワード

出力; CHAR(10) - このセッションに使用するパスワード。必要な場合、このパラメーターは左詰めを入力し、未満部分はブランクを入力します。

初期現行ライブラリー

出力; CHAR(10) - このセッションに使用する初期現行ライブラリー。必要な場合、このパラメーターは左詰めで入力し、未満部分は空白を入力します。出口プログラムが呼び出されると、このパラメーターは次の特殊値が設定されます。

*CURLIB

ユーザー・プロファイルが指定する現行ライブラリーを使用します。

初期ホーム・ディレクトリー

出力; CHAR(*) - このセッションに使用するホーム・ディレクトリーの初期設定。このパラメーターを入力するときは、有効絶対パス名であり、初期ホーム・ディレクトリー・パラメーターの長さが正しく設定されていることを確認する必要があります。

初期ホーム・ディレクトリーの長さ

入力 / 出力; Binary(4) - 出口プログラムによって戻される初期ホーム・ディレクトリー・パラメーターの長さ。このパラメーターは、アプリケーションが出口プログラムを呼び出すと、ゼロに初期設定されます。出口プログラムによってパラメーターの値が変更されない場合は、ホーム・ディレクトリーはユーザーのプロファイルで指定されているホーム・ディレクトリーに初期設定されます。

アプリケーション固有の情報

入力 / 出力; CHAR(*) - アプリケーション固有のログオン設定を通信する際に使用する情報。正しい形式については、『アプリケーション固有の情報パラメーターの形式』を参照してください。

アプリケーション固有の情報の長さ

入力; Binary(4) アプリケーション固有の情報の長さ (バイト単位)。

必要に応じて以下も参照してください。

- サーバー・ログオン出口点: インターフェース: この TCP/IP アプリケーション・サーバー・ログオン出口点を利用すれば、TCP/IP アプリケーション・サーバーに対するユーザー認証を制御できます。

アプリケーション固有の情報パラメーターの形式: アプリケーション識別コードによって FTP サーバー・プログラムが指定されている場合、アプリケーション固有の情報パラメーターには次のフィールドがあります。

オフセット 10 進数	オフセット 16 進数	タイプ	フィールド
0	0	BINARY(4)	初期名形式
4	4	BINARY(4)	初期現行作業ディレクトリー
8	8	BINARY(4)	形式を表示している初期ファイル
12	C	BINARY(4)	制御接続セキュリティー機構
16	10	BINARY(4)	データ接続暗号化オプション
20	14	BINARY(2)	制御接続暗号コード
22	16	BINARY(2)	データ接続暗号コード

フィールドの説明

初期名形式

このセッションのファイル名形式の初期設定を指定します。出口プログラムが呼び出されると、このフィールドの値は NAMEFMT パラメーターで指定されている FTP サーバー構成ファイルの値に一致するよう設定されます。有効値は以下のとおりです。

0

LIBRARY/FILE.MEMBER 命名形式を使用します。この設定は、CHGFTP コマンドの NAMEFMT(*LIB) オプションに該当するものであり、FTP サーバーに対して SITE NAMEFMT 0 サブコマンドを指定した場合と同じ結果が得られます。

1

パス命名形式を使用します。この設定は、CHGFTP コマンドの NAMEFMT(*PATH) オプションに該当するものであり、FTP サーバーに対して SITE NAMEFMT 1 サブコマンドを指定した場合と同じ結果が得られます。

初期現行作業ディレクトリー

FTP サーバーの現行作業ディレクトリーの初期設定を指定します。このディレクトリーはファイルやリストの操作として使う省略時のディレクトリーとなります。出口プログラムが呼び出されると、このフィールドの値は CURDIR で指定されている FTP サーバー構成ファイルの値に一致するように設定されます。有効値は以下のとおりです。

0

現行ライブラリーを FTP サーバーの初期現行作業ディレクトリーとして利用します。この設定は CHGFTP コマンドの CURDIR(*CURLIB) オプションに該当するものです。

1

ホーム・ディレクトリーを FTP サーバーの初期現行作業ディレクトリーとして利用します。この設定は CHGFTP コマンドの CURDIR(*HOMEDIR) オプションに該当するものです。

注: このフィールドを 1 に設定した場合、初期名形式フィールドも 1 に設定する必要があります。

初期ファイル・リスト形式

このセッションのファイル・リスト形式の初期設定を識別します。出口プログラムが呼び出されると、このフィールドの値は LISTFMT パラメーターで指定した FTP サーバー構成値に一致するように設定されます。有効値は以下のとおりです。

0

iSeries サーバー・ファイル・リスト形式を使用します。この設定は、CHGFTP コマンドの LISTFMT(*DFT) オプションに該当するものであり、FTP サーバーに対して SITE LISTFMT 0 サブコマンドを指定した場合と同じ結果が得られます。

1

UNIX ファイル・リスト形式を使用します。この設定は、CHGFTP コマンドの LISTFMT(*UNIX) オプションに該当するものであり、FTP サーバーに対して SITE LISTFMT 1 サブコマンドを指定した場合と同じ結果が得られます。

制御接続セキュリティー機構

この FTP セッションに対する制御接続に使用されるセキュリティー機構を識別します。有効値は以下のとおりです。

0

制御接続は保護されていません。 **1**

制御接続は、Sockets Layer (SSL) を使用して保護されています。AUTH サブコマンドで FTP クライアントによって指定される機構は、TLS-P または SSL です。

2

制御接続は SSL を使用して保護されています。AUTH サブコマンドでクライアントによって指定される機構は TLS-C または TLS です。

- 注:
- このフィールドは、出口プログラムへの入力専用フィールドです。出口プログラムによる変更は無視されます。
 - 機密保護機能のある FTP ポートに接続中のセッションに対して、この値は 1 に設定されています。機密保護機能のある FTP ポートへの接続は、暗黙的 AUTH SSL サブコマンドが FTP サーバーに送信されたかのように振る舞います。

データ接続暗号化オプション

この FTP セッションに対する FTP データ接続を暗号化するかどうかを指定します。有効値は以下のとおりです。

-1

FTP データ接続の暗号化は、この FTP セッションでは許可されていません。

0

FTP データ接続の暗号化は、この FTP セッションでは許可されています (必須ではありません)。

1

FTP データ接続の暗号化は、この FTP セッションでは必須です。

- 注:
- 制御接続セキュリティー機構の値が 1 の場合に、データ接続暗号化オプションを -1 に設定すると、データ転送を正常に行うには、クライアントからの FTP サブコマンドがさらに必要です。(TLS-P/SSL セキュリティー機構は、省略時で、データ接続を暗号化します。)
 - 制御接続セキュリティー機構の値が 2 の場合に、データ接続暗号化オプションを 1 に設定すると、データ転送を正常に行うには、クライアントからの FTP サブコマンドがさらに必要です。(TLS-C/TLS セキュリティー機構は、省略時で、データ接続を暗号化しません。)

制御接続の暗号コード

この FTP セッションへの制御接続の暗号化に使用される SSL 暗号コードを識別します。暗号コード値は、Secure Sockets Layer (SSL) API で定義されます。これらの API については、iSeries Information Center の『プログラミング』にある『Secure Sockets Layer (SSL) APIs』のトピックを参照してください。

- 注:
- このフィールドは、出口プログラムへの入力専用フィールドです。出口プログラムによる変更は無視されます。
 - この値は、制御接続セキュリティー機構の値が 1 または 2 の場合のみ有効です。

データ接続暗号コード

この FTP セッションへのデータ接続データの暗号化に使用される SSL 暗号コードを識別します。出口プログラムが呼び出されると、この値は 0 に設定されています。このことは、Secure Sockets Layer サポートが、使用される暗号コードを折衝できるということです。出口プログラムがこのフィールドを変更する場合には、有効な暗号コード値を指定する必要があります。暗号コード値は、Secure Sockets Layer (SSL) API で定義されます。これらの API については、iSeries Information Center の『プログラミング』にある『Secure Sockets Layer (SSL) APIs』のトピックを参照してくだ

さい。

- 注:
- このフィールドは、制御接続セキュリティー機構が 0 になっている場合、またはデータ接続暗号化オプションが -1 になっている場合には、無視されます。
 - このフィールドを 0 以外の値に設定する、または制御接続暗号コード・フィールドで指定されている値以外の値に設定すると、指定された暗号コードがその FTP クライアントでサポートされていない場合には、FTP サーバーおよび FTP クライアント間の SSL ハンドシェイクの実行は失敗します。

TCPL0300 出口点形式:

出口点の形式名: TCPL0300

出口点名: QIBM_QTMF_SVR_LOGON

出口点名: QIBM_QTMX_SVR_LOGON

必須のパラメーター・グループを次に示します。

1	アプリケーション識別コード	入力	Binary (4)
2	ユーザー ID	入力	Char(*)
3	ユーザー ID の長さ	入力	Binary(4)
4	認証ストリング	入力	Char(*)
5	認証ストリングの長さ	入力	Binary(4)
6	認証ストリングの CCSID	入力	Binary(4)
7	クライアント IP アドレス	入力	Char(*)
8	クライアント IP アドレスの長さ	入力	Binary(4)
9	ログオン許可	出力	Binary(4)
10	ユーザー・プロファイル	出力	Char(10)
11	パスワード	出力	Char(*)
12	パスワードの長さ	出力	Binary(4)
13	パスワードの CCSID	出力	Binary(4)
14	初期現行ライブラリー	入出力	Char(10)
15	初期ホーム・ディレクトリー	出力	Char(*)
16	初期ホーム・ディレクトリーの長さ	入出力	Binary(4)
17	初期ホーム・ディレクトリーの CCSID	入出力	Binary(4)
18	アプリケーション固有の情報	入出力	Char(*)
19	アプリケーション固有の情報の長さ	入力	Binary(4)

パラメーターの説明

アプリケーション識別コード

入力; Binary(4) - 要求を発信するアプリケーション・サーバーを指定します。有効値は以下のとおりです。

1

FTP サーバー・プログラム

2

REXEC サーバー・プログラム

ユーザー ID

入力; CHAR(*) - クライアント・プログラムで割り当てたユーザー ID。

FTP サーバーの場合、このパラメーターは USER サブコマンドの実行結果から得たデータ・フィールドの内容を表します。

ユーザー ID の長さ

入力; Binary(4) - ユーザー ID スtringの長さ (バイト)

認証String

入力; CHAR(*) - クライアント・プログラムで割り当てるString (パスワードなど)

FTP サーバーの場合、このパラメーターは PASS (パスワード) サブコマンドの実行結果から得たデータ・フィールドの内容を表します。(ユーザーがクライアント認証で認証されていない場合。認証されている場合には、このパラメーターに対してクライアント認証が提供されます)。

認証Stringの長さ

入力; Binary(4) - 認証Stringの長さ (バイト)

認証Stringの CCSID

入力; BINARY(4) 認証String・パラメーターの CCSID。FTP サーバーでは、ユーザーがクライアント認証で認証されている場合には、このパラメーターは -2 に設定されています。

クライアント IP アドレス

入力; CHAR(*) - セッションを開始するインターネット・プロトコル (IP) アドレス。このStringは小数点付き 10 進数形式で左詰めに入力します。

クライアント IP アドレスの長さ

入力; Binary(4) - クライアント IP アドレスの長さ (バイト) を指定します。

ログオン許可

出力; Binary(4) - ログオン操作の受諾 / 拒否、およびパスワード認証の方法を指定します。有効値は以下のとおりです。

0 ログオン操作を拒否します。その他の出力パラメーターはすべて無視します。

1 入力されたユーザー ID と認証Stringでログオン操作を継続します。ユーザー ID がユーザー・プロファイルとなり、認証Stringがパスワードとなります。現行ライブラリーと作業ディレクトリーは、出力パラメーターの設定に基づきます。アプリケーションは、ユーザー・プロファイルとパスワードの出力パラメーターを無視します。

注: ログオンを成功させるためには、認証ストリングとユーザー・プロファイルで指定されているパスワードが一致する必要があります。

2

ログオン操作を継続します。ユーザー・プロファイルとパスワードを、この出口プログラムの出力パラメーターで戻された値に、一時変更します。アプリケーションは、現行ライブラリーと作業ディレクトリーを、出力パラメーターの設定に基づいて初期設定します。

注: ログオンを成功させるためには、パスワード出力パラメーターとユーザー・プロファイルで指定されているパスワードが一致する必要があります。**重要:** IBM は、出口プログラムでパスワードを直接コード化しないことを、強くお勧めします。たとえば、暗号化を使用すれば、アルゴリズムによるパスワード決定を実行することができます。

3

ログオン操作を受諾します。ユーザー・プロファイルを、この出口プログラムのユーザー・プロファイル出力パラメーターで戻されたプロファイルに一時変更します。プログラムは、現行ライブラリーと作業ディレクトリーを、出力パラメーターの設定に基づいて初期設定します。パスワード出力パラメーターは無視します。

注: ご使用のシステムがセキュリティー・レベル 20 以上で動作している場合、この値を入力すると、通常の OS/400 パスワード処理が一時変更されます。パスワード認証はこれだけです。

ユーザー・プロファイル

出力; CHAR(10) - このセッションに使用するユーザー・プロファイル。必要な場合、このパラメーターは左詰めを入力し、未満部分はブランクを入力します。

パスワード

出力; CHAR(*) - このセッションに使用するホーム・ディレクトリーの初期設定。指定する必要がある場合には、パスワードの長さパラメーター、およびパスワードの CCSID パラメーターも、指定する必要があります。このパラメーターは、左寄せです。QPWDLVL システム値が 0 または 1 に設定されている場合には、10 文字まで指定できます。QPWDLVL システム値が 2 または 3 に設定されている場合には、128 文字まで指定できます。

パスワードの長さ

出力; BINARY(4) - パスワードの長さ (バイト)。指定する必要がある場合には、有効な範囲は 1 から 512 バイトです。

パスワードの CCSID

出力; BINARY(4) - パスワードの CCSID。このパラメーターは、パスワード・パラメーターが指定されている場合には、出口プログラムが設定する必要があります。有効値は以下のとおりです。

0

ジョブの CCSID は、変換されるデータの CCSID を決定するために使用します。ジョブ CCSID が 65535 の場合には、省略時 CCSID (DFTCCSID) ジョブ属性の CCSID が使用されます。

1-65533

この範囲内の有効な CCSID。

初期現行ライブラリー

出力; CHAR(10) - このセッションに使用する初期現行ライブラリー。必要な場合、このパラメーターは左詰めで入力し、未満部分はブランクを入力します。出口プログラムが呼び出されると、このパラメーターには次の特殊値が設定されます。*CURLIB - ユーザー・プロファイルで指定する現行ライブラリーを使用します。

初期ホーム・ディレクトリー

出力; CHAR(*) - このセッションに使用するホーム・ディレクトリーの初期設定。このパラメーターを入力するときは、このパラメーターは有効絶対パス名であり、初期ホーム・ディレクトリーの長さ、初期ホーム・ディレクトリー・パラメーターの CCSID は、正しい値に設定されていることを確認する必要があります。

初期ホーム・ディレクトリーの長さ

入力 / 出力; Binary(4) - 出口プログラムによって戻される初期ホーム・ディレクトリー・パラメーターの長さ。このパラメーターは、アプリケーションが出口プログラムを呼び出すと、ゼロに初期設定されます。出口プログラムによってパラメーターの値が変更されない場合は、ホーム・ディレクトリーはユーザーのプロファイルで指定されているホーム・ディレクトリーに初期設定されます。

初期ホーム・ディレクトリーの **CCSID** 出力; Binary(4) 初期ホーム・ディレクトリーの CCSID。このパラメーターは、初期ホーム・ディレクトリーが指定されている場合には、出口プログラムが設定する必要があります。有効値は以下のとおりです。

0

ジョブの CCSID は、変換されるデータの CCSID を決定するために使用します。ジョブ CCSID が 65535 の場合には、省略時 CCSID (DFTCCSID) ジョブ属性の CCSID が使用されます。

1-65533

この範囲内の有効な CCSID。

アプリケーション固有の情報

入力 / 出力; CHAR(*) - アプリケーション固有のログオン設定を通信する際に使用する情報。正しい形式については、『アプリケーション固有の情報パラメーターの形式』を参照してください。

アプリケーション固有の情報の長さ

入力; Binary(4) - アプリケーション固有の情報の長さ (バイト)。

必要に応じて以下も参照してください。

- サーバー・ログオン出口点: インターフェース: この TCP/IP アプリケーション・サーバー・ログオン出口点を利用すれば、TCP/IP アプリケーション・サーバーに対するユーザー認証を制御できます。

出口プログラムの削除

導入済みの出口プログラムを削除するには、次のようにします。

1. iSeries のコマンド行で **WRKREGINF** と入力します。
2. ページを送って次の FTP サーバー・ログオン出口点を表示します。

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

3. 出口点の入力箇所の左側にある Opt フィールドに **8** を入力し、**Enter** を押します。

4. 「**出口プログラムの処理**」画面に、**4** (削除) を入力します。
5. 「**出口プログラム**」フィールドに出口プログラムの名前を入力します。
6. 出口プログラムを格納するライブラリーの名前を「**ライブラリー**」フィールドに入力します。
7. **Enter** キーを押します。
8. 出口点の削除が終了したら、FTP サーバーをいったん終了させて再始動します。

データの転送方法

ファイル転送を開始する前に、適切なファイル転送タイプを選択する必要があります。省略時タイプや ASCII を使用することも、異なるタイプを指定することもできます。ASCII は、インターネット標準の、文字エンコード方式です。EBCDIC は、iSeries の標準です。以下に従って、適切なタイプを選択します。

- テキストのみが含まれているファイル (「**テキストのみ**」ファイル) を転送する場合は、ASCII を使用します。
- EBCDIC をサポートする 2 つのシステム間で EBCDIC データを転送する場合は、EBCDIC を使用します。これによって、両方のシステム上で EBCDIC と ASCII のデータ変換を行う必要はなくなります。
- 非テキスト・ファイル (バイナリー数値データ、グラフィックス・ファイル、および iSeries 保管ファイルなど) を転送する場合は、BINARY を使用します。

ファイル転送形式を選択すると、FTP を使用したファイル転送への準備完了です。

以下のトピックでは、特定のファイル・タイプについての詳細を説明しています。

- iSeries サーバー間でのパック 10 進データを含むファイルの転送
- *SAVF ファイルの転送
- QDLS 文書の転送
- “root”、QOpenSys、QLANSrv、QDLS、および QOPT ファイルの転送
- QfileSvr.400 を使用したファイルの転送
- QSYS.LIB ファイルの転送
- ファイルを事前に作成する場合の考慮事項
- CCSID の変換

iSeries サーバー間でのパック 10 進データを含むファイルの転送

➤ パック 10 進数やゾーン 10 進数など、特別な数値形式の変換は FTP ではサポートされていません。

パック 10 進数やゾーン 10 進数のデータの転送は、伝送モードを BLOCK に指定した転送タイプ TYPE I (BINARY) または TYPE E (EBCDIC) を使用する場合に、iSeries サーバー間でサポートされます。これらの転送タイプでは、データはそのまま変換されずに送信されます。これ以外の転送タイプでの送信結果は、予測できません。

外部記述 QSYS.LIB ファイル内のパック 10 進数やゾーン 10 進数のデータを転送する場合、ソース・ファイルと同じ方法でターゲット・ファイルを事前に作成しておく必要があります。この制限事項は、特殊な数値形式を含むデータまたはキー順アクセスが必要な場合に適用されます。

転送タイプがバイナリーのデータを転送する場合は、ターゲット・ファイルのレコード長をソース・ファイルのレコード長と同じにする必要があります。

他のシステム・アーキテクチャー (S/390^R や UNIX など) との間でパック 10 進数やゾーン 10 進数のデータを転送するにはまず、データを印刷できる形式に変換する必要があります。 ⚡

*SAVF ファイルの転送

▶ *SAVF ファイルはイメージとして送信する必要があるため、GET や PUT サブコマンドを実行する前に FTP BINARY サブコマンドを実行する必要があります。

命名形式 0 を使用して *SAVF ファイルを転送する場合、受信側のシステムの保管ファイルを事前に作成しておく必要があります。パフォーマンスと整合性の理由から、他の状況においてもファイルを事前に作成しておくことをお勧めします。

保管ファイルは iSeries に特有のファイル形式であるため、保管ファイルの転送は、送信サーバーと受信サーバーが両方とも iSeries サーバーであるときにのみ、有効になります。しかし、保管ファイルは、バックアップの目的で iSeries 以外のサーバーに送信し、保管することができます。保管ファイルは、後に FTP 経由で iSeries に転送することができます。

例: VM から iSeries への *SAVF ファイルの転送

以下の例では、NAMEFMT 0 および 1 の両方を使用する場合に、*SAVF ファイルを VM から iSeries に転送する方法を示します。FTP セッションはすでに開始済みであり、BINARY サブコマンドが実行され、NAMEFMT 0 が指定されています。

最初に、ファイル P162484 SAVF310L を VM の A ディスクから iSeries に転送します。VM FTP では、ファイル名とファイル・タイプの間にはピリオドを挿入する必要があります。このファイルに、iSeries のライブラリー P162484 にあるファイル名 P162484 を指定します。このファイルは以前に使用されたことはありませんが、事前に作成しておいたので、REPLACE を指定します。NAMEFMT 0 では、事前作成が必須であることを思い出してください。

NAMEFMT を 1 に変更し、新規の命名形式を使用してファイル転送を繰り返します。このファイルは直前のステップからの既存のファイルであるため、REPLACE を再度指定します。

注:

- NAMEFMT 0 で転送を実行する前に、iSeries でファイルを事前に作成していなかった場合でも、見かけ上、転送は正常に完了したかようになります。しかし、iSeries でこのファイルを検査してみると、保管ファイル (*SAVF) ではなく、物理ファイル (*PF) が作成されたことが分かります。
- *SAVF ファイルが VM に送信される方法によっては、VM システムでプリプロセスが一部必要な場合があります。
 - FTP を使用して *SAVF ファイルを VM に送信した場合は、GET サブコマンドを実行して、ファイルを iSeries に戻すことができます。
 - ネットワーク・ファイルの送信 (SNDNETF) コマンドを使用して *SAVF ファイルを VM に送信した場合は、FTP を使用してファイルを iSeries サーバーに戻す前にまず、VM システム上でファイルを可変のレコード形式 (RECFM) から固定の RECFM に変換する必要があります。それには、VM で COPYFILE コマンドを使用します。たとえば、次のようになります。

```
COPYFILE P162484 SAVF310L A = = = (RECFM F REPLACE
```



```

> GET P162484.SAVF310L P162484/P162484 (REPLACE
200 Port request OK.
150 Sending file 'P162484.SAVF310L'
250 Transfer completed successfully.
384912 bytes transferred in 3.625 seconds. Transfer rate 106.183 KB/sec

> namefmt 1
202 SITE not necessary; you may proceed
Client NAMEFMT is 1.
> GET P162484.SAVF310L/QSYS.LIB/P162484.LIB/P162484.savf (REPLACE
200 Port request OK.
150 Sending file 'P162484.SAVF310L'
250 Transfer completed successfully.
384912 bytes transferred in 3.569 seconds. Transfer rate 107.839 KB/sec
FTP サブコマンドを入力してください。
====>

```

図 1. NAMEFMT 0 と NAMEFMT 1 を使用して *SAVF を VM から iSeries へ転送



QDLS 文書の転送

➤ QDLS 文書が転送される際、文書タイプを表す QDLS ディレクトリー項目属性は、変更可能形式テキスト (RFT) 文書以外のすべての文書タイプについて、受信する iSeries サーバー上では省略時で文書タイプ PCFILE になります。RFT 文書は、省略時で文書タイプ RFTDCA になります。RFTDCA タイプの文書は、WRKDOC CL コマンドを使用して表示および編集できます。PCFILE タイプの文書は、WRKDOC CL コマンドを使用して表示または編集できません。 ⏪

“root”、QOpenSys、QLANSrv、QDLS、および QOPT ファイルの転送

➤ ファイルを “root”、QOpenSys、QLANSrv、QDLS、および QOPT ファイル・システムで転送する場合、ストリーム・モード (MODE S) およびファイル構造 (STRUCT F) を使用する必要があります。

“root”、QOpenSys、QDLS、および QOPT ファイルは、どのような有効なコード・ページにも存在することができます。QLANSrv ファイル・システムに転送されるファイルは、そのファイルを含むディレクトリーに対応するネットワーク・サーバー記述に定義されているコード・ページでタグ付けされます。

データの変換および CCSID の割り当ては、使用する転送タイプによって異なります。『iSeries ファイルの CCSID コード・ページのタグ付け』を参照してください。TYPE E は、QLANSrv ファイル・システムではサポートされていません。

データを既存のファイルに追加する場合、そのファイルの CCSID タグは変更されません。データを既存のファイルに TYPE A を使用して追加する場合、データはその既存のファイルのコード・ページに変換されます。 ⏪

QfileSvr.400 を使用したファイルの転送

➤ QFileSvr.400 ファイル・システムは、リモート iSeries サーバーにある他のファイル・システムへのアクセスを提供します。“root”、QOpenSys、QLANSrv、QDLS、および QOPT ファイル・システムのファイルの転送がサポートされています。QSYS.LIB ファイル・システムのファイルの転送はサポートされていません。

互換性のあるレコード長およびファイル・サイズ

すでに存在するファイルにデータを送信する場合、受信する側のファイルのレコードとファイルのサイズには、送信されるファイルとの互換性が必要です。互換性がないと、転送エラーが発生します。受信する側のファイルのレコードとファイルのサイズは、ソース・ファイルのレコードとファイルのサイズより大きい、または等しくなっている必要があります。既存のファイルのサイズと互換性があるかどうかを判別するには、現在のレコード数、使用できる拡張子の数、および使用できるレコードの最大サイズを考慮する必要があります。この情報は、ファイル記述表示 (DSPFD) コマンドを入力することで表示できます。

iSeries サーバーでの自動ファイル生成

iSeries サーバーは、ファイルを受信するときに物理ファイルが存在しない場合、自動的に物理ファイルを作成します。しかし、iSeries 上でファイルを事前に作成することをお勧めします。

データ・タイプ

TYPE I を使用してデータを転送する場合、データは変換されません。ファイルが存在しない場合、ファイルの作成時に CCSID 65535 でタグ付けされます。

注: MGET および MPUT サブコマンドを使用して複数のメンバーがあるファイルを転送する場合、ファイルを事前に作成することをお勧めします。ファイルが事前に作成されていない場合、FTP は、最大レコード長を最初に処理されたメンバーの最長のレコードに等しいファイルを作成します。他のファイル・メンバーのレコード長がこれより長い場合、そのメンバーを転送するときにデータ切り捨てのエラーが発生します。すべてのメンバーに対応できるようなレコード・サイズを使用してファイルを事前に作成することで、このエラーを防ぐことができます。

表 1: QSYS.LIB ファイル・システムのストリーム転送モード

ライブラリーが存在する	ファイルが存在する	メンバーが存在する	選択項目を置き換える	レコード長に互換性がある	ファイル・サイズに互換性がある	結果
はい	はい	はい	はい	はい	はい	データがメンバーに書き込まれる。
はい	はい	はい	いいえ	N/A	N/A	転送は拒否され、メッセージが送信される。
はい	はい	いいえ	N/A	いいえ	はい	ファイルの転送は完了するが、レコードは切り捨てられ、メッセージが戻される。
はい	はい	いいえ	はい	いいえ	はい	ファイルの転送は完了するが、レコードは切り捨てられ、メッセージが戻される。

ライブラリーが存在する	ファイルが存在する	メンバーが存在する	選択項目を置き換える	レコード長に互換性がある	ファイル・サイズに互換性がある	結果
はい	はい	いいえ	N/A	はい	はい	メンバーが作成され、データがメンバーに書き込まれる。
はい	はい	いいえ	いいえ	N/A	いいえ	転送は拒否され、メッセージが送信される。
はい	いいえ	N/A	N/A	N/A	N/A	着信するファイルの最大レコード長にレコード長が等しいファイルが作成される。メンバーが作成され、データがメンバーに書き込まれる。
いいえ	N/A	N/A	N/A	N/A	N/A	転送は拒否され、メッセージが送信される。CRTLIB コマンドを使用してリモート iSeries サーバーにライブラリーを作成する。

表 2: QSYS.LIB ファイル・システムのイメージ転送タイプ

ライブラリーが存在する	ファイルが存在する	メンバーが存在する	選択項目を置き換える	結果
はい	はい	はい	はい	データがメンバーに書き込まれる。
はい	はい	はい	いいえ	転送は拒否され、メッセージが送信される。
はい	はい	いいえ	N/A	作成されたメンバーとデータ
はい	いいえ	N/A	N/A	
いいえ	N/A	N/A	N/A	



QSYS.LIB でのテキスト・ファイルの受信: ➤ iSeries の QSYS.LIB ファイル・システムは、内部的にレコード構造をサポートするため、iSeries FTP は iSeries サーバーが受信するファイルをレコード構造に変換し、iSeries サーバーから送信されるファイルを FTP ファイル構造に変換します。FTP 経由で iSeries サーバーが受信したテキスト・ファイルは、以下の方法でレコード構造に変換されます。

- FTP ファイルを受信し、そのファイルがすでに iSeries サーバーに存在する場合、既存のファイルのレコード長が使用されます。
- FTP が iSeries サーバー上で新規ファイルを作成する場合、ファイルの最長の行またはレコード (末尾のスペースは除く) をファイルのレコード長に使用します。

FTP 経由で iSeries サーバーから送信するテキスト・ファイルは、各行または各レコードから末尾ブランクを除去して、その切り捨てられたレコードを送信することで、ファイル構造に変換されます。 ⏪

ファイルを事前に作成する場合の考慮事項

➤ iSeries の QSYS.LIB ファイル・システムに転送されるファイルは、事前に作成することを強くお勧めします。これは、パフォーマンスと整合性を最適に保ちながら、データが確実に効率良く転送されることを保証する最良の方法です。

ファイル全体に対応するために十分な数のレコードを割り振るようにします。iSeries では、これは物理ファイルの作成 (CRTPF) コマンドの SIZE パラメーターで行います。

予測される最大レコード長に対応できるように、物理ファイルの作成 (CRTPF) コマンドの RCDLEN パラメーターが適切に指定されていることを確認してください。

注: FTP サーバー・システムでは、QUOTE サブコマンドを使用してファイルを事前に作成できます。FTP クライアント・システムでは、SYSCMD サブコマンドを使用してファイルを事前に作成できます。



CCSID の変換

➤ iSeries では、コード化文字セット識別コード (CCSID) の情報を使用して入力データを解釈し、ディスプレイ用の出力データを適切な形式で提供します。入力には、ASCII または EBCDIC を使用できます。以下のトピックには、CCSID の変換についての詳細情報が記載されています。

- マッピング・テーブルの指定
- iSeries ファイルの CCSID コード・ページのタグ付け
- FTP における各国語サポートについての考慮事項



マッピング・テーブルの指定: ➤ FTP クライアントでは、ASCII マッピング・テーブルは FTP コマンドで指定されます。FTP サーバーでは、これは FTP 属性の変更 (CHGFTP) コマンドで行われます。FTP クライアントのマッピング・テーブルを指定するには、以下のように行います。

1. コマンド FTP を入力します。
2. **PF4** キーを押します。「TCP/IP FTP の開始 (Start TCP/IP FTP)」画面が表示されます。
3. **F10** キーを押します。発信および着信する ASCII/EBCDIC テーブルを入力するようプロンプトが表示されます。

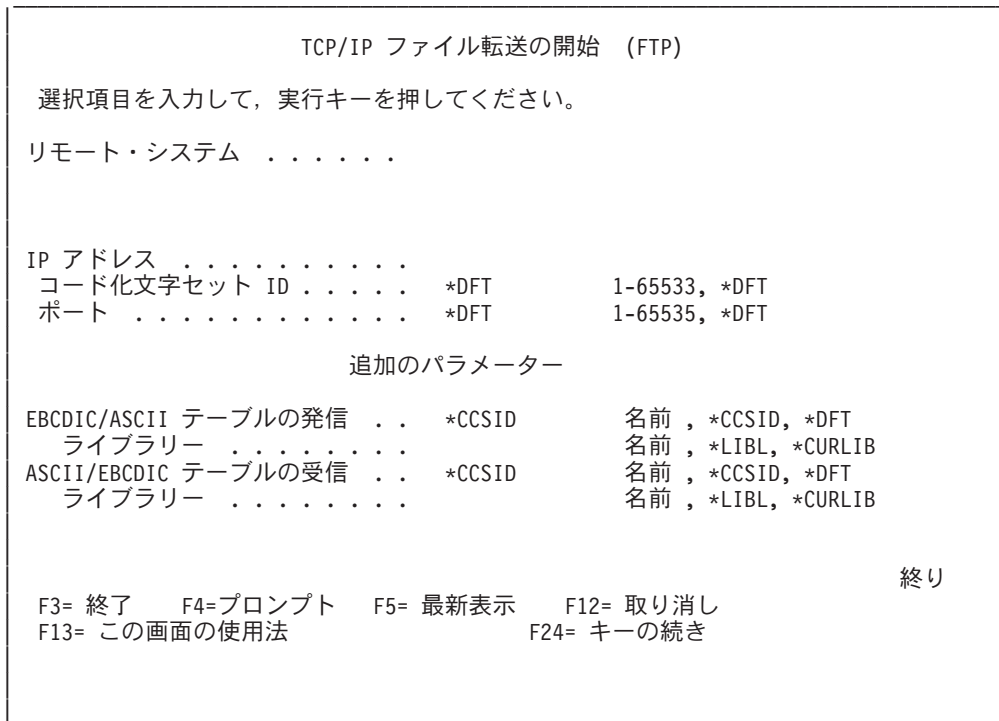


図 1. *CCSID 値で ASCII マッピング・テーブルを指定する

FTP クライアントが使用する CCSID (つまり、マッピング・テーブル) を指定します。*DFT 値が変更されない場合は、CCSID 値 00819 (ISO 8859-1 8 ビット ASCII) が使用されます。インバウンドの転送とアウトバウンドの転送の両方に特定の CCSID を指定することもできます。CCSID の使用については、『FTP における各国語サポートについての考慮事項』に説明があります。

- 注:
- 2 バイト文字セット (DBCS) CCSID 値は、CHGFTP コマンドの CCSID パラメーターでは使用できません。DBCS CCSID 値は、TYPE (ファイル転送タイプの指定) サブコマンドを使用して指定できます。
 - IBM では、V3R1 より前のリリースと確実に互換性を持たせるために、FTP でマッピングをサポートしています。着信する TYPE A のファイル転送にマッピング・テーブルを使用すると、ターゲット・ファイルを作成する必要がある場合には CCSID のタグ付けが損失します。IBM は、通常の操作では CCSID サポートを使用することを強くお勧めします。




iSeries ファイルの CCSID コード・ページのタグ付け: FTP が iSeries サーバーで新規ファイルを作成する場合、ファイル内の文字データを認識するために、そのファイルは CCSID または CCSID のコード・ページでタグ付けされます。データを既存のファイルと置き換える、または既存のファイルに追加する場合、そのファイルのタグは変更されません。以下の表に、FTP がこれらの値をさまざまなファイル・システムおよび転送タイプに割り当てる方法を要約します。

表 1: iSeries ファイルの CCSID コード・ページのタグ付け

ファイル・システムの受信	転送タイプ A (ASCII)	転送タイプ C (「ccsid」)	転送タイプ E (EBCDIC)	転送タイプ Type I (イメージ / バイナリー)
QSYS.LIB	EBCDIC コード化文字セット識別コードによって新規データベース・ファイル (CRTCCSID) 設定に指定される CCSID。	EBCDIC CCSID の場合、「ccsid」。CCSID が ASCII の場合、関連する省略時の EBCDIC CCSID。	65535	65535
“root”、QOpenSys、QDLS、QOPT	省略時の ASCII CCSID。	TYPE C ccid# サブコマンドに指定される「ccsid」値。	65535 でない場合、ジョブの CCSID。ジョブの CCSID が 65535 の場合、省略時のジョブの CCSID を割り当てる。	省略時の ASCII CCSID。
QLanSrv	ファイル・ディレクトリーのネットワーク・サーバー記述の ASCII コード・ページ	ファイル・ディレクトリーのネットワーク・サーバー記述の ASCII コード・ページ	サポートされていない。	ファイル・ディレクトリーのネットワーク・サーバー記述の ASCII コード・ページ

注: 省略時の ASCII CCSID は、FTP ジョブが開始するときに定義されます。クライアントでは、STRTCPTP (および FTP) コマンドの CCSID パラメーターになります。サーバーでは、CHGFTPA コマンドを使用して変更できる FTP 構成属性の CCSID パラメーターになります。QFileSvr.400 のファイル割り当ては、ファイルを受信するファイル・システムによって異なります。



FTP における各国語サポートについての考慮事項:  さまざまな主要言語を使用する環境で FTP を使用する場合、以下について注意してください。

- TYPE E (または EBCDIC) を使用してデータを転送する場合、データはそのまま保管され、送信元のファイルの EBCDIC コード・ページを使用します。このため、2 つの iSeries サーバーの主要言語が異なる場合、保管されるファイルは不適切な CCSID 値でタグ付けされる可能性があります。

たとえば、TYPE E を使用してコード・ページ 237 のデータが、ファイルが存在しないマシン上の QSYS.LIB ファイル・システムに送信された場合、データはそのまま CCSID 65535 でタグ付けされた新規ファイルに保管されます。受信する側のファイルがすでに存在する場合は、データはそのまま受信され、既存のファイルの CCSID (237 以外の場合もある) でタグ付けされます。

誤った CCSID でのタグ付けを回避するには、TYPE C CCSID サブコマンド (TYPE C 237 など) を使用して、転送するデータの CCSID を指定します。転送時に CCSID が指定されており、データが既存のファイルに書き込まれる場合、データは既存のファイルの CCSID に変換されます。転送前にターゲット・ファイルが存在しない場合は、ファイルが作成され、指定された CCSID でタグ付けされます。

前出の例でターゲット・ファイルが存在しないときには、受信側のシステムで、CCSID が 237 のファイルが作成されます。ターゲット・ファイルがすでに存在する場合には、データは CCSID 237 からターゲット・ファイルの CCSID に変換されます。

- FTP クライアントを開始するときに、メッセージ TCP3C14: データを CCSID &1 から CCSID &2 に変換できません (Unable to convert data from CCSID &1 to CCSID &2) が表示される場合があります。ジ

ジョブによって指定された EBCDIC CCSID とこの FTP セッションに指定された ASCII CCSID との間で文字変換が使用できない場合に、このメッセージが表示されます。

STRTCFTP CL コマンド のコード化文字セット識別コードのパラメーターに値を指定することで、ASCII CCSID を変更することができます。IBM パーソナル・コンピューター Latin-1 コード化文字セットを含む CCSID 850 は、ASCII CCSID であり、すべての有効なジョブの CCSID 値に対し、文字変換が使用できます。

- 2 つの EBCDIC システム間でモードを ASCII にして FTP を使用する場合、ファイルを送信する側のシステムのデータは、保管されている EBCDIC コード・ページから ASCII に変換され、受信側のシステムで ASCII から EBCDIC コード・ページに変換されます。通常は、これにより問題は起きません。送信側のシステムの EBCDIC 文字が ASCII コード・ページで定義されている限り、2 つのシステムが使用する 7 ビットの ASCII コード・ページが同じになるためです。また、一部の ASCII コード・ページの文字のマッピングが、2 つの EBCDIC コード・ページ間で異なる場合があります。これは、ASCII 文字の一部が可変 (文字が EBCDIC コード・ページで異なる 16 進コード・ポイントを占有する) である場合に起こる可能性があります。受信側のシステムの EBCDIC コード・ページがファイルを送信する側の EBCDIC コード・ページと異なる場合、可変文字の解釈が受信側のシステムで異なる場合があります。



ファイル・システムと命名規則

FTP サーバーは、ファイル・システムの情報単位を複数レベルのツリー構造に編成します。

FTP がサポートする OS/400 ファイル・システム は、ユーザーの iSeries サーバーのリリース・レベルによって異なります。OS/400 のファイル・システムは、データ、およびデータの階層グループ化に、異なる用語を使用することができます。

命名規則

OS/400 のファイル・システムは、それぞれ独自のファイル命名規則を持っています。ファイルの命名に使用する形式は、そのファイルが存在するファイル・システムの命名規則に従う必要があります。FTP がサポートする OS/400 ファイル・システムのファイル名の形式と例については、IFS で説明しています。詳細は、『ファイル・システムおよび管理』のトピックを参照してください。サーバーは、QUOTE HELP を使用すると、非 iSeries サーバーのファイルの命名情報を提供している場合があります。

FTP サーバー NAMEFMT

FTP サーバーのセッションが開始したときに、NAMEFMT は “0” に設定されます。NAMEFMT 値は、SITE サブコマンドを使用して変更できます。

以下のサブコマンドで受け取った「最初の」ファイルまたはパス名のパラメーターが以下のいずれかの場合、サーバーは、省略時の NAMEFMT 0 から NAMEFMT 1 に自動的に切り替わります。

- スラッシュ (/) 文字またはチルド (~) 文字で開始する
または
- ブランクである (LIST サブコマンドと NLST サブコマンド以外)

これに続く、ファイルまたはパス名のパラメーターを持つサーバー・サブコマンドは、NAMEFMT 値に影響を与えません。NAMEFMT の変更に加え、サーバーのサブコマンドに対する応答には、NAMEFMT 値が変更されたことを示すステートメントが含まれます。

たとえば、ファイルまたはパス名を持つ最初のサーバー・サブコマンドが以下の場合、サーバーの NAMEFMT 値は “1” に変更されます。

```
CWD /DIR1/DIR2A
```

サーバーの応答は、以下のようになります。

```
250-NAMEFMT set to 1.  
250 Current directory changed to /DIR1/DIR2A.
```

注: この機能によって、NAMEFMT 1 が必要な通常の Web ブラウザーが、SITE NAMEFMT 1 サブコマンドを実行せずに iSeries FTP サーバーと対話することができます。

NAMEFMT の詳細は、『ファイル・システムと命名規則』を参照してください。

ファイル名形式の作業で NAMEFMT サブコマンドを使用する方法については、NAMEFMT (ファイル命名形式の選択) クライアント・サブコマンドのページを参照してください。

FTP がサポートする OS/400 ファイル・システム

FTP がサポートするファイル・システムは、iSeries サーバーのリリース・レベルによって異なります。

QSYS.LIB ライブラリー・ファイル・システム - ライブラリー、ファイル、メンバー

FTP は、物理ファイル、論理ファイル、DDM ファイル、およびソース物理ファイルのメンバーと、保管ファイルの転送をサポートします。QSYS.LIB ファイル・システムの物理ファイルでは、転送されるデータは、ライブラリーに存在するファイルのメンバーです。

QDLS 文書ライブラリー・サービス - フォルダーおよび文書

文書ライブラリー・サービス (QDLS) ファイル・システムでは、転送されるデータは文書です。QDLS 文書は、フォルダーと呼ばれるディレクトリーに常駐しています。

“root”

/ ファイル・システム。このファイル・システムは統合ファイル・システムのストリーム・ファイル・サポートと階層ディレクトリー構造を十分に活用します。DOS および OS/2^R のファイル・システムの特徴を持っています。

QOpenSys

開放型システムのファイル・システム。このファイル・システムは、UNIX ベースの開放型システム標準 (POSIX、XPG など) と互換性があります。ルート・ファイル・システムと同様に、統合ファイル・システムに備わっているストリーム・ファイルおよびディレクトリー・サポートを活用します。名前の大文字小文字の区別をサポートします。

QOPT

QOPT 光ファイル・システム。このファイル・システムは、光メディアに保管されているストリーム・データへのアクセスを提供します。

QFileSvr.400

OS/400 ファイル・サーバー・ファイル・システム。このファイル・システムは、リモート iSeries サーバーにある他のファイル・システムへのアクセスを提供します。FTP は、QFileSvr.400 を使用する QSYS.LIB、QDLS、および QOPT へのアクセスをサポートしていません。

FTP がサポートするファイル・システムに関する包括的な情報は、『統合ファイル・システム』にあります。

FTP サーバーの応答状況メッセージ

FTP クライアント・セッション中にサブコマンドを入力すると、ディスプレイに 3 桁コード (xyz) の状況メッセージが戻されます。

最初の桁 (x) は、応答が正常であるか、異常であるか、不完全であることを示します。最初の桁には、次の 5 通りの値があります。

- 1yz = 正常。要求された処置を開始中です。別の応答が後に続きます。
- 2yz = 正常。要求された処置は正常に完了しました。新しい要求を開始できます。
- 3yz = 不完全。サブコマンドは受け入れられましたが、要求された処置はさらに情報を受信するまで保留中です。
- 4yz = 不完全。サーバーがサブコマンドを受け入れませんでした。要求された処置は実行されませんでした。エラーは一時的なものであり、処置を再度要求できます。
- 5yz = 異常。サブコマンドは受け入れられず、要求された処置は実行されませんでした。

2 番目の桁 (y) は、応答の機能カテゴリーを示します。

- x0z=構文。構文エラー、必要な操作に対して適切でないコマンド、および不必要なコマンドに関する応答です。
- x1z=情報。状況やヘルプなどの情報の要求に関する応答です。
- x2z=接続。制御またはデータ接続に関する応答です。
- x3z=認証。ログイン・プロセスに関する応答です。
- x5z=ファイル・システム。ファイル転送要求に関係するサーバーの状況に関する応答です。

3 番目の桁 (z) は、機能カテゴリーの詳細情報を示します。

一般的な応答コードとその意味を以下に示します。メッセージ・テキストは、サーバー・システムによって異なることがあります。

コード	意味
110	マーカー応答の再始動
120	サービスが nnn 分以内に作動可能
125	データ接続がすでにオープン。転送開始
150	ファイル正常開始。データ接続をオープン中
200	コマンドは OK
202	コマンドが組み込まれていない。このシステムでは使用できない
211	システム状況、またはシステム・ヘルプ応答
212	ディレクトリー状況
213	ファイル状況
214	ヘルプ・メッセージ
220	サービスが新規ユーザーに対して使用可能
226	データ接続のクローズ中。要求されたファイル処置は正常に完了
230	ユーザーはログインした
250	要求されたファイル処置は正常に完了した
257	パス名が作成された

コード	意味
331	パスワードが必要
332	アカウントが必要
425	データ接続がオープンできない
426	接続がクローズした。転送は異常終了した
450	要求されたファイル処置は実行されなかった。ファイル使用中
451	要求された処置は異常終了した。処理中にローカル・エラー
452	要求された処置は実行されなかった。システムの記憶容量不足
500	構文エラー。コマンドは認識されなかった
501	パラメーターまたは引き数の構文エラー
502	コマンドが組み込まれていない
503	コマンドの順序が正しくない
504	コマンドがこのパラメーター用に設定されていない
530	ログオンを試みたが拒否された
532	ファイルの保管のためにアカウントが必要
550	要求された処置は実行されなかった。ファイルが見つからなかったか、アクセス権がない
551	要求された処置は異常終了した。ページ・タイプが不明
552	要求されたファイル処置は異常終了した。割り当て記憶容量を超過している
553	要求された処置は実行されなかった。ファイル名が無効

FTP サーバーの構文規則

このトピックで説明する FTP サーバー・サブコマンドでは、次の構文規則を使用します。

大文字

大文字は、サブコマンドの構文定義に示されているとおりに入力する必要があります。文字の入力は、大文字でも小文字でもかまいません。

小文字の語またはハイフン付きの用語

小文字の語またはハイフン付きの用語 (たとえば、remotefile、account-information など) は、変数を表します。これは特定の情報で置き換える必要があります。

大括弧 []

大括弧に入れた語、記号、または句はオプションです。

左括弧 (とアスタリスク *

左括弧とアスタリスクは、構文定義に示されているとおりに入力する必要があります。

中括弧 { }

中括弧は、パラメーター、値、または変数のグループを示します。このグループは反復して指定できません。

省略記号 ...

省略記号は、その前の大括弧の中の変数を反復して指定できることを示します (指定しなくてもかまいません)。

縦線 |

パラメーターとパラメーターの間、または値と値の間の縦線は、どちらか 1 つを指定でき、同時に両方は指定できないことを示します。縦線は、大括弧または中括弧のセットの中で使用されます。

FTP クライアントの構文規則

このトピックで説明する FTP クライアント・サブコマンドでは、次の構文規則を使用します。

大文字

クライアント・サブコマンドの構文定義で大文字で印刷してある部分は、最小限入力しなければならない部分です。FTP クライアント・サブコマンドの入力は、大文字でも小文字でもかまいません。

小文字の語またはハイフン付きの用語

小文字の語またはハイフン付きの用語 (たとえば、*remotefile*、*account-information* など) は、変数を表します。これは特定の情報で置き換える必要があります。

大括弧 []

大括弧に入れた語、記号、または句はオプションです。

左括弧 (とアスタリスク *

左括弧とアスタリスクは、構文定義に示すとおりに入力する必要があります。

中括弧 { }

中括弧は、パラメーター、値、または変数のグループを示します。このグループは反復して指定できます。

省略記号 ...

省略記号は、その前の大括弧の中の変数を反復して指定できることを示します (指定しなくてもかまいません)。

縦線 |

パラメーターとパラメーターの間、または値と値の間の縦線は、どちらか 1 つを指定でき、同時に両方は指定できないことを示します。縦線は、大括弧または中括弧の中で使用されます。

詳細については、次を参照してください。

- サブコマンド・パラメーターの囲み方: [アポストロフィ \('\) または引用符 \(" \) のどちらかを使ってパラメーターを囲む方法へのリンク](#)です。
- クライアント転送サブコマンドのファイル名: [省略時値の詳細情報へのリンク](#)です。
- 転送するファイルの命名: [Localfile および Remotefile パラメーターの詳細情報へのリンク](#)です。

サブコマンド・パラメーターの囲み方

アポストロフィ (') または引用符 (") のどちらかを使って、サブコマンド・パラメーターを囲むことができます。パラメーター内にアポストロフィを入れる場合は、2 つのアポストロフィ (') を連続して入れ、パラメーターをアポストロフィで囲む必要があります。あるいは、パラメーター内に 1 つのアポストロフィを入力し、パラメーターを引用符 (") で囲む必要があります。

同様に、パラメーター内に引用符 (") を入れる場合は、次のいずれかの方法で引用符を入力する必要があります。

- パラメーター内に単一の引用符 (") を入れ、パラメーターをアポストロフィで囲む。
- パラメーター内に 2 つの引用符 ("") を連続して入れ、パラメーターを引用符で囲む。

アポストロフィまたは引用符は、次のように使用できます。

1. パラメーター内のアポストロフィまたは引用符が開始区切り文字および終了区切り文字と同じ場合は、パラメーター内の符号を繰り返します。たとえば、次のようになります。

```
'ABCD'12345'  
結果は ABCD'12345 になります。
```

```
"ABCD""12345"  
結果は ABCD"12345 になります。
```

2. 開始区切り文字および終了区切り文字がパラメーター内のアポストロフィまたは引用符と同じでない場合は、パラメーター内の符号は繰り返しません。たとえば、次のようになります。

```
"ABCD'12345"  
結果は ABCD'12345 になります。
```

```
'ABCD"12345'  
結果は ABCD"12345 になります。
```

3. パラメーター内にアポストロフィと引用符の両方を入れる場合は、1 つの符号を区切り文字として選択します。たとえば、次のようになります。

```
"ABC'12""345" または 'ABC'12"345'  
結果は ABC'12"345 になります。
```

詳細については、次を参照してください。

- FTP クライアントの構文規則

クライアント転送サブコマンドのファイル名

FTP クライアントは、PUT サブコマンド、APPEND サブコマンド、および GET サブコマンドのターゲット・ファイル名が省略されている場合に、省略時のファイル名を提供します。MPUT サブコマンドと MGET サブコマンドにはソース・ファイル名を指定できるので、FTP サーバーは MPUT および MGET に対してもターゲット・ファイル名を生成します。これらのサブコマンドの構文については、次のデータ転送サブコマンド表を参照してください。この表の「ターゲット」という列は、省略時名が提供されるパラメーターです。

サブコマンド	ソース	ターゲット	その他
APPEND	local filename	[server filename]	
PUT	local filename	[server filename]	
GET	server filename	[local file name]	[(Replace)]
MPUT	local filename		
MGET	server filename		[(Replace)]

PUT および APPEND

PUT サブコマンドおよび APPEND サブコマンドでは、省略時名を形成する規則は次の 2 つの場合に類別されます。

- iSeries サーバーの場合
 - ターゲット・ファイル・システムがライブラリー・ファイル・システムか文書ライブラリー・システムである場合、省略時名は、これらのファイル・システムの命名規則 (名前形式を含む) に従います。

- ターゲット・ファイル・システムがライブラリー・ファイル・システムでも文書ファイル・システムでもない場合、省略時名は次のいずれかになります。
 - ソース・ファイル名の最後のスラッシュの後の名前が省略時名になります。
 - スラッシュがない場合、省略時名はソース・ファイル名と同じ になります。
- iSeries サーバーでない場合
 - ソース・ファイルがライブラリー・ファイル・システムのファイルの場合、省略時名の構成は、ファイル名.メンバー名 となります。メンバー名がない場合は、ファイル名が省略時名になります。
 - ソース・ファイルが文書ライブラリー・サービスのファイルの場合、省略時名は、拡張子付きのファイル名です。
 - ソース・ファイルがライブラリー・ファイル・システムでも文書ライブラリー・サービス・ファイルでもない場合、ソース名の最後のスラッシュの後の名前が省略時名になります。スラッシュがない場合は、省略時名はソース名と同じです。

サーバーが iSeries サーバーの場合、これらのサブコマンドでは、PUT サブコマンドに適用されるのと同じ規則を使用して、省略時名が生成されます。

GET および MGET

サーバーが iSeries サーバーでない場合、GET サブコマンドと MGET サブコマンドで使用される省略時名は、ソース名の最後のスラッシュの後の部分に基づいて生成されます。スラッシュがない場合は、ソース名全体が省略時名となります。省略時名の形成の規則は次のとおりです。

- クライアント・ファイル・システムがライブラリー・ファイル・システム (iSeries データベース) である場合、次の規則が適用されます。
 - リモート・ファイル名がピリオド (.) を含む場合、ピリオドの前の文字は 10 文字に切り縮められ、この 10 文字がローカル・ファイル名となります。ピリオドの後の文字は 10 文字に切り縮められ、この 10 文字がメンバー名となります。
 - リモート・ファイル名にピリオドが含まれていない場合、ローカル・ファイル名を形成するファイル名とメンバー名は、どちらもリモート・ファイル名を 10 文字に切り縮めたものになります。
 - 名前形式が 1 の場合、サーバーが名前のファイル部分とメンバー部分に適切な拡張子を追加します。
- クライアント・ファイル・システムが文書ライブラリー・サービス の場合、次の規則が適用されます。
 - リモート名がピリオドを含む場合、ピリオドの前の文字は 8 文字に切り縮められます。ピリオドの後の文字は 3 文字に切り縮められます。
 - リモート名がピリオドを含まない場合、名前は拡張子なしの 8 文字に切り縮められます。
- 他のファイル・システムの場合、リモート名の最後のスラッシュの後の名前が省略時名になります。

- 注:
1. 保管ファイルにはメンバーがありません。したがって、保管ファイルの省略時名にはメンバー部分がありません。
 2. DEBUG モードがオンに設定されている場合には、サーバーが省略時名を表示します。

詳細については、次を参照してください。

- FTP クライアントの構文規則

転送するファイルの命名

データ転送に使用する FTP クライアント・サブコマンドは、**localfile** パラメーターと **remotefile** パラメーターのいずれか、または両方を取ることができます。これらのパラメーターを使用して、転送するデータに名前を付けることができます。転送サブコマンドは次のとおりです。

APPEND
localfile [remotefile]

DELETE
remotefile

GET
remotefile [localfile]

MDELETE
remotefiles

MGET
remotefiles

MPUT
localfiles

PUT
localfile [remotefile]

localfile パラメーターと remotefile パラメーターで指定する名前は、部分修飾名か完全修飾名のいずれかになります。部分修飾名には、データ自身の名前と、階層序列においてデータより上位にある 1 つまたは複数の名前が含まれます。完全修飾名には、階層序列においてデータより上位にあるすべての名前が含まれます。

名前が部分的に修飾されている場合、現行作業ディレクトリーによって、処理するファイルが識別されません。ローカル・クライアント・システムの作業ディレクトリーは、LCD サブコマンドを使用して設定できます。リモート・サーバー・システムの作業ディレクトリーは、CD サブコマンドを使用して設定できます。

ローカル・ファイル名のパラメーターの形式は、iSeries のファイル命名規則に合致しなければなりません。リモート・ファイル名は、リモート・システムのファイル命名規則に合致しなければなりません。

詳細については、次を参照してください。

- サブコマンド・パラメーターの囲み方: アポストロフィ (') または引用符 (") のどちらかを使って、パラメーターを囲むことができます。
- クライアント転送サブコマンドのファイル名: クライアント転送サブコマンドの省略時ファイル名に関する情報へのリンクです。
- FTP クライアントの構文規則: FTP クライアント・サブコマンドでは、これらの構文規則を使用します。

FTP のトラブルシューティング

このトピックでは、FTP のトラブルシューティングに関する基本的な情報が記載されています。

FTP についての問題の判別

ステップのリストを参照して、SMTP が正しく機能しているかどうか判別します。

FTP の問題報告に必要な資料

このトピックでは、サービス技術員が必要とする情報を説明しています。

FTP サーバーのトレース

DEBUG サブコマンドを使用して FTP サーバーの問題をトレースします。

FTP クライアントのトレース

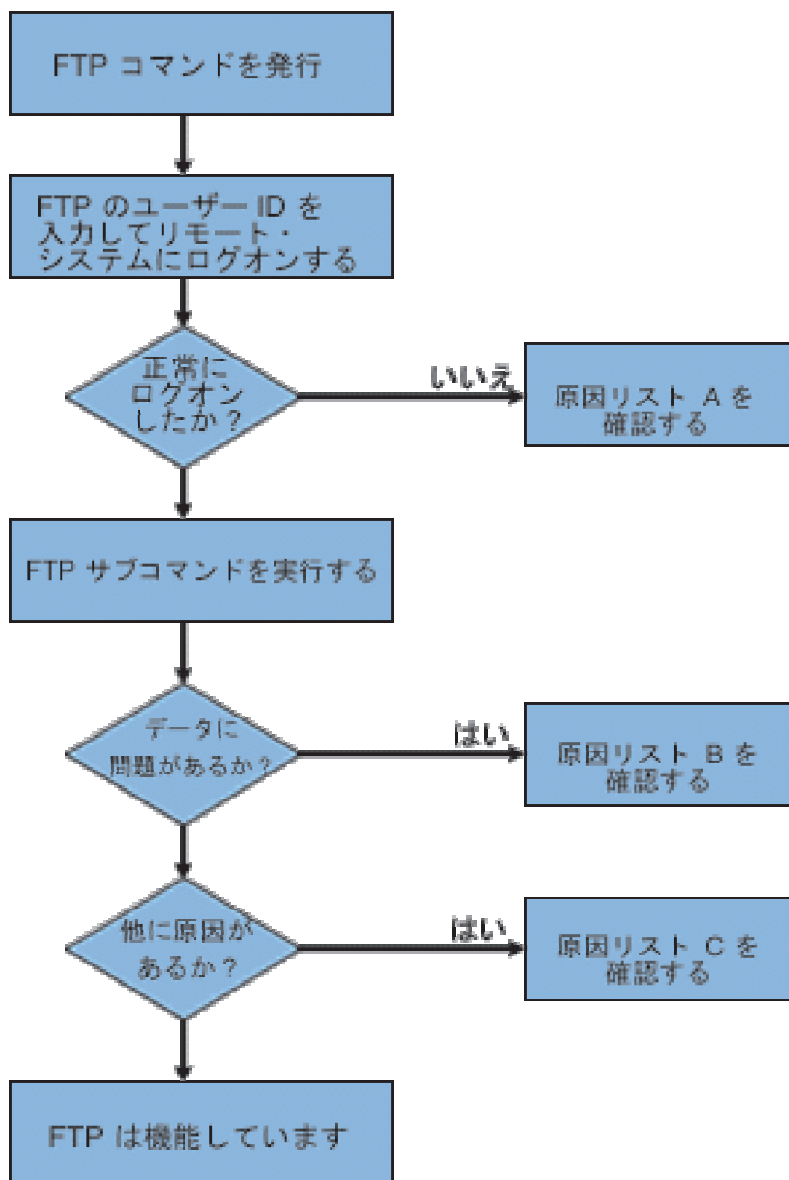
DEBUG サブコマンドを使用して FTP クライアントの問題をトレースします。

FTP サーバー・ジョブおよびジョブ・ログでの作業

FTP サーバー・ジョブ・ログのスパール・ファイルを取得して、エラーを調査します。

FTP についての問題の判別

FTP の使用中に問題を検出した場合には、以下のフローチャートを使用して、TCP/IP の一般的な問題を確認し、その後、FTP の問題の原因を判別します。次の原因リストは、問題の原因の判別に役立つステップをリストしたものです。



*

FTP 問題分析

原因リスト A

1. iSeries FTP サーバーへの接続と、ユーザー ID のプロンプトを受信するまでの間に、長い遅延はありますか? 長い遅延がある場合には、ユーザーの iSeries のドメイン・ネーム・サーバーの構成を検査します。FTP サーバーは、新規接続を受信すると、即座に DNS 照会を実行します。DNS 問題は、応答を受信するまで、サーバーを数分ハングさせることがあります。
2. 出口プログラムが FTP Server Logon Exit Point に追加されたかどうかを確認します。『サーバー・ログオン出口点』のサブトピックを参照してください。追加されている場合には、失敗したログオンが、その出口プログラムに許可されているかどうかを確認します。
3. パスワードが必要な場合には、リモート・ログインがパスワードを必要としているかどうかを検査します。システムによってはパスワードが尋ねられますが、実際には必要でない場合、接続は失敗することがあります。

4. リモート・システムにパスワードが必要であれば、パスワードを設定します。システムのセキュリティ情報を変更すると、システムの再始動が必要なことがあります。
5. リモート・システムへのサインオンを試行して、ユーザー ID およびパスワードを検査します。サインオンできない場合は、システムの所有者に、そのユーザー ID およびパスワードが正しいかどうかを確認します。

原因リスト B

1. バイナリー・ファイルを転送している場合には、バイナリー・モードが有効になっていることを確認します。
2. クライアント・システムおよびサーバー・システムのマッピング・テーブルに、互換性があることを確認します。これは、ユーザーが固有のマッピング・テーブルを使用している場合のみ必要です。
3. その転送に、正しい CCSID が指定されているかどうかを確認します。正しい CCSID が指定されていない場合には、TYPE または LTYPE サブコマンドを使用して、転送実行前に、正しい CCSID 値を設定します。
4. データを保管する予定のシステムに、ファイルを作成します。適切なレコード長、メンバー数、および増分数を設定します。データ転送をもう一度試行して、正常終了することを確認します。
5. そのファイルおよびファイル・メンバーの使用を、ユーザーが許可されていることを確認します。
6. 転送ファイルがパック 10 進データ、またはゾーン 10 進数データを含んでいるかどうかを確認します。
7. 保管ファイルを転送している場合には、適切な方式が使用されていたかを確認します。

原因リスト C

1. リモート・システムのファイル・サイズ制限を確認します。
2. FTP サーバーのタイマーが終了したかどうかを確認します。iSeries サーバーのタイムアウト値は、QUOTE TIME コマンドを使用して設定することができます。
3. NETSTAT コマンドを使用して、*LOOPBACK インターフェースが活動状態になっていることを確認します。次に、FTP LOOPBACK (iSeries 間内部) を実行して、問題を再現します。
 - 問題の再現ができない場合には、リモート・システムに問題がある可能性があります。
 - 問題を再現できた場合には、以下を行います。
 - a. 問題が FTP サーバーの問題の場合には、FTP サーバーを開始して、TRCTCPAPP コマンドを使用してトレースします。
 - b. 問題を再現します。
 - c. FTP 接続を終了します。『FTP サーバーの始動および停止』を参照してください。
 - d. TRCTCPAPP コマンドを使用して、FTP サーバー・トレースを終了します。
 - e. 以下の属性で、スプール・ファイルを検索します。
 - ファイル名は QTMFFTRC です
 - このファイルに関連付けられているユーザー名は、TRCTCPAPP コマンドを発行したユーザーの名前です。トレースは、その FTP サーバー・ジョブに関連付けられたシステムの、省略時出力待ち行列内のスプール・ファイルです。
 - f. トレースをスプール・ファイルに送信します。
 - g. 問題が iSeries FTP クライアント上で発生している場合には、トレースは DEBUG 100 クライアント・サブコマンドを使用して取得することができます。

- h. FTP クライアントを対話式に実行している場合には、F6 (印刷) キーを使用して、入力された FTP クライアント・サブコマンドの履歴と、関連付けされている FTP サーバーの応答履歴を含むスプール・ファイルを作成します。バッチの不在モードで FTP クライアントが実行されている場合には、このサブコマンドとサーバー応答の履歴は、指定された出力ファイルに書き込まれます。詳細については、『FTP のバッチ・ジョブ』を参照してください。

FTP の問題報告に必要な資料

FTP 問題を IBM に報告される場合は、以下のものを提供してください。

- 障害発生以降の通信トレース (TCP/IP データの要求のみ)。このトレースは、ASCII と EBCDIC の 2 つの形式で作成してください。
- FTP クライアントまたはサーバーがソフトウェア・エラーをログに記録していた場合は、そのデータを提供してください。

注: ソフトウェア・エラーをログに記録するには、システム値 QSFWERRLOG を *LOG に設定しておかなければなりません。QSFWERRLOG を *NOLOG に設定していてエラーが発生した場合は、その値を *LOG に変更して、エラーを再発生させ、ログに記録されたソフトウェア・エラー・データを提出してください。ログに記録されたソフトウェア・エラー・データの提供があれば、FTP のトレースを実行する必要はありません。

- QTCPIP と任意の FTP サーバーまたは FTP クライアント・ジョブのログ。
- FTP クライアントおよび FTP サーバーのデバッグ・トレース。
- FTP クライアント問題の場合は、FTP クライアント・セッションを含むスプール・ファイル (このスプール・ファイルを取得するには、FTP セッションでプリント (F6) キーを押します)。
- データ安全性に問題がある場合は、エラーの原因となったファイル、メンバー、またはライブラリーを、そのファイル、メンバー、またはライブラリーの説明のコピーと共にお送りください。

FTP サーバーのトレース

FTP サーバーは、TCP/IP を実行している iSeries または iSeries 以外の任意のサーバーからトレースすることができます。FTP サーバーをトレースするには、2 つの方法があります。FTP サーバーの DBUG サブコマンドは、FTP サーバー・セッション内でトレースを実行します。TCP/IP 適用業務のトレース (TRCTCPAPP) コマンドを使用すると、システム全体にわたってすべての FTP サーバーをトレースすることができます。

DEBUG サブコマンドを使用した FTP サーバーのトレース

以下に示す例では、FTP サーバーの DEBUG サブコマンドを使用しています。

```
                                ファイル転送プロトコル

前の FTP サブコマンドおよびメッセージ:
  ポート 21 を使用してアドレス N.NNN.NN.NNN でホスト
  XXXXXNNN.XXXXXXXXXX.XXX.XXX と接続中である。

220-QTCP AT XXXXXNNN.XXXXXXXXXX.XXX.XXX.
220 CONNECTION WILL CLOSE IF IDLE MORE THAN 5 MINUTES.
> TEST
331 ENTER PASSWORD.
230 TEST LOGGED ON.
   OS/400 IS THE REMOTE OPERATING SYSTEM. THE TCP/IP VERSION IS "V4R4M0".
250 NOW USING NAMING FORMAT "0".
257 "QGPL" IS CURRENT LIBRARY.

FTP サブコマンドを入力してください。
====> QUOTE DEBUG

F3= 終了      F6= 印刷
F17= 最上部   F18= 最下部   F21= CL コマンド入力行
```

FTP サーバーをトレースするには、次のようにします。

1. QUOTE DEBUG と入力して、トレースを開始します。

```
                                ファイル転送プロトコル

前の FTP サブコマンドおよびメッセージ:
  ポート 21 を使用してアドレス N.NNN.NN.NNN でホスト
  XXXXXNNN.XXXXXXXXXX.XXX.XXX と接続中である。

220-QTCP AT XXXXXNNN.XXXXXXXXXX.XXX.XXX.
220 CONNECTION WILL CLOSE IF IDLE MORE THAN 5 MINUTES.
> TEST
331 ENTER PASSWORD.
230 TEST LOGGED ON.
   OS/400 IS THE REMOTE OPERATING SYSTEM. THE TCP/IP VERSION IS "V4R4M0".
250 NOW USING NAMING FORMAT "0".
257 "QGPL" IS CURRENT LIBRARY.
> QUOTE DEBUG
250 DEBUG MODE IS NOW ON.
FTP サブコマンドを入力してください。
====> QUOTE DEBUG

F3= 終了      F6= 印刷
F17= 最上部   F18= 最下部   F21= CL コマンド入力行
```

2. トレースする FTP 操作を実行します。

3. もう一度 QUOTE DEBUG と入力して、トレースを終了します。このトレースにより、QTMFFTRC というスプール・ファイルが作成されます。省略時の出力待ち行列には、このスプール・ファイルが含まれます。ユーザーは常に、トレースの終了時に FTP にログオンしていたユーザーの名前です。
4. QUIT と入力して、FTP セッションを終了します。
5. 次のコマンドを入力して、出力待ち行列を見つけます。
 DSPSYSVAL QPRTDEV
 たとえば、以下のように表示されます。

```

                                システム値の表示
システム値 . . . . . : QPRTDEV
記述 . . . . . : 印刷装置記述
印刷装置 . . . . . : PRT01      名前

```

プリンター装置も省略時のシステム出力待ち行列の名前になります。

6. プリンター装置の名前を記録します。この例では、PRT01 がプリンター装置です。
7. F12 (取り消し) を押して、DSPSYSVAL コマンドを入力した表示画面に戻ります。
8. 次のコマンドを入力します。
 WRKOUTQ OUTQ(printer-device)
 printer-device を、前の画面で記録したプリンター装置に置き換えます。この例では、PRT01 が出力待ち行列です。たとえば、以下のように表示されます。

```

                                出力待ち行列処理
待ち行列 : PRT01      ライブラリー : QUSRSYS      状況 : RLS
オプションを入力して、実行キーを押してください。
  1= 送信  2= 変更  3= 保留  4= 削除  5= 表示  6= 解放  7= メッセージ
  8= 属性  9= 印刷状況の処理
OPT   ファイル   ユーザー   ユーザー・データ   STS   ページ   部数   用紙タイプ   PTY
QTCPPRT   QTCP     QTMSMTP     HLD     46     1   *STD     5
QTMFFTRC   QSECOFR     HLD     44     1   *STD     5

```

9. 画面に「続く...」が表示されたら、F18 (最下部) を押して、スプール・ファイル・リストの一番下に移動します。
10. トレースの作成時に FTP にログオンしていたユーザーと同じユーザーを持つ、QTMFFTRC という名前の最後のファイルを見つけてください。
11. F11 (ビュー 2) を押して、操作するファイルの日付と時刻を表示します。
12. 最新のスプール・ファイル QTMFFTRC であることを確認します。

問題報告書には、トレースを実行して失敗したことを記載してください。問題報告書と共に、トレース情報をすべて送ってください。

TCP/IP 適用業務のトレース (TRCTCPAPP) コマンドを使用した FTP サーバーのトレース

TCP/IP 適用業務のトレース (TRCTCPAPP) コマンド (V4R4 の新機能) を使用すると、システム全体に渡ってすべての FTP サーバーをトレースすることができます。

TRCTCPAPP コマンドは、特に訓練を積んだサービス担当員ならびに開発者用に提供されているものです。このコマンドを使用するには、*SERVICE 特殊権限が必要です。TRCTCPAPP は、サービスおよび開

発で使用するためにトレース・データを取り込む必要がある場合に使用してください。経験のある者がこのコマンドを使用すれば、アプリケーションのトレースの開始と停止を動的に実行することができます。

TRCTCPAPP を使用すると、次のように、FTP TCP/IP アプリケーション用にトレース情報を取り込むことができます。

- 内部トレース情報は、iSeries FTP サーバー用に取り込むことができます。FTP サーバー用に取り込むことのできる情報は、リモート IP アドレスとポート、あるいは iSeries ユーザー・プロファイルを使用して、フィルター操作することができます。システムで一度に活動状態にできるのは、1 つのトレースだけです。

以下に TRCTCPAPP コマンドを使用した例を 2 つ示します。

例 1:

```
TRCTCPAPP APP(*FTP) SET(*ON)
```

このコマンドを実行すると、すべての FTP サーバーのトレースを開始します。他の TCP アプリケーションのトレースにはまったく影響しません。

例 2:

```
TRCTCPAPP APP(*FTP) SET(*CHK)
```

このコマンドは、FTP サーバーの 1 つまたは複数のジョブのトレースの状況をチェックするために使用します。最後に次のコマンドを入力したと想定します。

```
TRCTCPAPP APP(*FTP) SET(*ON) USER(JOECOOL)
```

このコマンドに対する応答のフォーマットは、以下のようなメッセージのセットになります。

```
TCP45B7 TRCTCPAPP APP(*FTP) SET(*ON) USER(JOECOOL)
      MAXSTG(*DFT) TRCFULL(*WRAP)
TCP45B1 Tracing active for *FTP.
TCP45B2 Data capture begun for *FTP.
TCP45B3 Data buffer wrapped for *FTP.
```

FTP クライアントのトレース

FTP クライアントのトレースを作成する、または FTP サーバーに送信されたサブコマンドを表示するには、FTP クライアントの DEBUG サブコマンドを使用します。DEBUG サブコマンドは、デバッグ・モードを切り替えます。オプションにデバッグ値が指定されている場合は、この値を使ってデバッグ・レベルが設定されます。デバッグがオンになっている場合は、サーバーに送信された各サブコマンドは、先頭に文字列 '>>>' が付加されて表示されます。FTP クライアントのトレースを作成するには、デバッグ値を 100 に設定する必要があります。

```
DEBbug [debug value]
```

debug value

DEBUG 値が 0 の場合、DEBUG 機能はオフです。DEBUG 値が正の整数の場合、DEBUG 機能はオンです。

値を指定しない場合は、デバッグ値はゼロから 1 に切り替えられるか、正整数からゼロに切り替えられます。

100

FTP クライアントのトレースを開始します。クライアントは、DEBUG がオフになるまで、または FTP クライアントが終了するまで、トレースを実行し続けます。(トレースが終了しても、トレース・データをフォーマットする間、かなりの時間がかかることがあります。)

注: FTP クライアントのトレースは、IBM にソフトウェア問題を報告する場合にのみ使用してください。この機能によって、システム・パフォーマンスに悪い影響が出る場合があります。

V4R4 では、デバッグのための新しい機能が FTP クライアントに追加されました。この機能は、上述の DEBUG 100 に類似したものです。クライアントは、開始すると最初に QTMFTPD100 という名前のデータ域が存在しているかどうかをチェックします。

このコマンドを次のように使用して、QTEMP ライブラリーに QTMFTPD100 を作成する必要があります。

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

QTMFTPD100 データ域が存在する場合は、DEBUG 値が 100 に設定され、FTP クライアントのトレースが開始されます。この機能の目的は、DEBUG 100 サブコマンドを発行しても FTP クライアントのトレースを開始できない場合に、FTP クライアントのデバッグ・トレースを使用可能にすることです。

FTP サーバー・ジョブおよびジョブ・ログでの作業

FTP サーバー上で発生するエラーについて、詳細な情報を取得するには、FTP サーバー・ジョブ・ログのコピーが必要な場合があります。FTP サーバーは、エラーが発生して終了したときに、スプール・ファイルにサーバー・ジョブ・ログを自動的に書き込みます。

以下のサブコマンドを FTP クライアントから実行することによって、サーバーを終了せずにサーバー・ジョブ・ログをスプール・ファイルに書き込むことができます。

```
QUOTE RCMD DSPJOBLOG
```

サーバー・ジョブ・ログに書き込まれたエラー・メッセージのコピーを取得するには、エラー発生後にこのサブコマンドを発行する必要があります。その後で、WRKSPLF コマンドを使用してジョブ・ログの検査ができます。

この技法は、サーバーからクライアントに戻された応答メッセージが、サーバー・マシンで発生したエラーの情報をほとんど含んでいない場合に推奨されます。たとえば、この方式は、サーバー・マシンで発生する I/O エラーの詳細を取得するのに有効です。

エラーによってここで説明する方法で FTP サーバー・ジョブ・ログが取得できない場合には、次のコマンドを入力して、それぞれの FTP セッションに対してスプール・ジョブ・ログを強制的に作成します。

```
CHGJOB JOB(QUSRSYS/QTMFTPS) LOG(4 00 *SECLVL)
```

次に、そのエラーを発生させたシナリオを再現します。必要なデータを取得後に、元のジョブ・ログの振り舞いを復元するには、次のコマンドを入力します。

```
CHGJOB JOB(QUSRSYS/QTMFTPS) LOG(4 00 *NOLOG)
```

各 FTP セッションの終了時および FTP サーバーの終了時 (エラーの有無に関係なく) にスプール・ジョブ・ログを作成するには、ジョブ記述変更 (CHGJOB) コマンドを以下のように使用します。

```
CHGJOB JOB(QUSRSYS/QTMFTPS) LOG(4 00 *SECLVL)
```

サーバーの終了時にのみスプール・ジョブ・ログを取得するには、CHGJOB コマンドを以下のように使用します。

```
CHGJOB JOB(QUSRSYS/QTMFTPS) LOG(4 00 *NOLIST)
```

FTP サーバー・ジョブおよびジョブ名

FTP AUTOSTART パラメーターを *YES に設定して STRTCP コマンドを実行するか、または SERVER パラメーターを *FTP か *ALL に設定して STRTCPSVR コマンドを実行すると、FTP サーバー・ジョブが開始します。これらのジョブは QSYSWRK サブシステムで実行され、これらのジョブの目的は着信 FTP ユーザーのモニターです。これらのジョブ名の形式は、QTFTPnnnnn です。nnnnn は、このサーバーに実行依頼する FTP サーバー・ジョブのジョブ番号です。

FTP サーバー・ジョブを処理するには、以下の CL コマンドを入力します。

```
WRKACTJOB JOB(QTFTP*)
```




Printed in Japan