

IBM

@server

iSeries

FTP

*Verze 5, vydání 3*







@server

iSeries

FTP

*Verze 5, vydání 3*

**Poznámka**

Před použitím těchto informací a odpovídajícího produktu si přečtěte informace v části “Upozornění”, na stránce 149.

**Šesté vydání (srpen 2005)**

Toto vydání se týká verze 5, vydání 3, modifikace 0 licenčního programu IBM Operating System/400 (číslo produktu 5722-SS1) a všech následujících vydání a modifikací, dokud nebude v nových vydáních uvedeno jinak. Tato verze nefunguje na všech modelech počítačů RISC (reduced instruction set computer) ani na modelech CISC.

© Copyright International Business Machines Corporation 1998, 2005. Všechna práva vyhrazena.

# Obsah

<b>FTP</b> . . . . .	1
Tisk tohoto tématu. . . . .	1
Scénáře FTP . . . . .	2
Scénář: Přenos souboru ze vzdáleného hostitele . . . . .	2
Scénář: Zabezpečení FTP prostřednictvím SSL . . . . .	3
Podrobnosti ke konfiguraci. . . . .	4
Konfigurace FTP serveru . . . . .	8
FTP server v produktu iSeries <sup>(TM)</sup> Navigator . . . . .	8
Konfigurace FTP serverů pro grafické FTP klienty a nástroje pro Web . . . . .	9
Konfigurace anonymního FTP . . . . .	9
Příprava pro anonymní FTP. . . . .	10
Vytvoření programů výstupních bodů pro anonymní FTP . . . . .	11
Vytvoření uživatelského profilu OS/400 <sup>(R)</sup> : ANONYMOUS . . . . .	12
Vytvoření veřejné knihovny nebo adresáře . . . . .	12
Instalace a registrace programů výstupních bodů . . . . .	12
Zabezpečení FTP . . . . .	13
Zabránění přístupu k FTP serveru . . . . .	13
Řízení přístupu prostřednictvím FTP . . . . .	15
Zabezpečení FTP serveru pomocí SSL . . . . .	16
Vytvoření lokálního vydavatele certifikátů (CA) . . . . .	16
Asociace certifikátu s FTP serverem . . . . .	18
Vyžádání autentizace klienta pro FTP server (volitelné) . . . . .	18
Povolení SSL na FTP serveru . . . . .	19
Zabezpečení FTP klienta pomocí TLS/SSL . . . . .	19
Správa přístupu pomocí programů výstupních bodů FTP . . . . .	21
Správa přístupu pomocí produktu iSeries <sup>(TM)</sup> Navigator . . . . .	21
Monitorování přichozích uživatelů FTP . . . . .	22
Správa FTP serveru . . . . .	22
Spuštění a zastavení FTP serveru . . . . .	22
Nastavení počtu dostupných FTP serverů . . . . .	23
Zvýšení výkonu FTP serveru podporou konfigurovatelného podsystému . . . . .	23
Používání FTP klienta u serveru iSeries <sup>(TM)</sup> . . . . .	23
Spuštění a ukončení relace klienta . . . . .	24
Pokyny pro časovou prodlevu serveru . . . . .	28
Přenos souborů prostřednictvím FTP . . . . .	28
FTP jako dávková úloha . . . . .	29
Referenční informace o FTP . . . . .	37
Podpříkazy FTP serveru . . . . .	37
ADDM (Přidat člen fyzického souboru) . . . . .	39
ADDV (Přidat člen fyzického souboru s proměnnou délkou) . . . . .	39
APPE (Přidat k existujícímu souboru) . . . . .	40
AUTH (Oprávnění) . . . . .	40
CRTL (Vytvořit knihovnu) . . . . .	41
CRTP (Vytvořit fyzický soubor) . . . . .	41
CRTS (Vytvořit zdrojový fyzický soubor) . . . . .	41
CWD (Změnit pracovní adresář nebo knihovnu) . . . . .	41
DEBUG (Zapnout trasování FTP serveru) . . . . .	42
DELE (Vymazat soubor nebo dokument) . . . . .	42
DLTF (Vymazat soubor) . . . . .	42
DLTL (Vymazat knihovnu) . . . . .	43
HELP (Získání nápovědy ze vzdáleného serveru iSeries <sup>(TM)</sup> ) . . . . .	43
LIST (Seznam souborů) . . . . .	43
MKD (Vytvořit adresář) . . . . .	44

MODE (Nastavit režim přenosu)	44
NLST (Seznam jmen)	44
NOOP (Získat odezvu serveru)	45
PASS (Heslo)	45
PASV (Použit pasivní datové připojení)	45
PBSZ (Velikost ochranné vyrovnávací paměti)	45
PORT (Datový port)	46
PROT (Úroveň ochrany datového kanálu)	46
PWD (Zobrazit pracovní adresář nebo knihovnu)	47
QUIT (Ukončit relaci FTP serveru)	47
RCMD (Odeslat CL příkaz do systému FTP serveru)	47
REIN (Znovu inicializovat relaci mezi systémy)	48
RETR (Načíst soubor)	48
RMD (Odstranit adresář)	48
RNFR (Přejmenovat z)	48
RNTO (Přejmenovat na)	49
SITE (Odeslat informace používané systémem serveru)	49
STOR (Uložit soubor)	51
STOU (Uložit jedinečný soubor)	51
STRU (Zadat členění souboru)	52
SYST (Určit jméno operačního systému)	52
TIME (Nastavit hodnoty časové prodlevy pro FTP server)	53
TYPE (Zadat typ reprezentace dat)	53
USER (Odeslat uživatelské přihlašovací ID na server)	55
Podpříkazy FTP klienta	55
ACCT (Odeslat informace o účtu)	57
APPEND (Přidat člen lokálního souboru do vzdáleného souboru)	57
ASCII (Změnit typ souboru na ASCII)	58
BINARY (Nastavit typ přenosu na obraz)	58
CD (Změnit pracovní adresář nebo knihovnu)	59
CLOSE (Ukončit relaci FTP se vzdáleným systémem)	59
DEBUG (Vytvořit záznam z trasování klienta a řídit zobrazení podpříkazů serveru posílaných do vzdáleného systému)	60
DEBUG (Změnit hodnoty časového limitu klienta)	60
DELETE (Vymazat soubor ve vzdáleném systému)	61
DIR (Vypsát seznam položek adresáře, knihoven nebo souborů)	61
EBCDIC (Změnit typ souboru na EBCDIC)	62
GET (Zkopírovat soubor ze vzdáleného systému do lokálního systému)	62
HELP (Získání nápovědy k podpříkazům FTP)	63
LCD (Změnit pracovní knihovnu nebo adresář v lokálním systému)	64
LOCSITE (Zadat informace pro lokální systém)	64
LOCSTAT (Zobrazit informace o lokálním stavu)	66
LS (Vypsát seznam jmen vzdálených souborů)	66
LTYPE (Lokální typ)	67
MDELETE (Vymazat více souborů ze vzdáleného systému)	67
MGET (Zkopírovat více souborů ze vzdáleného systému do lokálního systému)	68
MKDIR (Vytvořit adresář)	69
MODE (Zadat režim přenosu dat)	69
MPUT (Odeslat více členů souboru z lokálního systému do vzdáleného systému)	70
NAMEFMT (Vybrat formát pojmenování souboru)	70
NULLFLDS (Povolit přenos souborů s poli NULL)	71
OPEN (Navázat spojení s FTP serverem ve vzdáleném systému)	72
PASS (Odeslat heslo)	72
PUT (Zkopírovat člen souboru z lokálního systému do souboru ve vzdáleném systému)	72
PWD (Zobrazit aktuální adresář, pořadač nebo knihovnu)	73
QUOTE (Odeslat podpříkaz na FTP server)	73

REINITIALIZE (Znovu nastavit relaci mezi systémy) . . . . .	74
RENAME (Přejmenovat soubor ve vzdáleném systému) . . . . .	75
RESET (Obnovit) . . . . .	75
RMDIR (Odstranit adresář) . . . . .	75
SECData (Nastavit zabezpečení ochrany dat) . . . . .	76
SECOpen (Nastavení zabezpečení ochrany dat) . . . . .	76
SENDPASV (Určit, zda se má odeslat podpříkaz PASV) . . . . .	77
SENDPORT (Určit, zda se má odeslat podpříkaz PORT) . . . . .	78
SENDSITE (Určit, zda se má odeslat podpříkaz SITE) . . . . .	79
SITE (Odeslat informace používané vzdáleným systémem) . . . . .	79
STATUS (Získat informace o stavu ze vzdáleného systému) . . . . .	80
STRUCT (Zadat členění souboru) . . . . .	80
SUNIQUE (Řídit přepisování souborů) . . . . .	81
SYSCMD (Předat CL příkaz iSeries <sup>(TM)</sup> lokálnímu systému iSeries) . . . . .	81
TYPE (Zadat typ přenosu souboru) . . . . .	82
USER (Odeslat vaše uživatelské ID do vzdáleného systému). . . . .	84
VERBOSE (Řídit textové zobrazení chybových zpráv) . . . . .	84
Programy výstupních bodů FTP . . . . .	84
Výstupní bod Ověření platnosti požadavku: klient a server . . . . .	85
Příklad: Program výstupního bodu Ověření platnosti požadavku na FTP klientu nebo FTP serveru v jazyce CL. . . . .	86
Příklad: Program výstupního bodu Ověření platnosti požadavku na FTP serveru v jazyce ILE RPG . . . . .	88
Formát výstupního bodu VLRQ0100. . . . .	93
Výstupní bod Přihlášení k serveru . . . . .	97
Příklad: Program výstupního bodu Přihlášení k FTP serveru v jazyce CL . . . . .	99
Příklad: Program výstupního bodu Přihlášení k FTP serveru v jazyce C . . . . .	100
Příklad: Program výstupního bodu Přihlášení k FTP serveru v jazyce ILE RPG . . . . .	109
Formát výstupního bodu TCPL0100 . . . . .	111
Formát výstupního bodu TCPL0200 . . . . .	115
Formát výstupního bodu TCPL0300 . . . . .	121
Odstraňování programů výstupních bodů . . . . .	124
Metody přenosu dat . . . . .	125
Přenos souborů, které obsahují pakovaná dekadická data, mezi servery iSeries <sup>(TM)</sup> . . . . .	125
Přenos souborů *SAVF . . . . .	125
Přenos dokumentů QDLS . . . . .	127
Přenos souborů typu "kořenový", QOpenSys, QLANSrv, QDLS a QOPT . . . . .	127
Přenos souborů pomocí QfileSvr.400 . . . . .	127
Přenos souborů QSYS.LIB. . . . .	128
Příjem textových souborů u QSYS.LIB . . . . .	130
Pokyny k předběžnému vytváření souborů . . . . .	131
Konverze CCSID . . . . .	131
Specifikace mapovacích tabulek. . . . .	131
Příznak kódové stránky CCSID u souborů iSeries <sup>(TM)</sup> . . . . .	132
Pokyny k FTP týkající se NLS . . . . .	133
Systémy souborů a konvence pojmenování . . . . .	134
Systémy souborů OS/400 <sup>(TM)</sup> , které jsou podporovány FTP . . . . .	135
Stavové zprávy FTP serveru . . . . .	135
Konvence syntaxe FTP serveru . . . . .	137
Konvence syntaxe FTP klienta . . . . .	138
Ohraničení parametrů podpříkazu . . . . .	138
Jména souborů v podpříkazech klienta pro přenos . . . . .	139
Pojmenování souborů pro přenos . . . . .	140
Odstraňování problémů FTP . . . . .	141
Určování problémů FTP. . . . .	142
Povinné podklady při nahlašování problémů FTP . . . . .	144

Trasování FTP serveru . . . . .	144
Trasování FTP klienta . . . . .	146
Práce s úlohami a s protokoly úloh FTP serveru . . . . .	147
<b>Dodatek. Upozornění . . . . .</b>	<b>149</b>
Ochranné známky . . . . .	150
Ustanovení a podmínky pro stahování a tisk publikací . . . . .	150
Prohlášení o vyloučení záruky na příklady programovacího kódu . . . . .	151



# FTP

Server iSeries<sup>(TM)</sup> lze nastavit tak, aby odesílal, přijímal a sdílel soubory v sítích prostřednictvím protokolu FTP (File Transfer Protocol). Tímto způsobem se soubory mohou také přejmenovávat, přidávat a mazat. Než nastavíte systém pro přenos souborů, musíte v něm nakonfigurovat a spustit protokol TCP/IP.

Více informací o používání FTP na serveru iSeries najdete v těchto tématech:

## **Tisk tohoto tématu**

Dokumentaci k FTP si můžete stáhnout nebo vytisknout ve formátu PDF.

## **Scénáře FTP**

Přečtěte si scénáře, které vám ukáží, jak FTP nakonfigurovat a jak jej používat.

## **Konfigurace FTP serveru**

Nastavte FTP server iSeries pro grafické FTP klienty, prohlížeče Web a nástroje pro Web. Nakonfigurujte anonymní FTP.

## **Zabezpečení ochrany FTP**

Chraňte svoje data zabezpečením FTP pomocí SSL, monitorováním uživatelů FTP a správou přístupu uživatelů k funkcím FTP.

## **Správa FTP serveru**

Obsahuje pokyny pro správu FTP serveru včetně zapínání a vypínání, zabezpečení ochrany dat u FTP a používání SSL.

## **Používání FTP klienta u serveru iSeries**

Dozvíte se, jak spustit a ukončit relaci klienta, jak odesílat a přijímat soubory a jak nastavit dávkové zpracování FTP.

## **Referenční informace o FTP**

Obsahuje informace o podpříkazech serveru i klienta, o programech výstupních bodů FTP, o metodách přenosu dat a další.

## **Odstraňování problémů FTP**

Obsahuje informace o odstraňování problémů s FTP serverem nebo s FTP klientem.

## **Poznámka:**

Důležité informace o souvisejících právech si můžete přečíst v Prohlášení k příkladům programového kódu.



---

## **Tisk tohoto tématu**

Chcete-li si prohlédnout nebo stáhnout PDF verzi, vyberte FTP (přibližně 934 kB).

## **Uložení souborů PDF**

Chcete-li uložit soubor PDF na pracovní stanici za účelem zobrazení nebo tisku:

1. V prohlížeči klepněte pravým tlačítkem myši na daný soubor PDF (na výše uvedený odkaz).
2. Klepněte na **Save Target As...** (Uložit cíl jako...).
3. Vyhledejte adresář, do kterého chcete soubor PDF uložit.
4. Klepněte na **Save** (Uložit).

## Stažení programu Adobe Acrobat Reader

Potřebujete-li k zobrazení nebo tisku těchto PDF program Adobe Acrobat Reader, můžete si stáhnout jeho kopii na Web stanici Adobe ([www.adobe.com/products/acrobat/readstep.html](http://www.adobe.com/products/acrobat/readstep.html))



---

## Scénáře FTP



Následující scénáře vás seznámí s tím, jak FTP funguje a jak používat prostředí FTP ve vaší síti. Tyto scénáře představují základní koncepce FTP, které mohou využít jak začátečníci, tak pokročilí uživatelé, než přejdou k úkolům plánování a konfigurace.

### Přenos souboru ze vzdáleného hostitele

Můžete přenést soubor na testovací server. K odeslání souboru na vzdáleného hostitele použijte základní FTP.

### Zabezpečení FTP prostřednictvím SSL

Můžete zabezpečit data, která odesíláte své partnerské firmě, pomocí SSL (Secure Sockets Layer).



## Scénář: Přenos souboru ze vzdáleného hostitele



### Cíle

Předpokládejme, že váš kolega pracoval na nějakém vývoji v Javě<sup>(TM)</sup> na vzdáleném serveru. Abyste mohli jeho práci otestovat, potřebujete přenést soubor `example.jar` z tohoto vzdáleného serveru na svůj lokální testovací server. K přenosu tohoto souboru (binárně) v síti TCP/IP použijte základní FTP. Klient i server jsou oba z řady iSeries<sup>(TM)</sup> a používají FTP operačního systému OS/400<sup>(R)</sup>.

### Podrobnosti

Na přenosu souboru se spolupodílejí dvě připojení: *řídící připojení* a *datové připojení*. Řídící připojení slouží k odesílání podpříkazů z klienta na server a k přijímání odpovědí na tyto podpříkazy ze serveru na klienta. Klient vydává podpříkazy FTP, které se odesílají na FTP server. Datové připojení slouží k přenosu skutečných souborů. Klient i server jsou napojeny na systém souborů OS/400. K přenosu souborů potřebujete mít obvykle ID uživatele v obou systémech. Dále potřebujete:

- Server iSeries s operačním systémem OS/400.
- TCP/IP Connectivity Utilities (5722-TC1).
- Nakonfigurovaný FTP server.
- Hostitelské jméno vzdáleného systému.
- Svoje uživatelské jméno a heslo ve vzdáleném systému.
- Jméno souboru, který chcete přenést.
- Umístění souboru, který chcete přenést.
- Formát souboru (tj. formát, ve kterém musíte soubor přenést, například binární nebo ASCII).

## Úlohy konfigurace

K jednoduchému přenosu souboru je třeba provést všechny následující kroky:

1. Spusíte relaci FTP klienta.  
V tomto scénáři: Ve znakovém rozhraní iSeries napište STRTCPFTP a stiskněte klávesu **Enter**.
2. Zadejte jméno vzdáleného systému, do kterého chcete soubor odeslat.  
V tomto scénáři: jejichspol.com
3. Sdělte vzdálenému systému svoje uživatelské jméno pro vzdálený server.  
V tomto scénáři:  
Enter login ID (yourid):  
====>vašeid
4. Sdělte vzdálenému systému svoje uživatelské heslo pro vzdálený server.  
V tomto scénáři:  
Enter password:  
====>vašeheslo
5. Vyhledejte adresář na serveru JejichSpol, ze kterého chcete soubor přenést.  
V tomto scénáři: ====>cd /qibm/userdata/os400/dirserv/usertools/windows
6. Na lokálním serveru přejděte do adresáře, do něhož chcete soubor přenést.  
V tomto scénáři: ====>lcd /qibm/userdata/os400/dirserv/usertools/windows
7. Zadejte typ souboru, ASCII nebo BINARY. Předvolený typ je ASCII.  
U souboru .jar musíte přepnout typ přenosu souboru na binární.  
V tomto scénáři: ====> binary
8. Zadejte požadavek na přenos souboru ze systému vzdáleného serveru do systému klienta.  
V tomto scénáři: ====> get example.jar
9. Po dokončení přenosu ukončete relaci FTP.  
V tomto scénáři: ====> QUIT

### Další krok

Běžte o krok dále. Soubory můžete přenášet také automatickým způsobem pomocí dávkového FTP.



## Scénář: Zabezpečení FTP prostřednictvím SSL



Předpokládejme, že pracujete ve firmě MojeSpol, která provádí průzkum začínajících firem a prodává výsledky svého průzkumu firmám působícím v oblasti plánování investic. Jedna z těchto firem, JejichSpol, má zájem o služby, které MojeSpol nabízí, a chtěla by dostávat výsledky průzkumu prostřednictvím FTP. MojeSpol vždy zaručovala důvěrnost a bezpečnost dat, která šíří mezi svými zákazníky, bez ohledu na formát. V tomto případě MojeSpol potřebuje mít relace FTP s firmou JejichSpol zabezpečeny prostřednictvím SSL.

### Cíle

Vaše cíle v tomto scénáři jsou následující:

- Vytvořit a zprovoznit lokálního vydavatele certifikátů (CA) na serveru iSeries<sup>(TM)</sup> firmy MojeSpol.
- Povolit SSL na FTP serveru firmy MojeSpol.
- Exportovat kopii lokálního certifikátu CA do souboru.
- Vytvořit paměť certifikátů \*SYSTEM na serveru JejichSpol.
- Importovat lokální certifikát CA firmy MojeSpol do paměti certifikátů \*SYSTEM firmy JejichSpol.

- Nastavit lokálního CA firmy MojeSpol jako důvěryhodného CA pro FTP klienta JejichSpol.

### Nezbytné předpoklady

#### MojeSpol

- Má server iSeries s operačním systémem OS/400<sup>(R)</sup> verze V5R1 nebo vyšší.
- Má na serveru iSeries instalován produkt TCP/IP Connectivity Utilities (5722-TC1) verze V5R1.
- Má na serveru iSeries instalován produkt Cryptographic Access Provider 128-bit for iSeries server (5722-AC3).
- Má na serveru iSeries instalován produkt IBM<sup>(R)</sup> Digital Certificate Manager (DCM) (5722-SS1 volba 34).
- Má na serveru iSeries instalován IBM HTTP Server (5722-DG1).
- K ochraně přístupu k veřejným aplikacím a zdrojům používá certifikáty (podrobné pokyny viz Scenario: Use certificates to protect access to public applications and resources).

#### JejichSpol

- Má server iSeries s operačním systémem OS/400 verze V5R2 nebo vyšší.
- Má na serveru iSeries instalován produkt TCP/IP Connectivity Utilities (5722-TC1) verze V5R2.
- Má na serveru iSeries instalován produkt Cryptographic Access Provider 128-bit for iSeries server (5722-AC3).
- Má na serveru iSeries instalován produkt IBM Digital Certificate Manager (5722-SS1 volba 34).
- Má na serveru iSeries instalován IBM HTTP Server (5722-DG1).
- Pro relace FTP používá FTP klienta TCP/IP OS/400.

#### Podrobnosti

JejichSpol pomocí FTP klienta OS/400 zadá k požadavek na zabezpečený přenos souboru z FTP serveru firmy MojeSpol prostřednictvím FTP. Podívejte se na téma Zabezpečení FTP klienta pomocí TLS/SSL. Provede se autentizace na serveru.

JejichSpol obdrží finanční hlášení od firmy MojeSpol v relaci FTP zabezpečené prostřednictvím SSL.

#### Úlohy konfigurace

K zabezpečení svých relací FTP prostřednictvím SSL musí firmy MojeSpol a JejichSpol splnit tyto úkoly:

##### Úkoly firmy MojeSpol:

1. Vytvoření a zprovoznění lokálního vydavatele certifikátů (CA) na serveru iSeries firmy MojeSpol (See 5).
2. Povolení SSL na FTP serveru MojeSpol (See 5).
3. Export kopie lokálního certifikátu CA do souboru (See 6).

##### Úkoly firmy JejichSpol:

1. Vytvoření paměti certifikátů \*SYSTEM na serveru JejichSpol (See 6).
2. Import lokálního certifikátu CA do paměti certifikátů \*SYSTEM na serveru JejichSpol (See 7).
3. Označení lokálního CA firmy MojeSpol jako důvěryhodného CA pro FTP klienta JejichSpol (See 7).



#### Podrobnosti ke konfiguraci

Chcete-li zabezpečit FTP prostřednictvím SSL, proveďte následující úkoly.

## Krok 1: Vytvoření a zprovoznění lokálního vydavatele certifikátů (CA) na serveru iSeries<sup>™</sup> firmy MojeSpol.

Tento scénář předpokládá, že MojeSpol dosud nevytvářela certifikáty pro server iSeries pomocí produktu DCM (Digital Certificate Manager). Na základě cílů tohoto scénáře se MojeSpol rozhodla vytvořit a zprovoznit lokálního vydavatele certifikátů (CA), aby mohla vydat certifikát pro FTP server. Namísto toho by MojeSpol také mohla pomocí DCM nakonfigurovat FTP server na používání veřejného certifikátu pro SSL.

Když k vytvoření lokálního CA použijete DCM (Digital Certificate Manager), provede vás tento program celým postupem, takže bude zajištěno, že nakonfigurujete vše, co je k povolení SSL potřeba.

K vytvoření a zprovoznění lokálního CA na svém serveru pomocí DCM použije MojeSpol tento postup:

1. Spusťte DCM.
2. V navigačním rámci DCM vyberte **Vytvoření vydavatele certifikátů (CA)**. Zobrazí se řada formulářů. Tyto formuláře vám pomohou vytvořit lokálního CA a provést i další kroky potřebné k tomu, abyste mohli začít používat digitální certifikáty pro SSL, pro podepisování objektů a pro ověřování podpisů.
3. Vyplňte všechny formuláře, které se zobrazí. Pro každý z úkolů potřebných k vytvoření a zprovoznění lokálního CA na serveru iSeries slouží jeden formulář. Úkoly jsou následující:
  - a. Zvolte způsob uložení soukromého klíče k lokálnímu certifikátu CA. Tento krok je potřebný pouze tehdy, máte-li na serveru iSeries nainstalován šifrovací koprocesor (IBM<sup>®</sup> 4758-023 PCI Cryptographic Coprocessor). Nemáte-li šifrovací koprocesor, DCM automaticky uloží certifikát i s jeho soukromým klíčem do paměti lokálních certifikátů CA.
  - b. Zadejte identifikační údaje lokálního CA.
  - c. Nainstalujte lokální certifikát CA na PC nebo do prohlížeče. To umožní, aby software rozpoznal lokálního CA a mohl ověřovat certifikáty, které tento CA vydává.
  - d. Zvolte strategická data pro lokálního CA.
  - e. Pomocí nového lokálního CA vydejte certifikát pro server nebo klienta, který budou aplikace používat k připojování prostřednictvím SSL. Máte-li na serveru iSeries nainstalován šifrovací koprocesor (IBM 4758-023 PCI Cryptographic Coprocessor), tento krok vám umožní vybrat i způsob uložení soukromého klíče k certifikátu serveru nebo klienta. Nemáte-li šifrovací koprocesor, DCM automaticky uloží certifikát i s jeho soukromým klíčem do paměti certifikátů \*SYSTEM. DCM vytvoří paměť certifikátů \*SYSTEM automaticky v rámci tohoto úkolu.
  - f. Vyberte aplikace, které budou moci používat tento certifikát serveru nebo klienta k připojování prostřednictvím SSL. Poznámka: Dbejte na to, abyste vybrali ID aplikace pro FTP server TCP/IP OS/400<sup>®</sup> (QIBM\_QTMF\_FTP\_SERVER).
  - g. Pomocí nového lokálního CA vydejte certifikát pro podepisování objektů, který budou aplikace moci používat k digitálnímu podepisování objektů. Přitom se vytvoří paměť certifikátů \*OBJECTSIGNING, která slouží ke správě certifikátů pro podepisování objektů. Poznámka: I když tento scénář nepočítá s používáním certifikátů pro podepisování objektů, přesto tento krok proveďte. Kdybyste nyní tento krok zrušili, celý úkol by se ukončil a další kroky k dokončení konfigurace certifikátů pro SSL byste museli provádět jednotlivě.
  - h. Vyberte aplikace, které budou důvěřovat vašemu lokálnímu CA. Poznámka: Dbejte na to, abyste vybrali ID aplikace pro FTP server TCP/IP OS/400 (QIBM\_QTMF\_FTP\_SERVER).

Když jsou všechny formuláře tohoto úkolu vyplněny, můžete přikročit ke konfiguraci FTP serveru na používání SSL.

## Krok 2: Nastavení SSL na FTP serveru firmy MojeSpol

Když má FTP server přidělen certifikát, může MojeSpol nakonfigurovat FTP server na používání SSL. K tomu slouží tento postup:

1. V produktu iSeries Navigator rozbalte **iSeries server** → **Síť** → **Servery** → **TCP/IP**.
2. Klepněte pravým tlačítkem myši na **FTP**.
3. Vyberte **Vlastnosti**.
4. Vyberte ouško **Obecné**.
5. Pro podporu SSL vyberte volbu **Pouze zabezpečené**.  
Když vyberete tuto volbu, budou na FTP serveru povoleny pouze relace SSL. Připojení se mohou uskutečnit i přes nezabezpečený FTP port, ale FTP klient musí nejprve vyjednat relaci SSL, a teprve potom se uživatel může připojit.

Když je tento úkol dokončen, může FTP server firmy MojeSpol pomocí SSL šifrovat komunikační relace a chránit důvěrnost dat přenášených při těchto relacích. Aby však bylo možné nakonfigurovat FTP klienta tak, že se bude moci účastnit relací SSL na FTP serveru, musí MojeSpol poskytnout jejich klientu (JejichSpol) kopii lokálního certifikátu CA. MojeSpol proto musí exportovat kopii lokálního certifikátu CA do souboru a zpřístupnit tento soubor firmě JejichSpol. Když JejichSpol získá tento soubor, může pomocí DCM importovat lokální certifikát CA do paměti certifikátů \*SYSTEM a nakonfigurovat FTP klienta OS/400 na používání SSL.

### Krok 3: Export kopie lokálního certifikátu CA do souboru

MojeSpol musí poskytnout JejichSpol kopii lokálního certifikátu CA. Aby se aplikace typu klient JejichSpol mohla účastnit relace SSL, musí být nakonfigurována tak, aby považovala tento certifikát CA za důvěryhodný.

K exportu kopie lokálního certifikátu CA do souboru použije MojeSpol tento postup:

1. Spusťte DCM.
2. Klepněte na **Výběr paměti certifikátů**.
3. Jako paměť certifikátů, která se má otevřít, vyberte **\*SYSTEM** a klepněte na **Pokračovat**.
4. Když se objeví obrazovka Paměť certifikátů a heslo, uveďte heslo, které bylo nastaveno pro paměť certifikátů při jejím vytvoření, a klepněte na **Pokračovat**.
5. Po obnovení navigačního rámce vyberte **Správa certifikátů** a potom **Export certifikátu**.
6. Vyberte **Vydavatel certifikátů (CA)** a klepněte na **Pokračovat**. Zobrazí se seznam certifikátů CA.
7. Ze seznamu vyberte lokální certifikát CA a klepněte na **Export**.
8. Jako cíl exportu uveďte **Soubor** a klepněte na **Pokračovat**.
9. Zadejte úplnou cestu a jméno souboru pro exportovaný lokální certifikát CA a klepněte na **Pokračovat**. Certifikát se vyexportuje.
10. Klepnutím na **OK** opusťte obrazovku Potvrzení exportu.

Nyní můžete tyto soubory přenést do koncových systémů, v nichž se budou ověřovat podpisy vytvořené tímto certifikátem. K přenesení souborů můžete použít elektronickou poštu nebo FTP, protože nemusí být posílány zabezpečené.

JejichSpol potom pomocí DCM naimportuje lokální certifikát CA do paměti certifikátů \*SYSTEM a uvede tento lokální certifikát CA firmy MojeSpol (a certifikáty, které vydá) jako důvěryhodný.

### Krok 4: Vytvoření paměti certifikátů \*SYSTEM na serveru JejichSpol

Aby se FTP klient OS/400 firmy JejichSpol mohl účastnit relace SSL, musí umět rozpoznat a potvrdit certifikát, který předloží FTP server firmy MojeSpol při zahájení relace SSL. Aby mohl FTP klient JejichSpol autentizovat tento certifikát CA, musí mít jeho kopii uloženou v paměti certifikátů \*SYSTEM. Paměť

certifikátů \*SYSTEM obsahuje kopie většiny známých veřejných certifikátů CA. Pokud však FTP server firmy MojeSpol používá lokální certifikát CA, musí FTP klient JejichSpol získat kopii tohoto lokálního certifikátu CA a naimportovat ji do paměti certifikátů \*SYSTEM.

Tento scénář předpokládá, že žádná z firem dosud nepoužívala DCM (Digital Certificate Manager) k vytváření a správě certifikátů. Proto nyní musí firma JejichSpol nejprve vytvořit paměť certifikátů \*SYSTEM. K tomu slouží tento postup:

1. Spusťte DCM.
2. V navigačním rámci DCM vyberte **Vytvoření nové paměti certifikátů** a jako paměť certifikátů, která se má vytvořit, vyberte **\*SYSTEM**. Klepněte na **Pokračovat**.
3. Vyberte **Ne** pro vytvoření certifikátu jako součást vytváření paměti certifikátů \*SYSTEM a klepněte na **Pokračovat**.
4. Zadejte heslo pro novou paměť certifikátů a klepněte na **Pokračovat**. Zobrazí se obrazovka pro potvrzení.
5. Klepněte na **OK**.

Nyní může JejichSpol naimportovat lokální certifikát CA do paměti certifikátů a označit ji za důvěryhodný zdroj certifikátů.

#### **Krok 5: Import lokálního certifikátu CA do paměti certifikátů \*SYSTEM na serveru JejichSpol**

K importu lokálního certifikátu CA do paměti certifikátů \*SYSTEM a označení této paměti za důvěryhodnou použije JejichSpol tento postup:

1. V navigačním rámci DCM klepněte na **Výběr paměti certifikátů** a jako paměť certifikátů, která se má otevřít, vyberte **\*SYSTEM**.
2. Když se objeví obrazovka Paměť certifikátů a heslo, uveďte heslo, které bylo nastaveno pro paměť certifikátů při jejím vytvoření, a klepněte na **Pokračovat**.
3. Po obnovení navigačního rámce vyberte **Správa certifikátů**. Zobrazí se seznam úloh.
4. Ze seznamu úloh vyberte **Import certifikátu**.
5. Jako typ certifikátu vyberte **Vydavatel certifikátu (CA)** a klepněte na **Pokračovat**.
6. Zadejte úplnou cestu a jméno souboru pro soubor s certifikátem CA a klepněte na **Pokračovat**. Objeví se zpráva, že import byl úspěšně dokončen, nebo v případě selhání se objeví chybová zpráva.

Nyní může JejichSpol zadat, že jejich FTP klient považuje lokální certifikát CA firmy MojeSpol za důvěryhodný, takže FTP klient firmy JejichSpol se bude moci účastnit relací SSL s aplikacemi typu server, které používají certifikát vydaný lokálním CA firmy MojeSpol.

#### **Krok 6: Označení lokálního CA firmy MojeSpol jako důvěryhodného CA pro FTP klienta JejichSpol**

Aby mohla JejichSpol používat svého FTP klienta k zabezpečenému připojování k FTP serveru firmy MojeSpol, musí pomocí DCM určit, kterému CA by měl jejich klient důvěřovat. To znamená, že JejichSpol musí zadat, že lokální certifikát CA naimportovaný v předchozím kroku má být považován za důvěryhodný.

K nastavení FTP klienta tak, aby považoval lokální certifikát CA firmy MojeSpol za důvěryhodný, použije JejichSpol tento postup:

1. Spusťte DCM.
2. Klepněte na **Výběr paměti certifikátů** a jako paměť certifikátů, která se má otevřít, vyberte \*SYSTEM.
3. Když se objeví obrazovka Paměť certifikátů a heslo, uveďte heslo, které bylo nastaveno pro paměť certifikátů při jejím vytvoření, a klepněte na **Pokračovat**.
4. V navigačním rámci vyberte **Správa aplikací**. Zobrazí se seznam úloh.
5. Ze seznamu úloh vyberte **Definování seznamu důvěryhodných CA**.



6. Jako typ aplikace, pro kterou chcete seznam definovat, vyberte **Klient** a klepněte na **Pokračovat**.
7. Ze seznamu vyberte aplikaci QIBM\_QTMF\_FTP\_CLIENT (FTP klient TCP/IP OS/400) a klepněte na **Pokračovat**. Zobrazí se seznam certifikátů CA.
8. Vyberte lokální certifikát CA firmy MojeSpol, který byl naimportován v předchozím kroku, a klepněte na **OK**. DCM zobrazí zprávu potvrzující tento výběr do důvěryhodného seznamu.

Když jsou všechny kroky dokončeny, FTP server firmy MojeSpol může zahájit relaci SSL s FTP klientem a serverem firmy JejichSpol. Viz ?Zabezpečení FTP klienta pomocí TLS/SSL.

---

## Konfigurace FTP serveru



Licencovaný program TCP/IP Connectivity Utilities přichází s nakonfigurovanými FTP servery pro TCP/IP. Když spustíte TCP/IP, spustí se současně i FTP server.

Než přikročíte ke konfigurování FTP serveru pro Internet, měli byste si v zájmu ochrany svých dat přečíst následující bezpečnostní pokyny:

- Použijte ochrannou bariéru (firewall) mezi svým serverem iSeries<sup>™</sup> a Internetem.
- Jako FTP server používejte jiný než provozní server iSeries.
- Nepřipojujte FTP server ke zbývajícím sítím LAN nebo WAN vaší firmy.
- K zabezpečení přístupu k FTP serveru používejte programy výstupních bodů FTP.
- Jednou měsíčně provádějte testování programů výstupních bodů FTP, abyste zjistili, zda neobsahují nějaké nedostatky v zabezpečení.
- Nepovolte anonymním uživatelům FTP přístup pro čtení i zápis současně ke stejnému adresáři. Tím by se anonymní uživatel stal v síti Internet nevysledovatelným.
- Zaznamenávejte veškerý přístup ke svému serveru iSeries a tyto protokoly prohlížejte jednou denně nebo jednou týdně, abyste zjistili možná napadení.
- Jednou měsíčně si ověřte, zda jsou pro FTP server registrovány správné programy výstupních bodů.
- Přečtěte si informace o zabezpečení FTP serveru iSeries v tématu Zabezpečení ochrany FTP.

Následující témata nabízejí způsoby, jak prohlížet a přizpůsobit FTP servery:

### FTP server v produktu iSeries Navigator

Popisuje, jak používat produkt iSeries Navigator ke konfiguraci a správě FTP serveru iSeries.

### Konfigurace FTP serverů pro grafické FTP klienty a nástroje pro Web

Popisuje, jak nakonfigurovat FTP server iSeries pro podporu grafických FTP klientů, prohlížečů Web a dalších nástrojů pro Web.

### Konfigurace anonymního FTP

Anonymní FTP umožňuje vzdáleným uživatelům používat váš FTP server, aniž by měli své vlastní uživatelské ID a heslo.



## FTP server v produktu iSeries<sup>™</sup> Navigator

K práci s konfigurací FTP serveru můžete použít iSeries Navigator. Chcete-li v produktu iSeries Navigator přejít na grafické uživatelské rozhraní pro FTP, použijte tento postup:

1. V produktu iSeries Navigator rozbalte **váš server iSeries** → **Síť** → **Servery** → **TCP/IP**.
2. V pravém okně klepněte pravým tlačítkem myši na **FTP** a vyberte **Vlastnosti**.



3. Pomocí této obrazovky můžete změnit vlastnosti FTP serveru. Klepnutím na tlačítka nápovědy si můžete zobrazit online nápovědu. Chcete-li zobrazit nápovědu ke konkrétnímu poli, klepněte na tlačítko s otazníkem a potom na požadované pole.

## Konfigurace FTP serverů pro grafické FTP klienty a nástroje pro Web



FTP server iSeries<sup>(TM)</sup> podporuje grafické FTP klienty, webové prohlížeče a nástroje pro vývoj webových aplikací.

Většina grafických FTP klientů používá jako formát seznamu UNIX<sup>(R)</sup> a jako formát jména souboru cestu. Podle následujících pokynů nastavte vlastnosti FTP serveru tak, aby používal podporované formáty:

1. V produktu iSeries Navigator rozbalte **váš server iSeries** → **Síť** → **Servery** → **TCP/IP**.
2. V pravém okně klepněte pravým tlačítkem myši na **FTP** a vyberte **Vlastnosti**.
3. Na straně **Vlastnosti** klepněte na ouško **Výchozí formáty**.
  - Jako formát pojmenování souborů aktivujte položku **Cesta**.
  - Jako formát seznamu souborů aktivujte položku **Formát výpisu UNIX**.

### Poznámka:

Nastavení parametrů LISTFMT a NAMEFMT pro konkrétní relace FTP můžete řídit pomocí programu výstupního bodu Přihlášení k FTP serveru ve formátu TCPL0200 nebo formátu TCPL0300.

Rovněž můžete změnit formát seznamu *poté*, co je relace FTP spuštěna, pomocí voleb podpříkazu FTP serveru SITE (Odeslat informace používané systémem serveru). Podle těchto nastavení se řídí výsledky vrácené podpříkazu FTP serveru LIST (Seznam souborů) a NLST (Seznam jmen).

Můžete se také podívat na:

- Informace podpříkazu LIST ve formátu iSeries.
- Informace podpříkazu LIST ve formátu UNIX.



## Konfigurace anonymního FTP

Anonymní FTP umožňuje nechráněný přístup (bez hesla) k vybraným informacím ze vzdáleného systému. Vzdálený systém určuje, kterou informaci dá k dispozici pro všeobecný přístup. Taková informace se považuje za veřejně přístupnou a může ji kdokoli číst. Je na zodpovědnosti osoby, která tyto informace a systém vlastní, aby zajistila, že budou zpřístupněny pouze patřičné informace.

Aby uživatel získal k těmto informacím přístup, přihlásí se k hostitelskému systému pod uživatelským ID ANONYMOUS. Uživatel ANONYMOUS má omezená přístupová práva k souborům na serveru a má omezeny některé činnosti. K činnostem, které má tento uživatel povoleny, patří:

- Přihlášení se pomocí FTP.
- Výpis obsahu omezené sady adresářů.
- Načítání souborů z těchto adresářů.

Anonymní uživatelé obvykle nemají povoleno přenášet soubory na FTP server. Některé systémy poskytují anonymním uživatelům adresář pro příchozí data, kam mohou svá data posílat. Speciální účet anonymního uživatele tradičně akceptuje jako heslo libovolný řetězec, obvykle se však používá buď heslo 'guest' nebo adresa elektronické pošty. Některé archivní uzly vyžadují přímo adresu elektronické pošty daného uživatele a nedovolují přihlásit se pomocí hesla guest. Poskytnutí adresy elektronické pošty je projevem laskavosti, který umožní operátorům archivního uzlu získat přehled o tom, kdo využívá jejich služeb.

## Anonymní FTP na serveru iSeries<sup>(TM)</sup>

Základní FTP (File Transfer Protocol) server nepodporuje anonymní FTP. Abyste mohli na serveru iSeries nastavit anonymní FTP, je třeba vytvořit programy pro dva výstupní body FTP serveru (výstupní bod Přihlášení k serveru a výstupní bod Ověření platnosti požadavku).

Možná máte v úmyslu poskytovat anonymní FTP, protože jde o praktickou a často také potřebnou službu. Používání anonymního FTP však klade vyšší nároky na zabezpečení serveru iSeries. Další informace o tom, jak zabezpečit systém, najdete v tématu Zabezpečení FTP pomocí programů výstupních bodů.

### Ke konfiguraci anonymního FTP je třeba splnit tyto úkoly:

- 1. Příprava pro anonymní FTP**  
Seznamte se s technickými požadavky a definujte strategii zabezpečení.
- 2. Vytvoření programů výstupních bodů pro anonymní FTP**  
Napište dva programy výstupních bodů pro podporu anonymního FTP. Toto téma popisuje mnoho příkladů programů výstupních bodů a užitečné rady a techniky.
- 3. Vytvoření uživatelského profilu OS/400<sup>(R)</sup> ANONYMOUS**  
Zabraňte tomu, aby se někdo přihlašoval na váš server iSeries *přímo* pod uživatelským ID ANONYMOUS.
- 4. Vytvoření veřejné knihovny nebo adresáře**  
Vytvořte, zaveďte a nastavte svoje veřejné knihovny nebo adresáře.
- 5. Instalace a registrace programů výstupních bodů**  
Vytvořte knihovnu, která bude obsahovat vaše programy výstupních bodů a jejich soubory protokolů, tyto programy zkompilujte a zaregistrujte je pro používání FTP serverem.

### Příprava pro anonymní FTP

Při přípravě konfigurace anonymního FTP zvažte následující informace.

#### Požadované znalosti

Abyste mohli nastavit anonymní FTP, měli byste mít tyto dispozice:

- Znalost znakového rozhraní serveru iSeries<sup>(TM)</sup> a příkazů s více parametry a klíčovými slovy.
- Možnost vytvářet na serveru iSeries knihovny, členy a zdrojové fyzické soubory (měli byste mít minimálně oprávnění \*SECOFR).
- Možnost přiřazovat oprávnění ke knihovnám, souborům, členům a programům.
- Možnost zapisovat, měnit, kompilovat a testovat programy na serveru iSeries.

#### Pokyny k zabezpečení ochrany dat

Prvním krokem při implementaci anonymního FTP je definovat strategii pro anonymní FTP na serveru. Tento plán definuje zabezpečení vašeho FTP serveru a určuje způsob kódování vašich programů výstupních bodů. Protože váš FTP server bude povolovat komukoli na světě přístup k vašim datům, musíte pečlivě zvážit, jak by se měl používat a která data je třeba chránit.

Přečtěte si několik doporučení týkajících se plánování strategie pro FTP:

- Používejte ochrannou bariéru (firewall) mezi svým serverem iSeries a Internetem.
- Jako FTP server používejte jiný než provozní server iSeries.
- Nepřipojujte FTP server ke zbývajícím sítím LAN nebo WAN vaší firmy.
- K zabezpečení přístupu k FTP serveru používejte programy výstupních bodů FTP.
- Provádějte testování programů výstupních bodů FTP, abyste zjistili, zda neobsahují nějaké nedostatky v zabezpečení.
- Nepovolte anonymním uživatelům FTP přístup pro čtení i zápis současně ke stejnému adresáři. Tím by se anonymní uživatel stal v síti Internet nevysledovatelným.
- Povolte přístup pouze uživateli ANONYMOUS. Nepovolte žádná jiná uživatelská ID a neautentizujte jejich hesla.

- Omezte přístup uživatele ANONYMOUS pouze na jednu veřejnou knihovnu nebo adresář. (Kde bude? Jak se bude jmenovat?)
- Do veřejné knihovny nebo adresáře umístěte pouze veřejně přístupné soubory.
- Omezte přístup uživatele ANONYMOUS pouze na podpříkazy sloužící k prohlížení a načítání (get, mget). **Za žádných okolností nepovolte uživateli ANONYMOUS používat CL příkazy.**
- Zaznamenávejte veškerý přístup ke svému serveru iSeries.
- Tyto protokoly FTP serveru denně nebo týdně prohlízejte, abyste byli informováni o možných napadeních.
- Jednou měsíčně si ověřte, že FTP server má zaregistrovány správné programy výstupních bodů.
- Jednou měsíčně otestujte FTP server na nedostatky v zabezpečení.

### Co dělat dále:

Vytvoření programů výstupních bodů pro anonymní FTP.

## Vytvoření programů výstupních bodů pro anonymní FTP



Pro podporu anonymního FTP je třeba napsat dva programy výstupních bodů: program Přihlášení k FTP serveru (FTP Server Logon) a program Ověření platnosti požadavku na FTP serveru (FTP Server Request Validation). Program výstupního bodu Přihlášení k FTP serveru povolí ID uživatele ANONYMOUS a donutí uživatele ANONYMOUS používat veřejnou knihovnu nebo adresář. Program výstupního bodu Ověření platnosti požadavku na FTP serveru omezuje příkazy, soubory a adresáře nebo knihovny, které může uživatel ANONYMOUS používat.

### Výstupní body a formáty výstupních bodů

FTP server komunikuje s každým programem výstupního bodu prostřednictvím určitého výstupního bodu. Mezi serverem a programem výstupního bodu se předávají parametry. Formát předávané informace je dán formátem výstupního bodu. Další informace o programech výstupních bodů pro FTP najdete v tématu Programy výstupních bodů FTP. U anonymního FTP se používají tyto výstupní body:

Program	Výstupní bod	Formát
Přihlášení k serveru	QIBM_QTMF_SVR_LOGON	TCPL0100, TCPL0200 nebo TCPL0300. <sup>1</sup>
Ověření platnosti požadavku	QIBM_QTMF_SERVER_REQVLRQ0100	

<sup>1</sup> -Výstupní bod může mít více než jeden formát, ale program může být registrován pouze pro jeden formát výstupního bodu. Vyzkoušejte si všechny tyto formáty a potom si vyberte ten, který nejvíce odpovídá vašemu systému.

### Příklady programů

Příklady programů, které jsou k dispozici, vám pomohou nastavit na serveru anonymní FTP. Můžete z nich vycházet při vytváření vlastních programů. Okopírované části kódu ze vzorových programů můžete přidat do programů, které jste sami napsali. Doporučujeme, abyste vzorové programy spouštěli na jiném než provozním systému.

### Poznámka:

Tyto příklady slouží pouze pro ilustraci. Neobsahují dostatek funkcí na to, aby mohly být používány v provozním systému v tom stavu, v jakém jsou. Můžete je použít jako předlohu nebo si můžete zkopírovat jednotlivé části kódu a použít je ve svých vlastních programech.

Uvedené vzorové programy najdete v části, která se týká programů na serveru, v tématech Výstupní bod Přihlášení k serveru a Výstupní bod Ověření požadavku.

#### Co dělat dále:

Vytvoření uživatelského profilu OS/400<sup>(R)</sup>: ANONYMOUS



### Vytvoření uživatelského profilu OS/400<sup>(R)</sup>: ANONYMOUS

Jestliže nechcete, aby se kdokoli mohl přihlásit k vašemu serveru iSeries<sup>(TM)</sup> pod uživatelským profilem ANONYMOUS, důrazně se doporučuje, abyste vytvořili uživatelský profil ANONYMOUS a přiřadili k němu heslo \*NONE. Tento profil můžete vytvořit pomocí produktu iSeries Navigator.

1. V produktu **iSeries Navigator** rozbalte **Uživatelé a skupiny**.
2. Klepněte pravým tlačítkem myši na **Všichni uživatelé** a vyberte **Noví uživatelé**.
3. V dialogovém okně Noví uživatelé zadejte následující údaje:  
**Jméno uživatele** = ANONYMOUS a  
**Heslo** = Bez hesla.
4. Klepněte na tlačítko **Úlohy** a vyberte ouško **Obecné**.
5. Pod ouškem **Obecné** nastavte **Aktuální knihovnu** a **Domovský adresář**, jež by měl anonymní uživatel používat.
6. Klepněte na **OK** a dokončete ostatní nastavení.
7. Klepnutím na **Přidat** vytvořte tento profil.

#### Co dělat dále:

Vytvoření veřejné knihovny nebo adresáře.

### Vytvoření veřejné knihovny nebo adresáře

Po vytvoření anonymních uživatelů můžete vytvořit veřejnou knihovnu nebo adresář, které budou moci používat. Anonymní uživatelé by obvykle měli mít povolen přístup pouze k veřejným souborům. Doporučujeme, abyste omezili přístup anonymních uživatelů pouze na jedinou knihovnu nebo na jediný adresářový strom, které obsahují pouze "veřejné" soubory.

1. Vytvořte veřejné knihovny nebo adresáře, které budou obsahovat soubory přístupné prostřednictvím anonymního FTP.
2. Zaveďte vaše veřejné knihovny nebo adresáře s veřejně přístupnými soubory do paměti.
3. Nastavte oprávnění k veřejným knihovnám nebo adresářům a k souborům na PUBLIC \*USE.

#### Co dělat dále:

Instalace a registrace programů výstupních bodů.

### Instalace a registrace programů výstupních bodů

#### Instalace programu výstupního bodu

1. Vytvořte knihovnu, která bude obsahovat vaše programy výstupních bodů a jejich soubory protokolu.
2. Zkompilujte svoje programy v této knihovně.
3. Udělte oprávnění PUBLIC \*EXCLUDE k objektům knihovny, k programu i k souboru.  
Applikace FTP server převezme v případě potřeby oprávnění, aby rozhodla a zavolala program výstupního bodu.

#### Registrace programu výstupního bodu

1. Ve znakovém rozhraní serveru iSeries<sup>(TM)</sup> zadejte **WRKREGINF**.

2. Tiskněte opakovaně klávesu Page Down, dokud se nezobrazí výstupní bod Přihlášení k FTP serveru:

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

3. Zadejte **8** do pole Vol vlevo od položky výstupního bodu a stiskněte klávesu **Enter**.
4. Na obrazovce Práce s ukončovacími programy zadejte **1** (Přidat).
5. Do pole Ukončovací program zadejte jméno programu výstupního bodu.
6. Do pole Knihovna zadejte jméno knihovny, která obsahuje tento program výstupního bodu.
7. Stiskněte klávesu **Enter**.
8. Ukončete a znovu spusťte FTP server, aby bylo jisté, že všechny instance FTP serveru budou program výstupního bodu používat.
9. Svůj program výstupního bodu důkladně otestujte.

**Poznámka:**

Program výstupního bodu se projevívá okamžikem, kdy FTP server požádá o novou relaci FTP. Relace, které jsou již aktivní, nejsou ovlivněny.

**Související téma:** Odstraňování programů výstupních bodů.

---

## Zabezpečení FTP

Když používáte server iSeries<sup>(TM)</sup> jako FTP server v síti Internet, je přístupný celému světu. Proto je nutné věnovat náležitou pozornost zabezpečení ochrany FTP, aby nedošlo k ohrožení vašich firemních dat, která máte uložená na serveru iSeries. Existuje i postup, kterým můžete chránit FTP klienta.

Informace o způsobech, jak zabezpečit FTP server, najdete v těchto tématech:

### Zabránění přístupu k FTP serveru

Nepoužíváte-li FTP, měli byste zabránit spuštění FTP, aby se nikdo nemohl dostat na váš server iSeries přes FTP port. Toto téma popisuje, jak FTP port zablokovat.

### Řízení přístupu prostřednictvím FTP

Používáte-li FTP, měli byste mít kontrolu nad uživateli, abyste mohli chránit svá data i síť. Toto téma nabízí rady a pokyny pro zabezpečení ochrany dat.

### Zabezpečení FTP serveru pomocí SSL

Podpora SSL umožňuje uživateli snížit riziko odesílání hesel a dat v "čitelné formě" po síti, když používáte FTP server OS/400<sup>(R)</sup> u klienta, který rovněž podporuje SSL.

### Správa přístupu pomocí programů výstupních bodů FTP

Toto téma popisuje, jak používat výstupní body FTP k ochraně serveru iSeries.

### Správa přístupu pomocí produktu iSeries Navigator

K ochraně FTP klienta nebo serveru iSeries můžete použít funkci omezení přístupu Administrativy aplikací produktu iSeries Navigator.

### Monitorování příchozích uživatelů FTP

Monitorujte, kdo se přihlašuje na váš FTP server.

## Zabránění přístupu k FTP serveru



Jestliže nechcete, aby se kdokoli mohl připojit k vašemu serveru iSeries<sup>™</sup> prostřednictvím FTP, měli byste zabránit spuštění FTP serveru. K zabránění přístupu k serveru iSeries prostřednictvím FTP použijte tento postup:

### Zabraňte automatickému spuštění FTP serveru

Chcete-li zabránit automatickému spuštění úloh FTP serveru při spuštění TCP/IP, postupujte takto:

1. V produktu iSeries Navigator rozbalte **váš server iSeries** → **Síť** → **Servery** → **TCP/IP**.
2. Klepněte pravým tlačítkem myši na **FTP** a vyberte **Vlastnosti**.
3. Zrušte výběr u položky **Spustit při spuštění TCP/IP**.

### Zabraňte přístupu na FTP porty

Chcete-li zabránit spuštění FTP a dále vyloučit možnost přiřazení uživatelské aplikace (například aplikace typu soket) k portu, který server iSeries obvykle používá pro FTP, postupujte takto:

1. V produktu iSeries Navigator rozbalte **váš server iSeries** → **Síť** → **Servery** → **TCP/IP**.
2. Klepněte pravým tlačítkem myši na **Konfigurace TCP/IP** a vyberte **Vlastnosti**.
3. V okně **Vlastnosti konfigurace TCP/IP** klepněte na ouško **Omezení portu**.
4. Na straně **Omezení portu** klepněte na **Přidat**.
5. Na straně **Přidání omezení portu** zadejte:
  - **Jméno uživatele:** Uveďte jméno chráněného uživatelského profilu na vašem serveru iSeries. (Chráněný uživatelský profil je profil, jenž nevlastní žádné programy, které by přebíraly oprávnění, a nemá heslo, které by znali ostatní uživatelé.) Omezením portu pouze na určitého uživatele se automaticky vyloučí všichni ostatní uživatelé.
  - **Počáteční port:** 20
  - **Koncový port:** 21
  - **Protokol:** TCP
6. Klepnutím na **OK** přidáte toto omezení.
7. Na straně **Omezení portu** klepněte na **Přidat** a opakujte postup pro protokol UDP.
8. Klepnutím na **OK** uložte vámi zadaná omezení portu a zavřete okno **Vlastnosti konfigurace TCP/IP**.
9. Omezení portu se projeví při příštím spuštění TCP/IP. Je-li v době, kdy nastavujete omezení portu, TCP/IP aktivní, měli byste jej ukončit a znovu spustit.

### Poznámky:

- Omezení portu se projeví při příštím spuštění TCP/IP. Je-li v době, kdy nastavujete omezení portu, TCP/IP aktivní, měli byste jej ukončit a znovu spustit.
- Na Web stanici IANA (Internet Assigned Numbers Authority) najdete informace o přidělených číslech portů na adrese <http://www.iana.org>



- Jsou-li porty 20 nebo 21 omezeny na jiný uživatelský profil než QTCP, pak každý pokus o spuštění FTP serveru způsobí jeho okamžité ukončení s chybami.
- Tuto metodu lze použít pouze k úplnému omezení určité aplikace, jakou je například FTP server. Nelze ji použít k omezení určitých uživatelů. Když se některý uživatel připojí k FTP serveru, je pro tento požadavek nejprve použit profil QTCP. Změna na konkrétního uživatele se provede teprve po úspěšném navázání spojení. Každý uživatel FTP serveru používá oprávnění uživatele QTCP k tomuto portu.





## Řízení přístupu prostřednictvím FTP



Chcete-li povolit FTP klientům přístup ke vašemu systému, měli byste mít na paměti tyto skutečnosti týkající se bezpečnosti:

- Když v systému povolujete FTP, nemusí být vaše schéma oprávnění k objektům zajišťovat dostatečnou ochranu. Když má například uživatel oprávnění k prohlížení souboru (oprávnění \*USE), může si tento uživatel daný soubor zkopírovat na PC nebo do jiného systému. V případě potřeby můžete chránit některé soubory před kopírováním do jiného systému.
- Pomocí programů výstupních bodů FTP můžete omezit činnosti, které mohou uživatelé prostřednictvím FTP vykonávat. K určení, které činnosti povolíte, můžete použít výstupní bod Ověření platnosti požadavku FTP. Můžete například zamítnout požadavky GET na určité databázové soubory.
- Pomocí výstupního bodu Přihlášení k serveru můžete autentizovat uživatele, kteří se přihlašují k FTP serveru. Téma Konfigurace anonymního FTP popisuje použití programů výstupních bodů k nastavení podpory anonymního FTP v systému.
- Pokud nepoužíváte TLS/SSL, hesla pro FTP se při posílání mezi serverem a klientem nešifrují. V závislosti na používané metodě připojení může být systém vystaven riziku krádeže hesla při čmouchání po lince.
- Je-li systémová hodnota QMAXSGNACN nastavena na 1, vztahuje se se systémová hodnota QMAXSIGN na TELNET, ale nikoli na FTP. Je-li QMAXSGNACN nastavena na 2 nebo 3 (hodnoty, které zablokují profil v případě, kdy je dosaženo maximálního počtu přihlášení), počítají se pokusy o přihlášení prostřednictvím FTP. V tom případě může hacker zaútočit na službu prostřednictvím FTP tak, že se opakovaně pokouší přihlásit pod chybným heslem, dokud se uživatelský profil nezablokuje.
- U každého neúspěšného pokusu zaznamená systém do protokolu QHST zprávu CPF2234. Můžete napsat program, který bude monitorovat protokol QHST, zda neobsahuje tuto zprávu. Když tento program zjistí opakované pokusy, může FTP servery ukončit.
- Použijete-li při konfiguraci FTP parametr INACTTIMO (Časový limit při nečinnosti), sníží se riziko při ponechání relace FTP bez dozoru. Nezapomeňte se podívat do dokumentace nebo online nápovědy, abyste věděli, jak funguje parametr INACTTIMO společně s časovačem připojení, který spouští server.

**Poznámka:** Systémová hodnota QINACTITV nijak neovlivňuje relace FTP.

- Jestliže používáte podporu dávkového FTP, musí program odeslat na server uživatelské ID i heslo. Uživatelské ID i heslo musí být zakódovány v programu, nebo je program musí načíst ze souboru. Oba tyto způsoby uložení hesel a uživatelských ID představují potenciální bezpečnostní riziko. Jestliže používáte dávkový FTP, musíte chránit údaje o uživatelských ID a heslech přes zabezpečení objektů. Také byste měli používat ID jediného uživatele, který má v cílovém systému omezená oprávnění. Měl by mít pouze dostatečné oprávnění k vykonávání funkcí, které mu chcete povolit, například přenos souborů.
- FTP umožňuje používání vzdálených příkazů, stejně jako protokol APPC (Advanced Program-to-Program Communications) a program iSeries<sup>(TM)</sup> Access for Windows<sup>(R)</sup>. Podpříkaz FTP serveru RCMD (Vzdálený příkaz) je ekvivalentem použití příkazového řádku v systému. Než povolíte FTP, musíte zajistit, aby vaše schéma zabezpečení ochrany dat bylo dostatečné. Můžete také pomocí programu výstupního bodu FTP omezit nebo zamítnout pokusy o použití podpříkazu RCMD. Téma Programy výstupních bodů FTP popisuje tento výstupní bod a uvádí vzorové programy.
- Pomocí FTP může uživatel přistupovat k objektům v integrovaném systému souborů. Proto je třeba zajistit, aby schéma oprávnění pro integrovaný systém souborů bylo dostatečné, když v systému používáte FTP server.
- Oblíbenou činností hackerů je nastavit nic netušíci Web stanici jako zdroj informací. Někdy se může jednat o nezákonné nebo pornografické informace. Když hacker získá prostřednictvím FTP přístup

k vaší stanici, přenese tyto nežádoucí informace na váš server iSeries. Potom sdělí ostatním hackerům vaši FTP adresu. Ti potom pomocí FTP získají přístup k vašemu serveru iSeries a tyto nežádoucí informace si stahují.

Proti tomuto typu útoku se můžete chránit pomocí programů výstupních bodů FTP. Můžete například všechny požadavky na přenos informací na server směřovat do adresáře, který je určen pouze pro zápis. Tím zmaříte hackerův cíl, protože jeho přátelé si nebudou moci tyto informace z adresáře stáhnout. Více informací o nebezpečích a o možných řešeních, když je povolen přenos informací na server prostřednictvím FTP, najdete v publikaci AS/400<sup>(R)</sup> Internet Security: Protecting Your AS/400 from HARM in the Internet



## Zabezpečení FTP serveru pomocí SSL



FTP server poskytuje rozšířené zabezpečení ochrany dat při posílání souborů před nedůvěryhodnou sítí. K zabezpečení hesel a dalších důvěrných informací používá FTP server protokol SSL (Secure Sockets Layer). FTP server podporuje relace chráněné protokolem SSL nebo TLS, včetně autentizace klienta a automatického přihlašování (další informace o protokolech SSL a TLS najdete v tématu SSL concepts).

Většina aplikací s povoleným SSL připojuje klienta přes samostatné TCP porty, z nichž jeden je určen pro "nechráněné" relace a druhý pro zabezpečené relace. Zabezpečený FTP je však o něco více flexibilní. Klient se může připojit přes nezašifrovaný TCP port (obvykle to bývá TCP port 21) a potom vyjednat možnosti autentizace a šifrování. Klient také může zvolit chráněný FTP port (obvykle TCP port 990), kde se předpokládají připojení SSL. FTP server iSeries<sup>(TM)</sup> umožňuje oba způsoby.

Než přikročíte ke konfiguraci FTP serveru pro používání SSL, musíte mít na serveru iSeries nainstalované prerekvizitní programy a nastavené digitální certifikáty.

Ke konfiguraci zabezpečení FTP pomocí SSL je třeba splnit tyto úkoly:

1. Vytvoření lokálního vydavatele certifikátů (CA) nebo nakonfigurování FTP serveru pomocí DCM k používání veřejného certifikátu pro SSL.
2. Asociace certifikátu s FTP serverem.
3. Vyžádání autentizace klienta pro FTP server (volitelné).
4. Povolení SSL na FTP serveru.

Související informace najdete v tématu Zabezpečení FTP klienta pomocí TLS/SSL.



## Vytvoření lokálního vydavatele certifikátů (CA)



Pomocí produktu IBM<sup>(R)</sup> Digital Certificate Manager (DCM) můžete na serveru iSeries<sup>(TM)</sup> vytvořit a zprovoznit lokálního vydavatele certifikátů (CA). Lokální CA slouží k vydávání soukromých certifikátů pro aplikace, které spouštíte na svém serveru iSeries.

K vytvoření a zprovoznění lokálního CA na serveru iSeries pomocí DCM použijte tento postup:

1. Spusťte DCM.



2. V navigačním rámci DCM vyberte **Vytvoření vydavatele certifikátů (CA)**. Zobrazí se řada formulářů. Tyto formuláře vám pomohou vytvořit lokálního CA a provést i další kroky potřebné k tomu, abyste mohli začít používat digitální certifikáty pro SSL, pro podepisování objektů a pro ověřování podpisů.
3. Vyplňte všechny formuláře, které se zobrazí. Pro každý z úkolů potřebných k vytvoření a zprovoznění lokálního CA na serveru iSeries slouží jeden formulář. Při vyplňování formulářů postupujte takto:
  - a. Zvolte způsob uložení soukromého klíče k lokálnímu certifikátu CA. Tento krok je potřebný pouze tehdy, máte-li na serveru iSeries nainstalován šifrovací koprocesor (IBM 4758-023 PCI Cryptographic Coprocessor). Nemáte-li šifrovací koprocesor, DCM automaticky uloží certifikát i s jeho soukromým klíčem do paměti lokálních certifikátů CA.
  - b. Zadejte identifikační údaje lokálního CA.
  - c. Nainstalujte lokální certifikát CA na PC nebo do prohlížeče. To umožní, aby software rozpoznal lokálního CA a mohl ověřovat certifikáty, které tento CA vydává.
  - d. Zvolte strategická data pro lokálního CA.
  - e. Pomocí nového lokálního CA vydejte certifikát pro server nebo klienta, který budou aplikace používat k připojování prostřednictvím SSL. Máte-li na serveru iSeries nainstalován šifrovací koprocesor (IBM 4758-023 PCI Cryptographic Coprocessor), tento krok vám umožní vybrat i způsob uložení soukromého klíče k certifikátu serveru nebo klienta. Nemáte-li šifrovací koprocesor, DCM automaticky uloží certifikát i s jeho soukromým klíčem do paměti certifikátů \*SYSTEM. DCM vytvoří paměť certifikátů \*SYSTEM automaticky v rámci tohoto úkolu.
  - f. Vyberte aplikace, které budou moci používat tento certifikát serveru nebo klienta k připojování prostřednictvím SSL. Poznámka: Dbejte na to, abyste vybrali ID aplikace pro FTP server OS/400<sup>(R)</sup> (QIBM\_QTMF\_FTP\_SERVER).
  - g. Pomocí nového lokálního CA vydejte certifikát pro podepisování objektů, který budou aplikace moci používat k digitálnímu podepisování objektů. Přitom se vytvoří paměť certifikátů \*OBJECTSIGNING, která slouží ke správě certifikátů pro podepisování objektů. Poznámka: I když tento scénář nepočítá s používáním certifikátů pro podepisování objektů, přesto tento krok proveďte. Kdybyste nyní tento krok zrušili, celý úkol by se ukončil a další kroky k dokončení konfigurace certifikátů pro SSL byste museli provádět jednotlivě.
  - h. Vyberte aplikace, které budou důvěřovat vašemu lokálnímu CA. Poznámka: Dbejte na to, abyste vybrali ID aplikace pro FTP server OS/400 (QIBM\_QTMF\_FTP\_SERVER).

Více informací o certifikátech najdete v těchto souvisejících tématech:

#### Manage user certificates

Zde se dozvíte, jak mohou vaši uživatelé pomocí DCM získávat certifikáty nebo přidělit existující certifikáty svým uživatelským profilům na serveru iSeries.

#### Use APIs to programmatically issue certificates to non-iSeries users

Zde se dozvíte, jak prostřednictvím lokálního CA vydávat soukromé certifikáty pro uživatele, aniž by byl se tento certifikát přidělen konkrétnímu uživatelskému profilu na serveru iSeries.

#### Obtain a copy of the private CA certificate

Zde se dozvíte, jak získat kopii soukromého certifikátu CA a nainstalovat ji na PC, takže budete moci autentizovat veškeré certifikáty na serveru, které tento CA vydá.

### Co dělat dále:

Asociace certifikátu s FTP serverem.



## Asociace certifikátu s FTP serverem



Tento úkol proveďte, jestliže jste nepřiradili certifikát k aplikaci typu FTP server při vytváření lokálního vydavatele certifikátů (CA) nebo jestliže máte systém nakonfigurován tak, že vyžaduje certifikát od veřejného CA.

1. Spusťte IBM<sup>(R)</sup> DCM (Digital Certificate Manager). Chcete-li získat nebo vytvořit nějaké certifikáty, nebo jinak nastavit nebo změnit systém certifikátů, udělejte to teď. Informace o nastavení systému certifikátů najdete v tématu Using Digital Certificate Manager.
2. Klepněte na tlačítko **Výběr paměti certifikátů**.
3. Vyberte **\*SYSTEM**. Klepněte na **Pokračovat**.
4. Zadejte příslušné heslo pro paměť certifikátů \*SYSTEM. Klepněte na **Pokračovat**.
5. Po obnovení levého navigačního menu rozbalte položku **Správa aplikací**.
6. Klepněte na **Aktualizace přiřazení certifikátu**.
7. Na další obrazovce vyberte aplikaci typu **Server**. Klepněte na **Pokračovat**.
8. Vyberte **FTP server TCP/IP OS/400<sup>(R)</sup>**.
9. Klepněte na **Aktualizace přiřazení certifikátu**, abyste certifikát přiřadili k FTP serveru TCP/IP OS/400.
10. Ze seznamu vyberte certifikát, který chcete přiřadit k serveru.
11. Klepněte na **Přiřazení nového certifikátu**.
12. DCM znovu načte stranu **Aktualizace přiřazení certifikátu** s potvrzující zprávou. Když jsou certifikáty pro FTP server nastaveny, klepněte na **Provedeno**.

### Co dělat dále:

Vyžádání autentizace klienta pro FTP server (volitelné).  
nebo  
Povolení SSL na FTP serveru.



## Vyžádání autentizace klienta pro FTP server (volitelné)



Potřebujete-li FTP server k autentizaci klientů, můžete změnit specifikace aplikací v IBM<sup>(R)</sup> DCM (Digital Certificate Manager).

**Poznámka:** FTP server podporuje autentizaci klienta, ale FTP klient OS/400<sup>(R)</sup> nikoli. Někteří uživatelé stále ještě vyžadují autentizaci klienta, to však při připojování přes SSL vylučuje použití FTP klienta OS/400.

Je-li při připojování FTP klienta na serveru nastavena autentizace klienta, musí klient přesto odeslat i příkaz USER. Když je informace příkazu USER odeslána, FTP server zkontroluje, zda se tento uživatel shoduje s profilem s přiděleným certifikátem klienta, který klient odeslal na server při navázání spojení přes SSL. Když se uživatel s tímto certifikátem klienta shoduje, nevyžaduje se žádné heslo a FTP server přihlásí uživatele do systému. Příkaz USER je potřebný proto, že v protokolu FTP neexistuje žádný jiný způsob, jak "informovat" klienta, že je přihlášen.

1. Spusťte DCM (Digital Certificate Manager). Chcete-li získat nebo vytvořit nějaké certifikáty, nebo jinak nastavit nebo změnit systém certifikátů, udělejte to teď. Informace o nastavení systému certifikátů najdete v tématu Using Digital Certificate Manager.
2. Klepněte na tlačítko **Výběr paměti certifikátů**.
3. Vyberte **\*SYSTEM**. Klepněte na **Pokračovat**.

4. Zadejte příslušné heslo pro paměť certifikátů \*SYSTEM. Klepněte na **Pokračovat**.
5. Po obnovení levého navigačního menu rozbalte položku **Správa aplikací**.
6. Klepněte na **Aktualizace definice aplikace**.
7. Na další obrazovce vyberte aplikaci typu **Server**. Klepněte na **Pokračovat**.
8. Vyberte **FTP server TCP/IP OS/400**.
9. Klepněte na **Aktualizace definice aplikace**.
10. V tabulce, která se objeví, vyberte **Ano**, že bude vyžadována autentizace klienta.
11. Klepněte na **Použít**.
12. DCM znovu načte stranu **Aktualizace definice aplikace** s potvrzující zprávou. Když je definice aplikace pro FTP server zaktualizována, klepněte na **Provedeno**.

#### Co dělat dále:

Povolení SSL na FTP serveru.



### Povolení SSL na FTP serveru



K povolení SSL na FTP serveru použijte tento postup:

1. V produktu iSeries<sup>(TM)</sup> Navigator rozbalte **váš server iSeries** → **Síť** → **Servery** → **TCP/IP**.
2. Klepněte pravým tlačítkem myši na **FTP**.
3. Vyberte **Vlastnosti**.
4. Vyberte ouško **Obecné**.
5. Vyberte jednu z následujících voleb podpory SSL:

- **Pouze zabezpečené**

Když vyberete tuto volbu, budou na FTP serveru povoleny pouze relace SSL. Připojení se mohou uskutečnit i přes nezabezpečený FTP port, ale FTP klient musí nejprve vyjednat relaci SSL, a teprve potom se uživatel může připojit.

- **Pouze nezabezpečené**

Když vyberete tuto volbu, nebudou na FTP serveru relace SSL povoleny. Veškeré pokusy o připojení přes SSL port budou neúspěšné.

- **Zabezpečené i nezabezpečené**

Na FTP serveru budou povoleny zabezpečené i nezabezpečené relace.

#### Poznámka:

FTP server nemusíte znovu spouštět. Sám si dynamicky zjistí, že k němu byl přiřazen certifikát. Jestliže server tuto změnu dynamicky nezjistil, zkontrolujte, zda máte na serveru iSeries aplikovaná nejnovější PTF.



### Zabezpečení FTP klienta pomocí TLS/SSL



Protokoly TLS/SSL (Transport Layer Security / Secure Sockets Layer) umožňují šifrovat data přenášená prostřednictvím řídicích a datových připojení. Primárním účelem řídicího připojení je skrytí hesla při přihlašování k FTP serveru.

Než budete moci pomocí FTP klienta navazovat zabezpečená připojení k serverům, je třeba pomocí DCM nakonfigurovat důvěryhodné vydavatele certifikátů pro FTP klienta. Musíte přidat všechny vydavatele certifikátů použité k vydání certifikátů, které máte přiřazeny k serverům, k nimž se chcete připojovat. Podle použitého vydavatele certifikátů (CA) může být nutný export nebo import certifikátů CA. Více informací o důvěryhodných vydavatelích certifikátů najdete pod DCM v tématu Define a CA trust list for an application.

Zvolíte-li šifrování TLS/SSL pro řídicí připojení, FTP klient standardně šifruje i data posílaná prostřednictvím datového připojení FTP. Protokol FTP nepovoluje provádět zabezpečené datové připojení bez zabezpečeného řídicího připojení.

Šifrování může mít značný vliv na výkon a u datového připojení je můžete vynechat. To vám umožňuje přenášet soubory, jež nejsou důvěrné, bez snížení výkonu, a přitom stále chránit bezpečnost systému utajením hesel.

FTP klient má parametry pro CL příkaz STRTCPFTP a podpříkazy, které jsou součástí podpory TLS/SSL (SECOpen a SECDData).

## **Nastavení ochrany TLS/SSL u FTP klienta iSeries<sup>(TM)</sup>**

### **Řídicí připojení**

Ochrana TLS/SSL může být nastavena v příkazu STRTCPFTP a podpříkazu SECOPEN implicit.

Chcete-li nastavit zabezpečené řídicí připojení, zadejte v příkazu STRTCPFTP (FTP) do parametru SECCNN (zabezpečené připojení) hodnotu \*SSL. Můžete také zadat hodnotu \*IMPLICIT, abyste získali zabezpečené připojení na předdefinovaném portu serveru. (Podrobnější informace najdete dále v odstavci Implicitní připojení SSL.)

V rámci relace FTP klienta můžete nastavit zabezpečené řídicí připojení pomocí podpříkazu SECOPEN.

### **Datové připojení**

Chcete-li nastavit zabezpečené datové připojení, zadejte v příkazu STRTCPFTP (FTP) do parametru DTAPROT (zabezpečení dat) hodnotu \*PRIVATE. Zadáte-li do parametru DTAPROT hodnotu \*CLEAR, budou se data posílat nezašifrovaná.

Jestliže máte zabezpečené řídicí připojení, můžete pomocí příkazu SECDATA změnit úroveň zabezpečení datového připojení.

### **Implicitní připojení SSL**

Některé FTP servery podporují tak zvané implicitní připojení SSL. Toto připojení poskytuje stejnou ochranu jako volba \*SSL, ale může se uskutečnit pouze na předem určeném portu serveru, obvykle 990, a server musí být nastaven tak, aby na tomto portu očekával vyjednávání připojení SSL/TLS.

Tato metoda nabízí bezpečné připojení pro takové implementace FTP, které nepodporují standardní protokol poskytování ochrany TLS/SSL.

Mnohé starší implementace podpory SSL používaly tento implicitní přístup, nyní se však již nedoporučuje a asociace IETF jej zamítá.

**Poznámka:**

Standardní protokol pro nastavení připojení TLS/SSL vyžaduje při navázání spojení se serverem podpříkaz AUTH (Oprávnění). K nastavení úrovně ochrany dat slouží také podpříkazy PBSZ a PROT.

U implicitního připojení SSL však **nebudou** podpříkazy AUTH, PBSZ a PROT na server odeslány. Server se bude chovat tak, jako kdyby klient poslal tyto podpříkazy s následujícími parametry:

- AUTH SSL
- PBSZ 0
- PROT P



## Správa přístupu pomocí programů výstupních bodů FTP

FTP poskytuje úroveň zabezpečení na základě zabezpečení objektů OS/400<sup>(TM)</sup>. To znamená, že se žádný vzdálený uživatel nemůže přihlásit na váš FTP server iSeries<sup>(TM)</sup>, pokud nemá platný uživatelský profil a heslo.

Další ochranu vytvoříte, když k výstupním bodům FTP serveru a klienta přiřadíte programy výstupních bodů FTP, kterými ještě více omezíte přístup ke svému systému prostřednictvím FTP. Můžete například omezit možnost přihlášení k FTP, stejně jako přístup ke knihovnám, objektům a k použití příkazů.

Můžete napsat program výstupního bodu Ověření platnosti požadavku na FTP serveru, a tak omezit CL příkazy a FTP podpříkazy, které mohou uživatelé používat. Pokyny a příklady najdete v tématu Výstupní bod Ověření platnosti požadavku: klient a server.

Můžete řídit autentizaci uživatelů na aplikačním serveru TCP/IP pomocí programu výstupního bodu Přihlášení k serveru.

Můžete napsat program Ověření platnosti požadavku na FTP klientu pro výstupní bod klienta: Ověření požadavku. Tento program řídí, které z funkcí FTP klienta může uživatel provádět.

Podle svých konkrétních podmínek můžete použít omezení přístupu k podpříkazům FTP také pomocí Administrativy aplikací, namísto psaní programů pro výstupní body Ověření platnosti požadavku na FTP serveru a Ověření platnosti požadavku na FTP klientu.

Aby vaše programy výstupních bodů správně fungovaly, je třeba provést Instalaci a registraci programů výstupních bodů. Jestliže tyto programy již nebudete potřebovat, je třeba zabránit jejich opětovnému použití tak, že provedete Odstranění programů výstupních bodů.

## Správa přístupu pomocí produktu iSeries<sup>(TM)</sup> Navigator

K omezení přístupu uživatelů k funkcím FTP serveru a klienta můžete použít iSeries Navigator. K udělení nebo zabránění přístupu k funkcím pro jednotlivé uživatele nebo skupiny uživatelů použijte Administrativu aplikací. Případně můžete spravovat přístup k funkcím FTP pomocí vlastních programů výstupního bodu Ověření platnosti požadavku FTP.

Ke správě přístupu uživatelů k funkcím FTP pomocí produktu iSeries Navigator použijte tento postup:

1. V produktu iSeries Navigator klepněte pravým tlačítkem myši na **váš server iSeries** a vyberte **Administrativa aplikací**.
2. Vyberte ouško **Hostitelské aplikace**.
3. Rozbalte **TCP/IP Utilities for iSeries**.

4. Rozbalte **FTP (File Transfer Protocol)**.
5. Rozbalte **FTP klient** nebo **FTP server**.
6. Vyberte funkci, ke které chcete povolit nebo zakázat přístup.
7. Klepněte na **Přizpůsobit**.
8. Pomocí dialogu **Přizpůsobení využití** upravte seznam uživatelů a skupin, jež mají povolen nebo zamezen přístup k této funkci.
9. Klepnutím na **OK** uložte změny provedené na straně **Přizpůsobení využití**.
10. Klepnutím na **OK** opusťte stranu **Administrativa aplikací**.

Jiný způsob, jak spravovat přístup určitých uživatelů nebo skupiny k registrovaným funkcím FTP, je přes nástroj pro správu uživatelů a skupin produktu iSeries Navigator. K tomu slouží následující postup:

1. V produktu iSeries Navigator rozbalte **váš server iSeries** → **Uživatelé a skupiny**.
2. Vyberte **Všichni uživatelé** nebo **Skupiny**.
3. Klepněte pravým tlačítkem myši na uživatele nebo skupinu a vyberte **Vlastnosti**.
4. Klepněte na **Schopnosti**.
5. Klepněte na **Aplikace**.

Zde můžete u vybraného uživatele nebo skupiny měnit nastavení pro uvedené funkce. Můžete také změnit nastavení pro všechny funkce v hierarchickém seskupení tak, že změníte nastavení pro "nadřazenou" funkci.

Další informace týkající se zabezpečení FTP serveru iSeries najdete v tématu Zabezpečení ochrany FTP.

## Monitorování příchozích uživatelů FTP

Protokolováním a kontrolou využívání FTP můžete monitorovat činnost a zjistit případná napadení z vnějšku. K monitorování příchozích uživatelů FTP použijte tento postup:

1. V produktu iSeries<sup>(TM)</sup> Navigator rozbalte **váš server** → **Síť** → **Servery** → **TCP/IP**.
2. V pravém okně klepněte pravým tlačítkem myši na **FTP** a vyberte **Úlohy serveru**.
3. Objeví se dialogové okno pro úlohu FTP serveru. Ve sloupci Aktuální uživatel je uveden uživatel přihlášený k dané úloze serveru. Není-li přihlášen žádný uživatel, je zde uvedeno Qtcp. Stisknutím klávesy **F5** nebo výběrem **Zobrazení** → **Obnovit** můžete toto okno aktualizovat.

Formát jména těchto úloh je *QTFTPnnnnn*, kde *nnnnn* je náhodně generované číslo.

Ke spuštění úloh FTP serveru použijte postup uvedený v tématu Spuštění FTP serveru.

---

## Správa FTP serveru

Server iSeries<sup>(TM)</sup> lze nastavit tak, aby odesílal, přijímal a sdílel soubory v sítích prostřednictvím protokolu FTP (File Transfer Protocol). FTP se skládá ze dvou částí: FTP klienta a FTP serveru. Vy pracujete s FTP klientem. FTP klient spolupracuje s FTP serverem. S FTP serverem obvykle nepracujete přímo. Následující témata obsahují postupy správy FTP serveru:

- Spuštění a zastavení FTP serveru.
- Nastavení počtu dostupných FTP serverů.
- Zvýšení výkonu FTP podporou konfigurovatelného podsystému.

## Spuštění a zastavení FTP serveru

FTP server lze spustit nebo zastavit pomocí produktu iSeries<sup>(TM)</sup> Navigator. Pokyny týkající se přístupu k FTP najdete v tématu Přístup k FTP pomocí produktu iSeries Navigator.

Ke spuštění FTP serveru použijte tento postup:

1. V produktu iSeries Navigator rozbalte **váš server iSeries** → **Síť** → **Servery** → **TCP/IP**.
2. V pravém okně klepněte pravým tlačítkem myši na **FTP** a vyberte **Spustit**.

K zastavení FTP serveru použijte tento postup:

1. V produktu iSeries Navigator rozbalte **váš server iSeries** → **Síť** → **Servery** → **TCP/IP**.
2. V pravém okně klepněte pravým tlačítkem myši na **FTP** a vyberte **Zastavit**.

## Nastavení počtu dostupných FTP serverů



Můžete určit minimální počet dostupných serverů, které budou stále k dispozici pro připojování klientů. Tuto hodnotu můžete nastavit na straně **Vlastnosti FTP** tak, že do pole **Výchozí počet serverů ke spuštění** zadáte číslo v rozmezí 1 až 20. Zadáte-li hodnotu 1, budou příchodí připojení k FTP serveru zpožděná. Doporučená hodnota je 3.

Když se klient připojuje k FTP serveru iSeries<sup>(TM)</sup>, server zkontroluje počet aktivních serverů, k nimž není připojen žádný klient, a porovná tento počet s hodnotou uvedenou jako výchozí počet serverů ke spuštění. Jestliže je hodnota výchozího počtu serverů vyšší než skutečný počet dostupných serverů, spustí se další servery, aby se obě tyto hodnoty vyrovnaly. Jestliže je hodnota výchozího počtu serverů nižší než skutečný počet dostupných serverů, neprovede se nic. Změna hodnoty výchozího počtu serverů se projeví hned při příštím připojení klienta, kdy se aktivuje výše uvedený proces.

Je-li například v současné chvíli aktivních 5 relací FTP klienta a výchozí počet serverů je nastaven na 10, je spuštěno 15 FTP serverů. Tento počet se skládá z pěti serverů, které obsluhují aktivní relace klientů, a deseti dalších dostupných serverů. Skutečný počet dostupných serverů může být větší než hodnota výchozího počtu serverů. Jestliže v našem příkladu nebude aktivních oněch pět relací klientů a nebudou spuštěny žádné další relace, bude v systému k dispozici 15 dostupných serverů.



## Zvýšení výkonu FTP serveru podporou konfigurovatelného podsystému

Mnoho úloh serveru IBM využívá předvolený podsystém (QSYS/QSYSWRK). Použijete-li jiný podsystém než tento předvolený, může se zvýšit výkon, protože se sníží potřeba sdílení prostředků.

Ke konfiguraci podsystému pro FTP server použijte tento postup:

1. V produktu iSeries<sup>(TM)</sup> Navigator rozbalte **váš Server iSeries** → **Síť** → **Servery** → **TCP/IP**.
2. Klepněte pravým tlačítkem myši na **FTP** a vyberte **Vlastnosti**.
3. Na straně **Vlastnosti FTP** vyberte **Popis podsystému**.
4. Zadejte popis podsystému a předem nadefinovanou knihovnu.

Jestliže zadaný podsystém dosud neexistuje, FTP ho vytvoří spolu s položkami směrovací tabulky a s popisy úloh. Úloha spuštění serveru při svém zpracování specifikuje parametry tohoto nově vytvořeného podsystému, a v tomto podsystému potom spustí úlohy serveru s dávkovým spuštěním.

---

## Používání FTP klienta u serveru iSeries<sup>(TM)</sup>

FTP klient umožňuje přenášet soubory, které se nacházejí na serveru iSeries včetně těch, které jsou uloženy v systémech souborů "kořenový", QSYS.Lib, QOpenSys, QOPT a QFileSvr.400. Umožňuje také přenášet pořadače a dokumenty ze systému souborů QDLS. FTP klienta můžete spouštět v neobsluhovaném režimu dávkového zpracování, kdy se podpříkazy klienta čtou ze souboru a odezvy na tyto podpříkazy se zapisují rovněž do souboru. FTP klient obsahuje i další funkce pro manipulaci se soubory v systému.



Klient má uživatelské rozhraní, ze kterého zadáváte podpříkazy klienta, pomocí nichž vytváříte požadavky na FTP server. Výsledky těchto požadavků se potom zobrazují.

Aby bylo možné přenášet soubory mezi klientem a serverem, vytvoří se dvě připojení. Řídící připojení se používá k požadavkům na služby serveru zadávané pomocí příkazů FTP serveru. Server vrací klientu svoje odpovědi, které vás informují, jak je požadavek zpracováván. Druhé, tak zvané datové připojení se používá k přenosu seznamů souborů a skutečných dat souborů.

Klient i server mají funkci pro přenos dat, která je propojena s rezidentním systémem souborů. Tyto funkce čtou nebo zapisují data do lokálních systémů souborů a datových připojení.

### **Spuštění a ukončení relace klienta**

Popisuje, jak spustit a ukončit relaci klienta.

### **Pokyny pro časovou prodlevu serveru**

Popisuje, jak zabránit překročení časového limitu při připojení.

### **Přenos souborů prostřednictvím FTP**

Popisuje, jak odesílat a přijímat soubory pomocí FTP.

### **FTP jako dávková úloha**

Uvádí příklady spuštění FTP v neobsluhovaném režimu.

## **Spuštění a ukončení relace klienta**

Toto téma uvádí podrobné informace o používání FTP klienta na serveru iSeries<sup>(TM)</sup>.

Spuštění relace FTP klienta (See 24)

Ukončení relace FTP klienta (See 27)

### **Spuštění relace FTP klienta**

Než spustíte funkci FTP klienta, musíte znát následující údaje:

- Jméno nebo internetovou adresu systému, do něhož nebo z něhož se soubory posílají.
- Přihlašovací ID a heslo (je-li požadováno) pro vzdálený systém, kde má dojít k přenosu souborů.
- Jméno souboru nebo souborů, s nimiž chcete pracovat (například odesílat a přijímat).

Příkaz STRTCPFTP "vzdálenýsystém" (Spuštění TCP/IP FTP) spustí relaci klienta na lokálním serveru iSeries a potom v zadaném vzdáleném systému otevře připojení k FTP serveru. Zadáte-li například příkaz FTP myserver.com, spustí se na serveru iSeries relace klienta, která otevře připojení k FTP serveru ve vzdáleném systému myserver.com. Můžete zadat i další parametry, nebo si nechat zobrazit výzvy k jejich zadání tak, že napíšete příkaz STRTCPFTP bez uvedení vzdáleného systému.

```
Spuštění přenosu soub. TCP/IP (STRTCPFTP)

Zapište volby, stiskněte Enter.

Vzdálený systém. . . . . > MYSERVER.COM

Identifikátor kód. znak. sady      *DFT      1-65533, *DFT
Port . . . . . > *SECURE          1-65535, *DFT, *SECURE
Bezpečně spojení . . . . . *DFT    *DFT, *NONE, *SSL, *IMPLICIT
Ochrana dat. . . . . *DFT        *DFT, *CLEAR, *PRIVATE
```



Když zadáte jméno vzdáleného systému, budete vyzváni k zadání dalších údajů. V následujícím textu je uveden přehled dostupných voleb. Další podrobnosti najdete v nápovědě k polím:

### **RMTSYS (Vzdálený systém)**

Udává jméno vzdáleného systému do něhož nebo z něhož se soubory přenášejí. Možné hodnoty jsou:

#### **\*INTERNETADR**

Je požadován parametr INTERNETADR (Internetová adresa). Internetová adresa se udává ve formátu nnn.nnn.nnn.nnn, kde nnn je dekadické číslo v rozmezí 0 až 255.

#### **vzdálený-systém**

Zadejte jméno vzdáleného systému do něhož nebo z něhož se soubory přenášejí.

### **CCSID (Identifikátor kódové sady znaků)**

Udává znakovou sadu ASCII, která slouží k přenosu souborů ve formátu SBCS (jednobajtová znaková sada) ASCII, když je režim přenosu TYPE nastaven na ASCII. Možné hodnoty jsou:

#### **\*DFT**

Použije se hodnota CCSID 00819 (ISO 8859-1 8bitová ASCII).

#### **hodnota-CCSID**

Použije se požadovaná hodnota CCSID. Tato hodnota se ověřuje, aby bylo zajištěno, že je požadován platný CCSID SBCS ASCII.

### **PORT (Port)**

Udává číslo portu, který se použije k připojení k FTP serveru. Obvykle se k připojování k FTP serveru používá "známý" port 21. Za určitých okolností může být FTP server kontaktován i přes jiný port než 21. V těchto případech zadejte do parametru PORT číslo portu, přes který se k serveru připojujete. Možné hodnoty jsou:

#### **\*DFT**

Použije se hodnota 00021.

#### **\*SECURE**

Použije se hodnota 00990. Port 990 je vyhrazen pro zabezpečené FTP servery, které k šifrování dat bezprostředně používají protokoly TLS (Transport Layer Security) nebo SSL (Secure Sockets Layer).

#### **hodnota\_portu**

Použije se požadovaná hodnota portu. Tato hodnota se ověřuje, aby bylo zajištěno, že je v platném rozsahu.

#### **Poznámka:**

Zadáte-li hodnotu 990, FTP klient bude provádět stejné funkce, jako kdybyste zadali \*SECURE.



### **SECCNN (Zabezpečené připojení)**

Udává typ mechanismu zabezpečení, který se má použít k ochraně informací přenášených řídicím připojením FTP (zahrnuje heslo používané k autentizaci relace s FTP serverem). TLS (Transport Layer Security) a SSL (Secure Sockets Layer) jsou kompatibilní protokoly, které pomocí šifrování chrání data před prohlížením během přenosu a kontrolují, zda nedošlo ke ztrátě nebo poškození dat.

#### **Poznámka:**

Pomocí příkazu FTP klienta SECOPEN můžete otevřít zabezpečené připojení FTP během relace FTP klienta.

Možné hodnoty jsou:

**\*DFT**

Když je v parametru PORT uvedeno \*SECURE nebo 990, použije se hodnota \*IMPLICIT; jinak se použije hodnota \*NONE.

**\*IMPLICIT**

FTP klient se při připojení k zadanému FTP serveru pokusí bezprostředně použít TLS/SSL (aniž by na server odeslal podpříkaz AUTH). Jestliže server nepodporuje implicitní TLS/SSL pro zadaný port nebo z nějakého důvodu selže vyjednávání TLS/SSL, spojení se ukončí.

**\*SSL**

Po připojení k zadanému FTP serveru odešle FTP klient podpříkaz AUTH (Oprávnění), kterým žádá o chráněnou relaci TLS/SSL. Jestliže server podporuje TLS/SSL, provede se vyjednání TLS/SSL. Jestliže server TLS/SSL nepodporuje nebo selže vyjednávání TLS/SSL, spojení se ukončí.

**\*NONE**

FTP klient nebude používat šifrování řídicího připojení k zadanému FTP serveru.



**DTAPROT (Ochrana dat)**

Udává typ ochrany dat, který se má použít pro informace přenášené prostřednictvím datového připojení. Toto připojení slouží k přenosu dat souborů a výpisů adresářů. Protokol FTP nepovolí ochranu datového připojení, jestliže není nastavena ochrana řídicího připojení.

**Poznámka:**

Následně můžete pomocí podpříkazu FTP klienta SECDATA, změnit úroveň ochrany dat. Po vytvoření zabezpečeného řídicího připojení FTP klient pomocí podpříkazu PROT požádá o vámi zadanou ochranu dat.

Možné hodnoty jsou:

**\*DFT**

Jestliže je v parametru SECCNN zadáno chráněné řídicí připojení, použije se hodnota \*PRIVATE; jinak e použije hodnota \*CLEAR.

**\*PRIVATE**

Informace posílané přes datové připojení FTP budou šifrovány. Jestliže je v parametru SECCNN uvedeno, že řídicí připojení FTP není šifrováno, nelze hodnotu \*PRIVATE zadat.

**\*CLEAR** Informace posílané přes datové připojení FTP nebudou šifrovány.



**TBLFTPOUT (Odchozí mapovací tabulka ASCII/EBCDIC)**

Udává objekt tabulky, která na FTP klientu mapuje veškerá odchozí data. Odchozí data jsou mapována z formátu EBCDIC na ASCII. Není-li v parametru TBLFTPOUT uveden žádný objekt tabulky, použije se k určení způsobu mapování odchozích dat parametr CCSID. Možné hodnoty jsou:

**\*CCSID**

K určení způsobu mapování odchozích dat se použije parametr CCSID.

**\*DFT**

K určení způsobu mapování odchozích dat se použije parametr CCSID.

Jméno odchozí mapovací tabulky může být odvozeno podle jedné z následujících hodnot knihovny:

**\*LIBL**

Prohledávají se všechny knihovny z uživatelské i systémové části seznamu knihoven úlohy, dokud se nenarazí na první shodu.

**\*CURLIB**

Prohledává se aktuální knihovna úlohy. Není-li uvedena žádná aktuální knihovna pro danou úlohu, použije se knihovna QGPL.

**jméno-knihovny**

Zadejte jméno knihovny, která se má prohledat.

**odchozí-mapovací-tabulka**

Zadejte objekt tabulky, který má použít FTP klient k mapování odchozích dat.

**TBLFTPIN (Příchozí mapovací tabulka ASCII/EBCDIC)**

Udává objekt tabulky, která na FTP klientu mapuje veškerá příchozí data. Příchozí data se mapují z formátu ASCII na EBCDIC. Není-li v parametru TBLFTPIN uveden žádný objekt tabulky, použije se k určení způsobu mapování příchozích dat parametr CCSID. Možné hodnoty jsou:

**\*CCSID**

K určení způsobu mapování příchozích dat se použije parametr CCSID.

**\*DFT**

K určení způsobu mapování příchozích dat se použije parametr CCSID.

Jméno příchozí mapovací tabulky může být odvozeno podle jedné z následujících hodnot knihovny:

**\*LIBL**

Prohledávají se všechny knihovny z uživatelské i systémové části seznamu knihoven úlohy, dokud se nenarazí na první shodu.

**\*CURLIB**

Prohledává se aktuální knihovna úlohy. Není-li uvedena žádná aktuální knihovna pro danou úlohu, použije se knihovna QGPL.

**jméno-knihovny**

Zadejte jméno knihovny, která se má prohledat.

**příchozí-mapovací-tabulka**

Zadejte objekt tabulky, který má použít FTP klient k mapování příchozích dat.

Postup při přenosu souborů mezi systémy najdete v tématu Přenos souborů pomocí FTP.

**Ukončení relace FTP klienta**

FTP relace se ukončí podpříkazem QUIT. Podpříkaz QUIT uzavře spojení se vzdáleným hostitelem a ukončí relaci FTP na serveru iSeries. Můžete také stisknout klávesu F3 (Exit) a potvrdit ukončení relace FTP klienta.

## Pokyny pro časovou prodlevu serveru



S hodnotou časové prodlevy při nečinnosti souvisí určitá opatření. Tato hodnota udává délku času nečinnosti FTP serveru (v sekundách), po jehož uplynutí server relaci ukončí. Některé vzdálené servery umožňují klientu tuto hodnotu změnit. Server iSeries<sup>(TM)</sup> například podporuje podpříkaz FTP serveru TIME, který může být poslán na server prostřednictvím podpříkazu FTP klienta QUOTE, jak je popsáno v tématu QUOTE (Odeslat podpříkaz na FTP server). Servery s operačním systémem UNIX<sup>(R)</sup> často podporují podpříkaz SITE IDLE.

Když používáte podpříkazy lokálního serveru iSeries prostřednictvím podpříkazu SYSCMD nebo F21, nedochází k interakci mezi klientem a serverem. Pokud tedy doba zpracování těchto příkazů lokálního serveru iSeries překročí časovou prodlevu při nečinnosti, server spojení ukončí. Ztratíte-li spojení, musíte se znovu přihlásit k serveru pomocí příkazu OPEN (OPEN <jméno-vzdáleného-systému>) a USER, jak je popsáno v poznámce k Přihlášení ke vzdálenému systému (serveru).



## Přenos souborů prostřednictvím FTP

K přenosu souborů prostřednictvím FTP použijte následující postup.

1. Shromážděte tyto informace:
  - TCP/IP jméno nebo IP adresu vzdáleného počítače
  - Přihlašovací jméno a heslo pro vzdálený počítač (pokud vzdálený počítač nepodporuje anonymní FTP).
  - Jméno a umístění souboru, který chcete přenášet.
  - Umístění cíle.
  - Typ přenosu souboru, který budete používat: ASCII, EBCDIC nebo BINARY
  - Zda budete používat zabezpečené připojení TLS (Transport Layer Security) nebo SSL (Secure Sockets Layer).
2. Na příkazový řádek napište **FTP** a stiskněte klávesu **Enter**.
3. Na náznak napište TCP/IP jméno nebo IP adresu vzdáleného počítačového systému a stiskněte klávesu **Enter**. Můžete použít buď jméno, nebo IP adresu, například:  
remote.systemname.com  
nebo  
110.25.9.13
4. Zadejte identifikátor kódové sady znaků (CCSID). Pokud nepožadujete žádný určitý CCSID, použijte předvolenou hodnotu (\*DFT).
- 5.



Chcete-li použít zabezpečené připojení, které bude chránit vaše hesla a data, zadejte do parametru Port hodnotu \*SECURE.



6. Stisknutím klávesy **Enter** navažte spojení. FTP klient zobrazí zprávy oznamující, že spojení se vzdáleným systémem bylo úspěšně navázáno.



**Poznámka:**

Jestliže zadáte hodnotu \*SECURE pro port, u něhož server nepodporuje implicitní TLS/SSL, nebo z nějakého důvodu selže vyjednávání TLS/SSL, spojení se ukončí.



7. Chcete-li změnit typ přenosu souborů, postupujte takto:
  - a. Přepnutí na EBCDIC: Před zahájením přenosu souboru napište **EBCDIC** a stiskněte klávesu **Enter**.
  - b. Přepnutí na BINARY: Před zahájením přenosu souboru napište **BINARY** a stiskněte klávesu **Enter**.
  - c. Přepnutí zpět na předvolený typ ASCII: Před zahájením přenosu souboru napište **ASCII** a stiskněte klávesu **Enter**.
8. Nyní můžete začít přenášet soubory:
  - a. Zadejte **CD** a jméno adresáře. Stiskněte klávesu **Enter**.
  - b. Proveďte jeden z následujících kroků:
    - K přenosu souboru ze systému serveru do systému klienta zadejte podpříkaz **GET** následovaný jménem souboru:  
GET myfile.txt
    - K odeslání souboru ze systému klienta do systému serveru zadejte podpříkaz **PUT** následovaný jménem souboru:  
PUT myfile.txt
9. K ukončení relace FTP klienta a návrat na příkazový řádek iSeries<sup>(TM)</sup> zadejte podpříkaz FTP **QUIT**.

## FTP jako dávková úloha

Kromě interaktivního spouštění FTP klienta můžete FTP klienta spouštět také v neobsluhovaném režimu. Toto téma uvádí dva příklady této metody: jednoduchý (See 29) příklad a složitější souhrnný (See 30) příklad. Další příklad můžete najít v části 6.7 Batch FTP v publikaci V4 TCP/IP for AS/400<sup>(R)</sup>: More Cool Things Than Ever



(cca 744 stran).

Dávkový FTP: Jednoduchý příklad

Zde je jednoduchý příklad dávkového přenosu souboru, kterým se uskuteční úspěšný přenos jednoho souboru ze vzdáleného systému.

Úkol má následující součásti:

- CL program
- vstupní soubor příkazů FTP
- výstupní soubor zpráv FTP

### CL program

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

**Poznámka:**

Chcete-li použít tuto ukázkou, když je napsaná v jazyce ILECL, musíte k příkazům OVRDBF přidat ještě OVRSCOPE(\*CALLLVL).

Program BATCHFTP změní hodnotu parametru INPUT na zdrojový fyzický soubor ITSOLIB1/QCLSRC MBR(FTPCMDS). Výstup je posílán do MBR(OUT).

**Vstupní soubor příkazů**

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

Požadované FTP podpříkazy jsou uvedeny v souboru FTPCMDS.

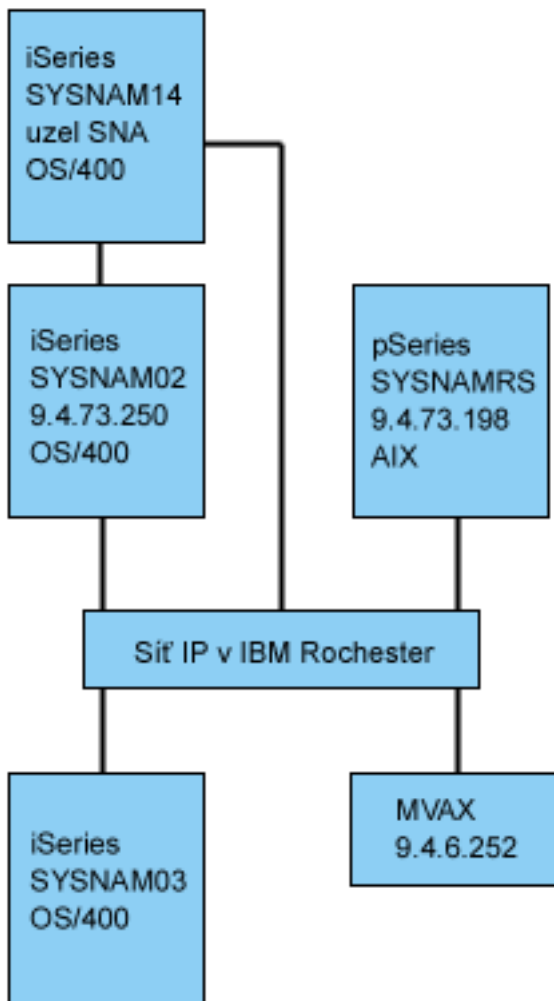
**Výstupní soubor zpráv**

```
*****
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.
*****
```

Je zobrazen výstupní soubor. Není těžké napsat program, který zpracuje tento soubor a zobrazí chybové zprávy na QSYSOPR, pokud tento soubor nějaké chybové zprávy obsahuje. Chybové zprávy FTP mají čísla, která začínají číslicí 4 nebo 5.

Dávkový FTP: Souhrnný příklad

Tento příklad ukazuje, jak načíst soubory z několika vzdálených hostitelských systémů na centrální server iSeries<sup>(TM)</sup> v dávkovém režimu:



\*

Uživatel GWIL na serveru iSeries SYSNAM03 chce provést tyto kroky:

1. Načíst soubory z hostitelských systémů SYSNAMRS (RS/6000<sup>(R)</sup>) a MVAX (VAX).
2. Po načtení souboru ze SYSNAMRS přenést tento soubor pomocí FTP na SYSNAM02 (jiný server iSeries).
3. Odtud se má soubor odeslat pomocí TCP/IP na server iSeries SYSNAM14.

#### Vytvoření CL programu ke spuštění FTP

1. Jak jsme viděli u předchozího příkladu, FTP používá ke vstupu příkazů (INPUT) a výstupu zpráv (OUTPUT) obrazovku, což je nutné pro dávkový režim změnit. Použijeme příkaz OVRDBF k přepsání těchto souborů takovými, které budeme moci použít v dávkovém zpracování:

```

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

2. Povinným parametrem příkazu STRTCPFTP obsaženém v souboru CL programů je jméno hostitele nebo jeho internetová adresa. Avšak jestliže chcete zadat vzdálené systémy do vstupního souboru příkazů namísto do souboru CL programů, musíte do příkazu STRTCPFTP zadat fiktivní jméno hostitele, abyste dodrželi povinnou syntaxi. Toto fiktivní jméno může být jméno fiktivního hostitele i jméno skutečného hostitele. Jedná-li se o skutečného hostitele, pak první položkou ve vstupním souboru

příkazů musí být uživatelské ID a heslo a druhou položkou musí být podpříkaz CLOSE. Nejedná-li se o skutečného hostitele, tyto položky nejsou povinné a první položkou může být podpříkaz OPEN, který provede připojení k požadovanému serveru.

```
FTP RMTSYS(LOOPBACK)
```

FTP zpracuje vstupní soubor a zapíše zprávy do výstupního souboru (FTPLOG).

- Po ukončení FTP aplikace vymažte všechny dočasné změny hodnot parametrů:

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

CL program pro dávkové zpracování FTP bude vypadat jako v následující ukázce ze systému SYSNAM01:

```
Columns . . . : 1 71          Browse          GERRYLIB/QCLSRC
SEU==>          FTPBATCH
FMT **  ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
***** Beginning of data *****
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
***** End of data *****

F3=Exit  F5=Refresh  F9=Retrieve  F10=Cursor  F12=Cancel
F16=Repeat find      F24=More keys

(C) COPYRIGHT IBM CORP. 1981, 1994.
```

Obrázek 1. CL program FTPBATCH pro dávkový FTP.

### Vytvoření vstupního souboru FTP (FTPCMDS)

Tento soubor by měl obsahovat všechny podpříkazy FTP klienta, které jsou nutné k navázání spojení a přihlášení k serveru, k nastavení přenosu souborů, k vlastnímu přenosu souborů, k ukončení spojení se serverem a k ukončení relace klienta. Tato ukázka uvádí podpříkazy, použité k přenosu souborů na dva různé vzdálené systémy.

```
Columns . . . : 1 71          Browse          GERRYLIB/QCLSRC
SEU==>          FTPCMDS
FMT **  ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
***** Beginning of data *****
0001.00 gwil ****
0002.00 close
0003.00 open sysnamrs
0004.00 user root root
0005.00 ascii
0006.00 syscmd dltf 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((gwi1 sysnam14))
0018.00 close
0019.00 quit
***** End of data *****
F3=Exit F5=Refresh F9=Retrieve F10=Cursor F12=Cancel
F16=Repeat find F24=More keys
```

**Obrázek 2.** Přenos souborů na dva vzdálené systémy.

Zde je uveden popis podpříkazů FTP klienta uvedených na obrázku 2. Čísla řádků na obrázku odpovídají číslům v následujícím textu.

**0001**

Uživatelské ID a heslo pro fiktivní připojení v rámci klienta systému iSeries SYSNAM03.

**0002**

Uzavření fiktivního připojení v systému iSeries SYSNAM03.

**0003**

Otevření řídicího připojení k systému RISC System/6000<sup>(R)</sup> SYSNAMRS.

**0004**

Podpříkaz USER s uživatelským ID a heslem pro systém SYSNAMRS.

**Poznámka:**

Když provádíte zpracování FTP v dávkovém režimu, musí za podpříkazem OPEN následovat podpříkaz USER. U podpříkazu USER musíte zadat oba parametry - uživatelské ID i heslo. To se liší od online režimu zpracování FTP. Když provádíte zpracování FTP interaktivně online, klient automaticky volá podpříkaz USER a požádá vás o přihlašovací ID. Při zpracování FTP v dávkovém režimu neexistuje automatické volání podpříkazu USER.

**0005**

Přenos ASCII dat (budou na serveru iSeries konvertována z/na EBCDIC).

**0006**

CL příkaz, který se má spustit na klientu iSeries: výmaz souboru. Namísto tohoto příkazu můžete v dalším příkazu použít parametr (REPLACE).

**0007**

Načtení souboru ze systému RISC System/6000.

**0008**

Uzavření řídicího připojení k systému RISC System/6000 SYSNAMRS.

**0009**

Otevření připojení k systému VAX MVAX.

**0010**

Podpříkaz USER s uživatelským ID a heslem pro MVAX.

**0011**

Načtení souboru z VAX, který nahradí stávající soubor iSeries.

**0012**

Uzavření řídicího připojení k VAX MVAX.

**0013**

Otevření řídicího připojení ke vzdálenému systému iSeries SYSNAMRS.

**0014**

Podpříkaz USER s uživatelským ID a heslem pro SYSNAM02.

**0015**

Přenos EBCDIC dat (jde o přenos ze serveru iSeries na iSeries).

**0016**

Odeslání souboru iSeries do systému iSeries SYSNAM02 prostřednictvím TCP/IP.

**0017**

Odeslání tohoto souboru ze systému iSeries SYSNAM03 do vzdáleného systému iSeries SYSNAM14 přes síť TCP/IP.

**0018**

Uzavření řídicího připojení k systému iSeries SYSNAM02.

**0019**

Ukončení aplikace FTP.

### Vytvoření CL programu pro spuštění úlohy FTPBATCH

K naplánování přenosů souborů a jejich provedení bez obsluhy vytvořte CL program, který spustí úlohu FTPBATCH. V následujícím příkladu se předpokládá, že přenosy souborů se uskuteční příští pátek v 17:00 hodin, v neobsluhovaném režimu.

```

Columns . . . : 1 71          Browse          GERRYLIB/QCLSRC
SEU==>          FTPSUBMIT
FMT **  ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
***** Beginning of data *****
0001.00 PGM
0002.00          SBMJOB      CMD(CALL PGM(GERRYLIB/FTPbatch)) +
0003.00          JOB(FTPFRIDAY) OUTQ(QUSRSYS/GERRYQ)      +
0004.00          SCDDATE(*FRI) SCDDTIME(170000) /* FTP for +
0005.00          Friday, 5:00 in the afternoon */
0006.00 ENDPGM
***** End of data *****

F3=Exit  F5=Refresh  F9=Retrieve  F10=Cursor  F12=Cancel
F16=Repeat find  F24=More keys

(C) COPYRIGHT IBM CORP. 1981, 1994.

```

**Obrázek 3.** CL program pro spuštění úlohy dávkového FTP.

### Kontrola chyb ve výstupním souboru FTP

Během zpracování v naplánovaném čase vytváří FTP data ve výše uvedeném členu souboru FTPLOG. Data ve členu souboru FTPLOG odpovídají původním příkazům, které vidíte v obou příkladech.

```

Connecting to host name LOOPBACK at address 127.0.0.1 using port 21.
220-QTCP at localhost.
220 Connection will close if idle more than 5 minutes.
Enter login ID (gwil):

```

```
>>>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.
Enter an FTP subcommand.

> CLOSE
221 QUIT subcommand received.
Enter an FTP subcommand.

> OPEN SYSNAMRS
Connecting to host name SYSNAMRS at address 9.4.73.198 using port 21.
220 sysnamrs.sysnam123.ibm.com FTP server (Version 4.9 Thu Sep 2 20:35:07 CDT
    1993) ready.
Enter an FTP subcommand.
```

**Obrázek 4.** Výstup FTP (FTPLLOG) po zpracování programu FTPBATCH (Část 1)

```
> USER root ****
331 Password required for root.
230 User root logged in.
UNIX Type: L8 Version: BSD-44
Enter an FTP subcommand.

> ASCII
200 Type set to A; form set to N.
Enter an FTP subcommand.

> SYSCMD DLTF FILE(GERRYLIB/RS6)
Enter an FTP subcommand.

> 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.
Enter an FTP subcommand.
```

**Obrázek 5.** Výstup FTP (FTPLLOG) po zpracování programu FTPBATCH (Část 2)

```
> CLOSE
221 Goodbye.
Enter an FTP subcommand.

> OPEN MVAX
Connecting to host system mvax at address 9.4.6.252 using port 21.
220 FTP Service Ready
Enter an FTP subcommand.

> USER TESTER *****
331 User name TESTER received, please send password
230 TESTER logged in, directory $DISK1:[TESTER]
Enter an FTP subcommand.

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.
Enter an FTP subcommand.

> CLOSE
221 Goodbye.
Enter an FTP subcommand.

OPEN SYSNAM02
Connecting to host system SYSNAM02 at address 9.4.73.250 using port 21.
220-QTCP at SYSNAM02.sysnam123.ibm.com.
220 Connection will close if idle more than 5 minutes.
Enter an FTP subcommand.
```

**Obrázek 6.** Výstup FTP (FTPLOG) po zpracování programu FTPBATC (Část 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.
Enter an FTP subcommand.

> EBCDIC
200 Representation type is EBCDIC nonprint.
Enter an FTP subcommand.

> 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.
467 bytes transferred in 0.148 seconds. Transfer rate 3.146 KB/sec.
Enter an FTP subcommand.

> RCMD SNDNETF FILE(GERRYLIB/RS6) TOUSRID((GERRYLIB SYSNAM14))
250 Command SNDNETF FILE(GERRYLIB/RS6) TOUSRID((GWIL SYSNAM14))
successful.
Enter an FTP subcommand.
```

**Obrázek 7.** Výstup FTP (FTPLOG) po zpracování programu FTPBATC (Část 4)

```
> CLOSE
221 QUIT subcommand received.
Enter an FTP subcommand.
> QUIT
(This ends the FTP application)
```

**Obrázek 8.** Výstup FTP (FTPLOG) po zpracování programu FTPBATC (Část 5)

Můžete zkontrolovat chyby v tomto výstupu, které se mohly vyskytnout během zpracování FTP. Můžete provést vizuální kontrolu, nebo spustit program, který testuje kódy chybových odpovědí. Třímístné chybové kódy odpovědí FTP začínají číslem 4 nebo 5. Při kontrole chyb se nevěšíte zpráv typu '467 bytes transferred...' (467 bajtů přeneseno...)

*Vzorová procedura:* Jako součást produktu TCP/IP se dodává vzorová procedura REXX a vzorový člen fyzického souboru. Soubor QATMPINC v knihovně QTCP zahrnuje tyto dva členy:

- BATCHFTP obsahuje zdrojový kód REXX, který určuje vstupní a výstupní dávkové soubory a spouští FTP.
- BFTPFILE obsahuje podpříkazy a data, která jsou povinná pro přihlášení a zpracování FTP.

---

## Referenční informace o FTP



Následující témata podávají informace, které se vám mohou hodit při práci s FTP serverem a klientem:

### Podpříkazy FTP serveru.

Tyto příkazy slouží ke komunikaci mezi klientem a serverem. Toto téma obsahuje popisy podpříkazů ekvivalentních s příkazy jazyka CL iSeries<sup>(TM)</sup>, které jsou jedinečné pro FTP server iSeries (See 39)

### Podpříkazy FTP klienta

Podpříkazy FTP klienta slouží k vytvoření spojení se vzdáleným FTP serverem, k procházení knihovny a adresáři, k vytváření, mazání a přenosu souborů.

### Programy výstupních bodů FTP

Programy výstupních bodů FTP slouží k zabezpečení přístupu na FTP server. FTP server komunikuje s každým programem výstupního bodu prostřednictvím určitého výstupního bodu. Toto téma obsahuje popisy parametrů a ukázky kódu.

### Další referenční informace

- Metody přenosu dat.
- Systémy souborů a konvence pojmenování.
- Stavové zprávy FTP serveru.
- Konvence syntaxe FTP serveru.
- Konvence syntaxe FTP klienta.



## Podpříkazy FTP serveru

Toto téma obsahuje přehled podpříkazů FTP serveru. Pomocí podpříkazů serveru komunikuje FTP klient se serverem. Protože uživatel běžně s FTP serverem nekomunikuje, poskytuje toto téma pro váš přehled informace o podpříkazech serveru, o jejich funkci, jejich syntaxi a o stavových zprávách v odpovědích FTP.

FTP server iSeries<sup>(TM)</sup> podporuje tyto podpříkazy:

Podpříkaz	Co dělá
ABOR	Zruší předchozí podpříkaz
ADDM	Přidá člen do fyzického souboru
ADDV	Přidá člen s proměnlivou délkou do fyzického souboru
APPE	Přidá data do zadaného souboru
AUTH	Definuje metodu autentizace použitou pro aktuální relaci FTP.
CDUP	Změní adresář na nadřazený adresář

<b>Podpříkaz</b>	<b>Co dělá</b>
CRTL	Vytvoří knihovnu
CRTP	Vytvoří fyzický soubor
CRTS	Vytvoří zdrojový fyzický soubor
CWD	Změní pracovní adresář nebo knihovnu
DEBUG	Spustí nebo zastaví trasování serveru
DELE	Vymaže soubor, člen nebo dokument
DLTF	Vymaže soubor
DLTL	Vymaže knihovnu
HELP	Získá informace o podpříkazech FTP serveru
LIST	Vypíše seznam souborů nebo položek adresáře
MKD	Vytvoří adresář
MODE	Udává formát pro přenos dat
NLST	Vypíše seznam jmen souborů nebo adresářů
NOOP	Zkontroluje, zda server odpovídá
PASS	Odešle heslo na server
PASV	Přikáže serveru, aby pasivně otevřel další datové připojení
PBSZ	Definuje maximální velikost vyrovnávací paměti pro ochranu vyrovnávací paměti, která se použije pro data kódovaná na úrovni aplikace, která se odesílají a přijímají prostřednictvím datového připojení.
PORT	Určuje datový port, na kterém bude klient naslouchat a čekat na připojení
PROT	Definuje ochranu, která se použije pro datová připojení FTP
PWD	Zobrazí aktuální pracovní adresář
QUIT	Odhlásí uživatele; uzavře spojení
RCMD	Odešle CL příkaz na FTP server
REIN	Znovu spustí relaci na serveru
RETR	Načte data ze serveru
RMD	Odstraní adresář
RNFR	Udává soubor, který se má přejmenovat
RNTO	Udává jméno nového souboru
SITE	Odešle informace pro použití serverem
STAT	Získá informace o stavu ze serveru
STOR	Uloží data na server a nahradí stávající soubor
STOU	Uloží data na server, ale nenahradí stávající soubor
STRU	Udává členění souboru
SYST	Vytiskne jméno operačního systému na serveru
TIME	Nastaví hodnotu časové prodlevy pro FTP server
TYPE	Udává typ přenosu souboru
USER	Odešle přihlašovací ID uživatele na server
XCUP	Přejde do nadřazeného adresáře

Podpříkaz	Co dělá
XCWD	Přejde do pracovního adresáře
XMKD	Vytvoří adresář
XPWD	Zobrazí aktuální adresář nebo knihovnu
XRMD	Odstraní adresář

### Podpříkazy, které slouží výhradně pro FTP server iSeries

K podpříkazům FTP serveru iSeries patří speciální sada příkazů, které jsou skutečnými zkrácenými jmény rovnocenných, ale delších CL příkazů iSeries. Jména těchto speciálních podpříkazů serveru se musí skládat ze čtyř znaků, aby se vyhovělo omezením architektury FTP. Když server iSeries obdrží tyto podpříkazy, interpretuje je následujícím způsobem:

- ADDM = ADDPFM (Přidání členu fyzického souboru).
- ADDV = ADDPVLN (Přidání členu fyzického souboru s proměnnou délkou).
- CRTL = CRTLIB (Vytvoření knihovny).
- CRTP = CRTPF (Vytvoření fyzického souboru).
- CRTS = CRTSRCPF (Vytvoření zdrojového fyzického souboru).
- DLTF = DLTF (Vymazání souboru).
- DLTL = DLTLIB (Vymazání knihovny).

Kromě těchto speciálních podpříkazů můžete použít podpříkaz FTP serveru RCMD k odeslání jakéhokoliv CL příkazu na server.

### Související témata:

- Konvence syntaxe FTP serveru.
- Stavové zprávy FTP serveru.
- Podpříkazy FTP klienta: Tyto podpříkazy slouží k vytvoření spojení se vzdáleným FTP serverem, k procházení knihovnami a adresáři a k vytváření, mazání a přenosu souborů.

## ADDM (Přidat člen fyzického souboru)

### Podpříkaz FTP serveru

ADDM parametry

#### parametry

Parametry tohoto podpříkazu jsou shodné s parametry CL příkazu ADDPFM.

Chcete-li například přidat člen BANANA do fyzického souboru GEORGE v knihovně RLKAYS na serveru iSeries<sup>™</sup>, zadejte:

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

### Jděte na:

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## ADDV (Přidat člen fyzického souboru s proměnnou délkou)

### Podpříkaz FTP serveru

ADDV parametry

## parametry

Parametry tohoto podpříkazu jsou shodné s parametry CL příkazu ADDPVLM.

Chcete-li například přidat člen POLEBEAN do fyzického souboru GEORGE v knihovně RLKAYS na serveru iSeries<sup>™</sup>, zadejte:

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

### Jděte na:

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## APPE (Přidat k existujícímu souboru)

### Podpříkaz FTP serveru

Podpříkaz FTP serveru APPE přijme přenesená data a uloží je do souboru v systému serveru. Pokud zadaný soubor již existuje, APPE přidá data do tohoto souboru; jinak zadaný soubor vytvoří.

```
APPE jménosouboru
```

### jménosouboru

Soubor na serveru, do něhož se přidají vaše data.

### Jděte na:

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## AUTH (Oprávnění)

### Podpříkaz FTP serveru

Podpříkaz FTP serveru AUTH definuje způsob autentizace/zabezpečení, který se použije pro aktuální relaci FTP. Syntaxe tohoto podpříkazu je:

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

Hodnoty parametru:

TLS-C	K zabezpečení ochrany dat se použije protokol TLS (Transport Layer Security). Nastavení zabezpečení ochrany dat pro datové připojení používá předvolby RFC2228, tzn. že není použita žádná implicitní ochrana datového připojení.
TLS-P	K zabezpečení ochrany dat se použije protokol TLS. Současně se nastaví implicitní ochrana datového připojení (což odpovídá posloupnosti příkazů AUTH TLC-C, PBSZ 0, PROT P).
TLS	Synonymum pro TLS-C.
SSL	Synonymum pro TLS-P.

**Poznámka:** Protokol TLS je kompatibilní s protokolem SSL (Secure Sockets Layer).

### Jděte na:

- Podpříkazy FTP serveru.



- Konvence syntaxe FTP serveru.

## **CRTL (Vytvořit knihovnu)**

### **Podpříkaz FTP serveru**

CRTL parametry

#### **parametry**

Parametry tohoto podpříkazu jsou shodné s parametry CL příkazu CRTLIB.

Chcete-li například vytvořit knihovnu jménem TESTTCP na serveru iSeries<sup>(TM)</sup>, zadejte:

```
CRTL TESTTCP
```

#### **Jděte na:**

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## **CRTP (Vytvořit fyzický soubor)**

### **Podpříkaz FTP serveru**

CRTP parametry

#### **parametry**

Parametry tohoto podpříkazu jsou shodné s parametry CL příkazu CRTPF.

Chcete-li například vytvořit fyzický soubor jménem MYFILE s délkou záznamu 80 a žádným omezením počtu členů, zadejte:

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

#### **Jděte na:**

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## **CRTS (Vytvořit zdrojový fyzický soubor)**

### **Podpříkaz FTP serveru**

CRTS parametry

#### **parametry**

Parametry tohoto podpříkazu jsou shodné s parametry CL příkazu CRTSRCPF.

Chcete-li například vytvořit zdrojový fyzický soubor jménem GEORGE v knihovně RLKAYS, zadejte:

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

#### **Jděte na:**

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## **CWD (Změnit pracovní adresář nebo knihovnu)**

### **Podpříkaz FTP serveru**

Podpříkaz FTP serveru CWD slouží ke změně pracovního adresáře, knihovny nebo skupiny souborů.

CWD adresář

**Jděte na:**

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## **DEBUG (Zapnout trasování FTP serveru)**

**Poznámka:**

Trasování FTP serveru používejte pouze k nahlašování problémů softwaru IBM. Tato funkce může negativně ovlivnit výkon systému.

### **Podpříkaz FTP serveru**

DEBUG

Není-li trasování FTP serveru aktivní, server je spustí. Server pokračuje v trasování tak dlouho, dokud neobdrží další podpříkaz DEBUG nebo podpříkaz QUIT. Když server ukončí trasování, může nastat delší časová prodleva, než naformátuje data z trasování.

**Jděte na:**

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## **DELE (Vymazat soubor nebo dokument)**

### **Podpříkaz FTP serveru**

Podpříkaz FTP serveru DELE slouží k vymazání souboru, členu nebo dokumentu.

DELE vzdálenýsoubor

**Jděte na:**

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## **DLTF (Vymazat soubor)**

### **Podpříkaz FTP serveru**

DLTF parametry

#### **parametry**

Parametry tohoto podpříkazu jsou shodné s parametry CL příkazu DLTF.

Chcete-li například vymazat soubor MYFILE v knihovně RLKAYS, zadejte:

DLTF FILE(RLKAYS/MYFILE)

**Jděte na:**

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## DLTL (Vymazat knihovnu)

### Podpříkaz FTP serveru

DLTL parametry

#### parametry

Parametry tohoto podpříkazu jsou shodné s parametry CL příkazu DLTLIB.

Chcete-li například vymazat některou knihovnu, zadejte:

DLTL jménoknihovny

#### Jděte na:

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## HELP (Získání nápovědy ze vzdáleného serveru iSeries<sup>™</sup>)

Chcete-li získat informace o podpříkazech FTP serveru, použijte podpříkaz HELP v tomto formátu:

HELP [podpříkaz]

#### podpříkaz

Jméno podpříkazu serveru, o němž chcete získat bližší informace. Například HELP ADDM vám poskytne informace o tom, jak přidat člen do fyzického souboru na serveru iSeries.

Chcete-li zjistit syntaxi podpříkazu ADDV, který používá server iSeries, použijte podpříkaz serveru:

HELP ADDV

#### Jděte na:

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## LIST (Seznam souborů)

### Podpříkaz FTP serveru

Chcete-li získat seznam položek adresáře, obsah knihovny nebo soubory ze skupiny souborů, použijte podpříkaz FTP serveru LIST:

LIST [adresář | jméno]

Vypíše seznam pouze těch souborů, které může FTP přenášet.

#### Jděte na:

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.
- SITE (Odeslat informace používané systémem serveru): Tento podpříkaz slouží ke změně nastavení informací, které vrací podpříkaz LIST.
- Informace podpříkazu LIST ve formátu UNIX<sup>®</sup>: Tento podpříkaz slouží k zobrazení informací, které vrací podpříkaz LIST.
- Informace podpříkazu LIST ve formátu iSeries<sup>™</sup>: Tento podpříkaz slouží k zobrazení informací, které vrací podpříkaz LIST.

## MKD (Vytvořit adresář)

### Podpříkaz FTP serveru

Chcete-li vytvořit adresář, použijte podpříkaz FTP serveru MKD.

```
MKD jménoadresáře
```

#### Jděte na:

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## MODE (Nastavit režim přenosu)

### Podpříkaz FTP serveru

Chcete-li určit, jak se mají přenášet bity dat, zadejte režim nebo formát dat pomocí podpříkazu FTP serveru MODE:

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

#### **B**

Určuje blokový režim. V tomto režimu jsou data přenášena jako série datových bloků, začínajících jedním nebo více bajty hlavičky.

#### **S**

Určuje proudový režim. V tomto režimu se data přenášejí jako proud bajtů. V proudovém režimu můžete použít libovolnou reprezentaci znaků. Tento režim přenosu je efektivnější, protože server nepřenáší žádné informace o datovém bloku.

#### Poznámky:

1. Proudový režim je předvolený režim přenosu, který používá server iSeries<sup>(TM)</sup>, a je současně režimem upřednostňovaným.
2. Pokud ne zadáte žádný parametr, server vrátí odpověď, která udává aktuální nastavení režimu přenosu.

#### Jděte na:

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## NLST (Seznam jmen)

### Podpříkaz FTP serveru

Chcete-li získat výpis více souborů, skupiny souborů, obsahu adresáře nebo knihovny, který bude obsahovat pouze jména souborů, použijte podpříkaz FTP serveru NLST:

```
NLST [adresář | jméno]
```

Vypíše seznam pouze těch souborů, které může FTP přenášet.

#### Jděte na:

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## NOOP (Získat odezvu serveru)

### Podpříkaz FTP serveru

Podpříkaz FTP serveru NOOP pošle klientovi odpověď "OK". Tento příkaz žádným způsobem neovlivní vlastní zpracování na serveru. Klient používá tento podpříkaz, aby zjistil, zda server je připojen a odpovídá. Podpříkaz NOOP zadejte ve tvaru:

```
NOOP
```

#### Jděte na:

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## PASS (Heslo)

### Podpříkaz FTP serveru

```
PASS heslo
```

#### **heslo**

Řetězec, který představuje vaše heslo do systému serveru.

**Poznámka:** Podpříkazu serveru PASS musí bezprostředně předcházet podpříkaz serveru USER.

#### Jděte na:

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## PASV (Použit pasivní datové připojení)

### Podpříkaz FTP serveru

Chcete-li dát serveru pokyn k pasivnímu otevření dalšího datového připojení, použijte podpříkaz FTP serveru PASV v tomto formátu:

```
PASV
```

#### Jděte na:

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## PBSZ (Velikost ochranné vyrovnávací paměti)

### Podpříkaz FTP serveru

Podpříkaz PBSZ definuje maximální velikost vyrovnávací paměti, která se má použít k odeslání nebo příjmu dat kódovaných na aplikační úrovni při datovém připojení. Syntaxe tohoto podpříkazu je:

PBSZ *hodnota*

kde *hodnota* je řetězec ASCII znaků vyjádřený dekadickým celým číslem.

#### **Poznámka:**

RFC2228 vyžaduje zadání podpříkazu PBSZ před podpříkazem PROT. TLS/SSL však pracuje s bloky dat, takže jediná možná hodnota je '0'.

**Jděte na:**

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

**PORT (Datový port)****Podpříkaz FTP serveru**

Chcete-li označit port, na kterém bude klient naslouchat datovému připojení, použijte podpříkaz FTP serveru PORT v tomto formátu:

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

**hn**

Představuje IP adresu systému. Je to znakový řetězec vyjádřený dekadickou hodnotou v rozmezí 0 až 255.

**pn**

Představuje číslo TCP portu. Je to znakový řetězec vyjádřený dekadickou hodnotou v rozmezí 0 až 255.

Ke konverzi hodnot p1 a p2 na číslo TCP portu použijte vzorec:

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

Například tento podpříkaz PORT:

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

udává číslo portu 1032 a IP adresu 9.180.128.180.

**Poznámka:**

Když server uzavře připojení, nemůže se připojit ke stejné IP adrese a číslu portu klienta dříve, než uplyne dvouminutová časová prodleva, jak je uvedeno v pravidlech RFC 1122 pro TCP/IP. Toto omezení neplatí pro navázání spojení serveru se stejnou IP adresou klienta a rozdílným číslem portu.

**Jděte na:**

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

**PROT (Úroveň ochrany datového kanálu)****Podpříkaz FTP serveru**

Podpříkaz PROT definuje ochranu, která se použije pro datová připojení FTP (datová připojení slouží k přenosu výpisů adresářů a dat souborů). Syntaxe tohoto podpříkazu je:

PROT [ C | P ]

Hodnoty parametru:

C	Čitelný (Clear). Datové připojení přenáší "prvotní data" souborů bez použití jakéhokoli zabezpečení.
P	Soukromý (Private). Datové připojení použije TLS/SSL, umožňující ochranu integrity a důvěrnosti.

**Jděte na:**

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

**PWD (Zobrazit pracovní adresář nebo knihovnu)****Podpříkaz FTP serveru**

Když zadáte podpříkaz FTP serveru PWD, server vrátí klientu odpověď se jménem aktuálního adresáře nebo knihovny.

```
PWD
```

**Jděte na:**

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

**QUIT (Ukončit relaci FTP serveru)****Podpříkaz FTP serveru**

Podpříkaz FTP serveru QUIT odhlásí uživatele klienta a uzavře řídicí připojení. Probíhá-li právě přenos souboru, připojení zůstane otevřeno, dokud se přenos souboru nedokončí, a teprve potom je server uzavře.

```
QUIT
```

**Jděte na:**

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

**RCMD (Odeslat CL příkaz do systému FTP serveru)****Podpříkaz FTP serveru**

Podpříkaz serveru RCMD použijte, chcete-li zpracovávat příkazy jazyka CL iSeries<sup>™</sup> v systému FTP serveru. Délka řetězce v podpříkazu RCMD může být až 1000 znaků. Protože při použití podpříkazu RCMD nejsou k dispozici náznaky, řetězec podpříkazu RCMD musí zahrnovat všechny parametry nutné ke zpracování CL příkazu.

Proběhne-li CL příkaz zadaný prostřednictvím podpříkazu RCMD úspěšně, zobrazí se zpráva, že podpříkaz byl úspěšný. Objeví-li se chyby, zobrazí se zpráva, že došlo k chybě. Zpráva neobsahuje informaci, o jakou chybu se jedná, pokud se nejedná o chybu v důsledku neplatnosti jména knihovny, souboru nebo členu.

Toto je příklad použití RCMD ke zpracování příkazu DLTF (Vymazání souboru):

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

mylib je jméno knihovny, ze které se má vymazat soubor. myfile je jméno souboru, který se má vymazat.

**Jděte na:**

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

Také si můžete přečíst informace o REXEC serveru, který nabízí alternativní způsob provádění CL příkazů ve vzdáleném systému.

## REIN (Znovu inicializovat relaci mezi systémy)

### Podpříkaz FTP serveru

```
REIN
```

Podpříkaz REINITIALIZE:

1. Umožní dokončit právě probíhající přenos.
2. Ukončí relaci uživatele a odstraní všechny informace o vstupech/výstupech a účtech.
3. Nastaví všechny parametry serveru na předvolené hodnoty.
4. Ponechá otevřené řídicí připojení.

#### Jděte na:

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## RETR (Načíst soubor)

### Podpříkaz FTP serveru

Chcete-li načíst data ze systému serveru, použijte podpříkaz FTP serveru RETR.

```
RETR vzdálenýsoubor
```

#### Jděte na:

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## RMD (Odstranit adresář)

### Podpříkaz FTP serveru

Chcete-li odstranit adresář, použijte podpříkaz FTP serveru RMD.

```
RMD jménoadresáře
```

#### Jděte na:

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## RNFR (Přejmenovat z)

### Podpříkaz FTP serveru

Podpříkaz FTP serveru RNFR slouží k přejmenování souboru. Musí být bezprostředně následován podpříkazem serveru RNTO (Přejmenovat na).

```
RNFR jménosouboru
```

#### **jménosouboru**

Jméno souboru, který chcete přejmenovat.

**Poznámka:** Na serveru iSeries<sup>(TM)</sup> nemůžete přejmenovat soubor na odlišný systém souborů.



**Jděte na:**

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

**RNTO (Přejmenovat na)  
Podpříkaz FTP serveru**

Podpříkaz FTP serveru RNTO udává nové jméno souboru při přejmenování souborů v systému serveru. Tento podpříkaz musí bezprostředně následovat po podpříkazu RNFR, který udává původní jméno souboru.

RNTO jménosouboru

**jménosouboru**

Jméno souboru, na které chcete soubor přejmenovat.

**Poznámka:** Na serveru iSeries<sup>(TM)</sup> nemůžete přejmenovat soubor na odlišný systém souborů.

**Jděte na:**

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

**SITE (Odeslat informace používané systémem serveru)  
Podpříkaz FTP serveru**

Chcete-li odeslat informace, které použije systém serveru k poskytování služeb pro něj specifických, použijte podpříkaz FTP serveru SITE v tomto formátu:

SITE [*parametry*]

FTP server iSeries<sup>(TM)</sup> podporuje tyto parametry podpříkazu SITE:

**LISTFMT 0**

Server vrátí informace pro podpříkaz LIST ve formátu iSeries (viz Informace podpříkazu LIST ve formátu iSeries), který byl uveden ve verzi V3R1M0. Klient iSeries podporuje jak formát iSeries, tak formát UNIX.

**LISTFMT 1**

Server vrátí informace pro podpříkaz LIST ve formátu UNIX (viz Informace podpříkazu LIST ve formátu UNIX). Jméno souboru je uvedeno jako poslední položka každého vráceného řádku. Klient iSeries podporuje jak formát iSeries, tak formát UNIX.

**LISTFMT**

Vrátí zprávu, která udává aktuální nastavení parametru LISTFMT na FTP serveru.

**Poznámky:**

Chcete-li změnit předvolené nastavení parametru LISTFMT na serveru, použijte volbu LISTFMT v příkazu CHGFTP. K nastavení této vlastnosti FTP serveru můžete také použít iSeries Navigator:

1. V produktu iSeries Navigator rozbalte **váš server iSeries** → **Síť** → **Servery** → **TCP/IP**.
2. V pravém okně klepněte pravým tlačítkem myši na **FTP** a vyberte **Vlastnosti**.
3. Klepněte na ouško **Výchozí formáty**.
4. Pod hlavičkou **Výpis souborů** nastavte formát iSeries, nebo UNIX jako předvolbu pro LISTFMT na serveru.
5. Klepnutím na **OK** potvrďte změny.

**NAMEFMT 0**

Použijte se formát jména KNIHOVNA/SOUBOR.ČLEN. Tento formát jména je určen pouze pro databázové soubory systému souborů knihoven.

**NAMEFMT 1**

Použijte se formát jména cesty. Tento formát jména je určen pro všechny systémy souborů, které jsou podporovány FTP, včetně systému souborů knihoven. Formát jména 1 se musí používat pro práci se všemi systémy souborů iSeries jinými než systémy souborů knihoven.

**NAMEFMT**

Zobrazí zprávu, která obsahuje aktuální formát jmen souborů na serveru.

**Poznámka:**

Chcete-li změnit předvolené nastavení parametru NAMEFMT na FTP serveru iSeries, použijte volbu NAMEFMT v příkazu CHGFTP.

**CRTCCSID \*CALC**

Nové databázové soubory, které byly vytvořeny během přenosů souborů ASCII, budou používat předvolený CCSID EBCDIC, který odpovídá CCSID použitému při přenosu souborů ASCII.

**CRTCCSID \*USER**

Nové databázové soubory, které byly vytvořeny během přenosů souborů ASCII, budou používat CCSID aktuální úlohy. Je-li však tento CCSID roven 65535, potom předvolený CCSID se určí podle ID jazyka aktuální úlohy.

**CRTCCSID \*SYSVAL**

Nové databázové soubory, které byly vytvořeny během přenosů souborů ASCII, budou používat CCSID určený systémovou hodnotou QCCSID.

**CRTCCSID [CCSID-číslo]**

Zadejte, jaký CCSID chcete používat při vytváření databázových souborů na klientu během přenosů souborů ASCII. Server ověří tuto hodnotu.

**CRTCCSID**

Zobrazí zprávu, která obsahuje aktuální nastavení CRTCCSID FTP klienta.

**NULLFLDS 0**

Server nepovolí přenos databázových souborů, které obsahují pole NULL. Je to předvolená hodnota.

**NULLFLDS 1**

Server povolí přenos databázových souborů, které obsahují pole NULL.

**Poznámka:**

Přenos souborů, které obsahují pole NULL, vyžaduje, aby toto nastavení bylo povoleno jak u klienta, tak u serveru. Když přenášíte soubor, který obsahuje pole NULL, na jiný systém než iSeries, nebo když typ přenosu dat vyžaduje konverzi kódové stránky dat, může dojít k nepředvídatelným výsledkům.

**NULLFLDS**

Vrátí zprávu, která obsahuje aktuální nastavení parametru NULLFDLS na FTP serveru.

**TRIM 0**

Nastaví volbu Trim na OFF. Server bude odesílat koncové mezery databázových záznamů.

**TRIM 1**

Nastaví volbu Trim na ON. Server nebude odesílat koncové mezery databázových záznamů při přenosu databázových souborů, které používají souborové členění a proudový režim. Je to předvolená hodnota.

**TRIM 2**

Server nebude odesílat koncové mezery databázových záznamů při žádných přenosech, tzn. ani při záznamovém členění a blokovém režimu.

**TRIM**

Zobrazí zprávu, která obsahuje aktuální nastavení parametru TRIM na FTP serveru.

**Poznámky:**

1. Před vznikem tohoto podpříkazu byly koncové mezery záznamů v systému souborů QSYS.LIB vždy před přenosem souboru do systému serveru odstraněny.
2. Na nastavení parametru TRIM se nebere ohled při binárním přenosu souborů (TYPE I). Při přenosu souborů typu TYPE I se mezery nikdy neořezávají, bez ohledu na nastavení parametru TRIM.

**Jděte na:**

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

**STOR (Uložit soubor)****Podpříkaz FTP serveru**

Chcete-li uložit data do systému serveru s tím, že se stávající soubor přepíše, použijte podpříkaz FTP serveru STOR v následujícím formátu:

STOR vzdálenýsoubor
---------------------

**Jděte na:**

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

**STOU (Uložit jedinečný soubor)****Podpříkaz FTP serveru**

Chcete-li uložit data do systému serveru a nepřepsat stávající soubor, použijte podpříkaz FTP serveru STOU:

```
STOU vzdálenýsoubor
```

Server vygeneruje jedinečné jméno souboru. Jméno přidělené souboru je uvedeno v odpovědi, která se vrátí klientu.

**Jděte na:**

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## **STRU (Zadat členění souboru)**

### **Podpříkaz FTP serveru**

Chcete-li zadat členění souboru jako souvislou posloupnost datových bajtů, použijte podpříkaz FTP serveru STRU v tomto formátu:

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

**F**

Souborové členění. Soubor bude členěn jako souvislá řada datových bajtů.

**R**

Záznamové členění. Soubor bude členěn jako řada po sobě jdoucích záznamů.

**Poznámky:**

1. Členění souboru ovlivňuje režim přenosu i interpretaci a uložení souboru.
2. Pokud nezádáte žádné parametry, server vrátí odpověď, která bude obsahovat aktuální nastavení členění souboru.

**Jděte na:**

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## **SYST (Určit jméno operačního systému)**

### **Podpříkaz FTP serveru**

Chcete-li získat jméno operačního systému na serveru, použijte podpříkaz FTP serveru SYST:

```
SYST
```

Vrácená informace je závislá na systému.

Server iSeries<sup>(TM)</sup> zahrnuje verzi TCP/IP.

Toto je příklad odpovědi serveru:

OS/400 is the remote operating system. The TCP/IP version is "V4R4M0". (OS/400 je vzdálený operační systém. Verze TCP/IP je "V4R4M0".)

**Jděte na:**

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## TIME (Nastavit hodnoty časové prodlevy pro FTP server)

### Podpříkaz FTP serveru

Po navázání řídicího připojení FTP mezi FTP klientem a FTP serverem se časová prodleva pro toto připojení řídí podle FTP serveru. V tomto případě se jedná o hodnotu časové prodlevy při nečinnosti.

Existuje také hodnota časové prodlevy pro datové připojení, známá jako časová prodleva při přenosu.

Formát podpříkazu FTP serveru TIME je:

```
TIME nečinnost [přenos]
```

#### nečinnost

Počet sekund, po které server čeká, než uzavře spojení s klientem. Hodnoty časové prodlevy při nečinnosti mohou být v rozsahu 1 až 9 999 999 sekund. Předvolená hodnota časové prodlevy při nečinnosti je 300 sekund.

#### přenos

Časová prodleva při přenosu souboru v sekundách. Tento parametr je volitelný. Pokud jej ne zadáte, zachová server aktuální hodnotu. Hodnoty časové prodlevy mohou být v rozsahu 1 až 9 999 999 sekund. Předvolená hodnota časové prodlevy při přenosu je 420 sekund.

Chcete-li například nastavit hodnotu časové prodlevy při nečinnosti FTP serveru na 1000 sekund a uchovat stávající hodnotu časové prodlevy při přenosu, zadejte:

```
QUOTE TIME 1000
```

Podpříkaz TIME není standardním podpříkazem FTP. Je to podpříkaz specifický pro FTP server iSeries<sup>(TM)</sup>.

#### Jděte na:

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## TYPE (Zadat typ reprezentace dat)

### Podpříkaz FTP serveru

Chcete-li zadat typ přenosu nebo reprezentaci dat pro přenos souborů, použijte podpříkaz FTP serveru TYPE v tomto formátu:

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

#### A

Nastaví typ přenosu jako předvolený (ASCII) typ přenosu. Server k souboru nepřirazuje řízení vertikálního formátu. Pro ASCII server podporuje pouze předvolený formát NON PRINT. Typ přenosu ASCII je určen pro přenos textových souborů, kromě případů, kdy oba systémy používají reprezentaci dat EBCDIC.

**Poznámka:**

Hodnota CCSID pro TYPE A je shodná s hodnotou CCSID atributů konfigurace FTP serveru. Tyto atributy můžete změnit příkazem CHGFTP.

**B**

Shift JIS Kanji (CCSID 932)

**B 1**

Shift JIS Kanji (CCSID 932)

**B 2**

Kanji - rozšířený UNIX kód (CCSID 5050)

**B 3**

JIS 1983 používající ASCII shift-in escape sekvenci (CCSID 5054)

**B 3 A**

JIS 1983 používající ASCII shift-in escape sekvenci (CCSID 5054)

**B 3 R**

JIS 1983 používající JISROMAN shift-in escape sekvenci (CCSID 5052)

**B 4**

JIS 1978 používající ASCII shift-in escape sekvenci (CCSID 5055)

**B 4 A**

JIS 1978 používající ASCII shift-in escape sekvenci (CCSID 5055)

**B 4 R**

JIS 1978 používající JISROMAN shift-in escape sekvenci (CCSID 5053)

**B 5**

Hangeul (CCSID 934)

**B 6**

Korejština - standardní kód KSC-5601, verze 1989 (CCSID 949)

**B 7**

Tradiční čínština (5550) (CCSID 938)

**C**

Nastaví typ přenosu na jakýkoli CCSID (identifikátor kódové sady znaků), který je v systému nainstalován. Po písmenu C následuje číslo CCSID.

**E**

Nastaví typ přenosu na EBCDIC. Server k souboru nepřirazuje řízení vertikálního formátu. Pro EBCDIC podporuje server pouze předvolený formát NON PRINT. Typ přenosu EBCDIC je určen pro výkonný přenos dat mezi systémy, které používají EBCDIC jako svou interní reprezentaci dat.

**F**

IBM EBCDIC Kanji (CCSID 5035)

**F 1**

IBM EBCDIC Kanji (CCSID 5035)

## I

Nastaví typ přenosu na obraz. Je-li zadán typ přenosu obraz, data jsou reprezentována jako řetězec bitů, pakovaný do bajtů po 8 bitech. Typ přenosu obraz účinně ukládá a načítá soubory a přenáší binární data jako například kód objektu.

### Jděte na:

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## USER (Odeslat uživatelské přihlašovací ID na server)

### Podpříkaz FTP serveru

USER jménouživatele

#### jménouživatele

Uživatelský profil na serveru iSeries<sup>(TM)</sup>.

### Poznámka:

Byl-li podpříkaz USER úspěšný a je-li systém serveru iSeries nakonfigurován na zabezpečení dat pomocí hesla, potom server odešle klientu odpověď s výzvou k zadání hesla. Klient serveru odešle heslo prostřednictvím podpříkazu serveru PASS. Má-li server nastavenou úroveň zabezpečení 10, heslo není požadováno.

### Jděte na:

- Podpříkazy FTP serveru.
- Konvence syntaxe FTP serveru.

## Podpříkazy FTP klienta

Podpříkazy FTP (File Transfer Protocol) dávají FTP klientu pokyny k přenosu souborů z jednoho počítače na druhý. Pomocí podpříkazů FTP klienta můžete navázat spojení se vzdáleným FTP serverem, procházet knihovny a adresáři a vytvářet, mazat a přenášet soubory.

Popis podpříkazů klienta a jejich syntaxi najdete vždy pod příslušným heslem v tomto tématu.

FTP klient iSeries<sup>(TM)</sup> podporuje tyto podpříkazy. Následující tabulka uvádí přehled podpříkazů klienta, schválené zkratky a funkci každého podpříkazu.

Podpříkaz	Co dělá
?	Popisuje, jak používat FTP.
ACCT	Odešle do vzdáleného systému informaci o účtu uživatele.
APPEND	Přidá lokální člen souboru do souboru ve vzdáleném systému.
ASCII	Nastaví typ přenosu souboru na formát ASCII.
BINARY	Nastaví typ přenosu souboru na formát BINARY.
CD	Změní pracovní adresář ve vzdáleném systému.
CDUP	Přejde do nadřazeného adresáře ve vzdáleném systému.
CLOSE	Ukončí relaci se vzdáleným systémem.
DEBUG	Zapne nebo vypne ladění.

<b>Podpříkaz</b>	<b>Co dělá</b>
DEBUG	Změní hodnoty časové prodlevy u klienta.
DELETE	Vymaže soubor ze vzdáleného systému.
DIR	Zobrazí adresáře a soubory ze vzdáleného systému.
EBCDIC	Nastaví typ přenosu souboru na formát EBCDIC.
GET	Zkopíruje soubor ze vzdáleného do lokálního systému.
HELP	Získá informace o podpříkazech FTP klienta.
LCD	Změní pracovní adresář v lokálním systému.
LOCSITE	Udává informace o lokálním počítači.
LOCSTAT	Zobrazí informace o lokálním stavu.
LPWD	Zobrazí pracovní adresář v lokálním systému.
LS	Vypíše seznam jmen souborů ze sady souborů ve vzdáleném systému.
LTYPE	Udává typ přenosu souboru v lokálním systému.
MDELETE	Vymaže více souborů ze systému serveru.
MGET	Zkopíruje soubor nebo soubory ze vzdáleného systému.
MKDIR	Vytvoří adresář nebo podadresář.
MODE	Udává formát dat pro přenos souborů.
MPUT	Odešle lokální soubor nebo soubory do vzdáleného systému.
NAMEFMT	Udává formát jména souboru, který se použije.
NOOP	Kontroluje odezvu serveru.
NULLFLDS	Povoluje pole NULL.
OPEN	Otevře připojení k FTP serverem.
PASS	Odešle heslo uživatele.
PUT	Zkopíruje lokální člen souboru do vzdáleného systému.
PWD	Zobrazí aktuální adresář ve vzdáleném systému.
QUIT	Ukončí relaci FTP.
QUOTE	Odešle podpříkaz na FTP server.
REINITIALIZE	Znovu spustí relaci ve vzdáleném systému.
RENAME	Přejmenuje soubor ve vzdáleném systému.
RESET	Vymaže obsah fronty odpovědí serveru.
RMDIR	Vymaže adresář ve vzdáleném systému.
SECDATA	Udává úroveň zabezpečení použitou pro datové připojení, je-li nastaveno zabezpečené připojení s FTP serverem.
SECOPEN	Otevře zabezpečené řídicí připojení k FTP serveru pomocí zadaného protokolu zabezpečení.
SENDPASV	Uvádí, zda byl odeslán podpříkaz PASV.
SENDPORT	Uvádí, zda byl odeslán podpříkaz PORT.
SENDSITE	Uvádí, zda byl odeslán podpříkaz SITE.
SITE	Odešle informace, které použije vzdálený systém.
STATUS	Získá informace o stavu ze vzdáleného systému.
STRUCT	Udává členění souboru u odesílaných dat.



Podpříkaz	Co dělá
SUNIQUE	Řídí nahrazování souborů.
SYSCMD	Spouští CL příkaz v lokálním systému bez ukončení FTP.
SYSTEM	Zobrazí OS vzdáleného systému.
TYPE	Udává typ přenosu souboru.
USER	Odešle ID uživatele do vzdáleného systému.
VERBOSE	Řídí zobrazování odpovědí FTP serveru.

### Související témata:

- Konvence syntaxe FTP klienta.
- Podpříkazy FTP serveru.
- Stavové zprávy FTP serveru : Přehled běžných kódů odpovědí a jejich význam.

## ACCT (Odeslat informace o účtu)

### Podpříkaz FTP klienta

Některé systémy vyžadují k aktivaci určitých systémových funkcí informace o účtu. O tyto informace požádá vzdálený systém. K odeslání informací o účtu použijte podpříkaz FTP klienta ACCT:

```
ACCT informace-o-úctu
```

#### **informace-o-úctu**

Řetězec, který představuje účet uživatele. Informace o účtu mohou mít formu hesla, které hostitelský systém používá k poskytnutí určitých oprávnění. Toto heslo není vaše uživatelské heslo, ale je to heslo ve vzdáleném systému.

Například TCP/IP v operačním systému IBM<sup>(R)</sup> VM (Virtual Machine) může vyžadovat heslo pro přístup ke čtení a zápis na minidisky. Pomocí podpříkazu ACCT můžete dodat heslo pro minidisk aktuálního adresáře. Je-li vzdáleným systémem server iSeries<sup>(TM)</sup>, podpříkaz ACCT neprovede žádnou akci.

### Jděte na:

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## APPEND (Přidat člen lokálního souboru do vzdáleného souboru)

### Podpříkaz FTP klienta

Chcete-li přidat člen lokálního souboru, dokumentu nebo jiného souboru systému souborů do vzdáleného souboru, použijte podpříkaz FTP klienta APPEND v tomto formátu:

```
APpend lokální_soubor [vzdálený_soubor]
```

#### **lokální\_soubor**

Jméno členu lokálního souboru, dokumentu nebo jiného souboru iSeries<sup>(TM)</sup>. Jméno souboru hierarchického systému souborů (HFS - Hierarchical File System), které se přidá do adresáře ve vzdáleném systému. Informace o pojmenování souborů najdete v tématu NAMEFMT (Vybrat formát pojmenování souboru).

### **vzdálenýsoubor**

Jméno souboru ve vzdáleném systému. Pokud vzdálený soubor nezádáte, FTP klient použije předvolené jméno. Informace o tom, jak FTP vytváří předvolená jména, najdete v tématu Předvolená jména souborů v podpříkazech klienta pro přenos.

Jestliže zadaný soubor na vzdáleném serveru neexistuje, FTP server jej vytvoří.

Abyste mohli přidat soubor do vzdáleného systému, musíte mít oprávnění k zápisu do tohoto systému. Příslušné informace o účtu můžete poskytnout pomocí podpříkazu ACCT (Odeslat informace o účtu).

Předvoleným režimem při kopírování souboru je proud dat. Tento režim můžete změnit pomocí podpříkazu MODE. V případě, že vzdálený soubor má formát s pevně stanoveným záznamem, server zachová formát souboru i délku záznamu vzdáleného souboru. Záznamy z lokálního členu souboru se zkrátí nebo rozšíří mezerami, je-li to nezbytné.

#### **Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

### **ASCII (Změnit typ souboru na ASCII)**

#### **Podpříkaz FTP klienta**

AScii
-------

Existují dva základní typy souborů, které můžete použít při přenosu souborů prostřednictvím FTP: ASCII a BINARY. ASCII soubory obsahují prostý text. Mají přípony typu .txt nebo nemají žádnou příponu. Binární soubory jsou programy nebo jiné než textové soubory, které jsou uloženy ve formátu souborů odpovídajícím aplikaci, která je vytvořila, nebo ve formátu archivních nebo komprimovaných souborů.

Typ přenosu ASCII použijte při přenosu textových souborů do nebo z ASCII systému, který nepodporuje reprezentaci znaků EBCDIC. ASCII je předvolený typ přenosu. Server nepřirazuje k souboru řízení vertikálního formátu. ASCII podporuje pouze předvolený formát NON PRINT.

#### **Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

### **BINARY (Nastavit typ přenosu na obraz)**

#### **Podpříkaz FTP klienta**

Binary
--------

Existují dva základní typy souborů, které můžete použít při přenosu souborů prostřednictvím FTP: ASCII a BINARY. ASCII soubory obsahují prostý text. Mají přípony typu .txt nebo nemají žádnou příponu. Binární soubory jsou programy nebo jiné než textové soubory, které jsou uloženy ve formátu souborů odpovídajícím aplikaci, která je vytvořila, nebo ve formátu archivních nebo komprimovaných souborů.

Přenášíte-li binární data do existujícího souboru na serveru iSeries<sup>(TM)</sup>, pak délka záznamu odpovídá délce záznamu tohoto existujícího souboru. Velikost tohoto stávajícího souboru by měla být dostatečná pro umístění nových dat. Pokud soubor na serveru iSeries dosud neexistuje, FTP zvolí délku záznamu za vás.

Některé soubory, jako například soubory typu save, vyžadují přenos v podobě binárního obrazu. Když parametr TYPE není nastaven na hodnotu binary, potom při pokusu o přenos podobných souborů obdržíte zprávu, která vás požádá, abyste použili binární formát.

**Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

**CD (Změnit pracovní adresář nebo knihovnu)****Podpříkaz FTP klienta**

Chcete-li změnit pracovní adresář, knihovnu nebo skupinu souborů ve vzdáleném systému, použijte podpříkaz FTP klienta CD:

CD adresář

**adresář**

Jméno adresáře souborů, knihovny nebo jiného systémově závislého označovače skupiny souborů ve vzdáleném systému.

Je-li vzdáleným systémem server iSeries<sup>(TM)</sup>, pak tento podpříkaz změní aktuální knihovnu nebo adresář. Abyste zjistili, které adresáře jsou dostupné ve vzdáleném systému, použitím podpříkazu DIR (Directory) získáte jejich výpis.

S podpříkazem DIR zacházejte opatrně. Další informace a doporučení najdete v popisu podpříkazu DIR (Vypsat seznam položek adresáře, knihoven nebo souborů).

**Poznámka:**

Jestliže použijete podpříkaz CD (nebo LCD) k přechodu z jednoho systému souborů iSeries na jiný, musíte zadat i kořenový adresář systému souborů, který obsahuje nový aktuální adresář.

**Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

**CLOSE (Ukončit relaci FTP se vzdáleným systémem)****Podpříkaz FTP klienta**

Chcete-li ukončit svoji relaci se vzdáleným systémem a ponechat FTP v lokálním systému iSeries<sup>(TM)</sup> aktivní, použijte podpříkaz FTP klienta CLOSE.

CLOSE

Podpříkaz CLOSE vám umožňuje zůstat v prostředí FTP, abyste mohli otevřít jinou FTP relaci s dalším systémem. Použitím podpříkazu OPEN vytvoříte nové připojení se stejným nebo jiným vzdáleným systémem. Podpříkazem QUIT ukončíte služby FTP a vrátíte se do prostředí iSeries, ze kterého byl FTP spuštěn.

**Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## DEBUG (Vytvořit záznam z trasování klienta a řídit zobrazení podpříkazů serveru posílaných do vzdáleného systému)

### Poznámka:

Trasování FTP klienta byste měli používat pouze k nahlašování problémů softwaru IBM. Tato funkce může negativně ovlivnit výkon systému.

Tato funkce je v operačním systému OS/400<sup>(R)</sup> k dispozici ve verzi V4R4 a vyšších.

### Podpříkaz FTP klienta

Chcete-li vytvořit nebo zobrazit záznam z trasování FTP klienta, použijte podpříkaz FTP klienta DEBUG. Podpříkaz DEBUG přepíná režim ladění programu. Uvede-li klient nepovinnou hodnotu ladění, použije se tato hodnota k nastavení úrovně ladění programu. Když je ladění programu aktivováno, klient zobrazí řetězec '>>>'. Chcete-li získat záznam z trasování FTP klienta, nastavte hodnotu ladění na 100.

```
DEBbug [hodnotaladění]
```

#### hodnota-ladění

Je-li hodnota-ladění rovna 0, je ladění programu vypnuto. Je-li hodnota-ladění kladné celé číslo, je ladění programu zapnuto. Nezádáte-li žádnou hodnotu, hodnota ladění se přepne z nuly na jedničku, nebo z kladného celého čísla na nulu.

#### 100

Zahájí se trasování FTP klienta. Trasování klienta pokračuje, dokud není hodnota DEBUG nastavena na nulu nebo dokud není ukončena činnost FTP klienta. Když server ukončí trasování, může nastat delší časová prodleva, než naformátuje data z trasování.

Chcete-li spustit trasování bezprostředně se spuštěním klienta, musíte vytvořit datovou oblast QTMFTPD100 v knihovně QTEMP pomocí příkazu:

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

Když je datová oblast QTMFTPD100 vytvořena, bude hodnota ladění nastavena na 100 a trasování FTP klienta se spustí. Účelem této funkce je umožnit ladění FTP klienta v situacích, kdy trasování FTP klienta *nelze* spustit podpříkazem DEBUG 100.

### Jděte na:

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## DEBUG (Změnit hodnoty časového limitu klienta)

### Podpříkaz FTP klienta

Chcete-li změnit časový limit klienta, když předvolené hodnoty časové prodlevy nejsou dostatečně dlouhé, aby bylo možné úspěšně dokončit přenos dat, použijte podpříkaz DEBUG. Tyto hodnoty byste měli měnit pouze v případě, kdy provoz sítě nebo jiné podmínky způsobují výrazné zpomalení přenosů.

Chcete-li změnit hodnoty časové prodlevy u FTP klienta, použijte podpříkaz FTP klienta DEBUG:

```
DEBbug T1 | T2 [ hodnota ]
```

#### T1

Změní nebo zobrazí časový limit FTP klienta pro čtení odpovědí ze serveru. Pokud FTP klient neobdrží očekávanou odpověď serveru v rámci tohoto časového limitu, klient uzavře řídicí spojení se serverem.

#### T2

Změní nebo zobrazí časový limit FTP klienta pro přenos dat. Pokud FTP klient neobdrží při datovém připojení očekávanou reakci v tomto časovém limitu, klient uzavře toto datové připojení k serveru.

#### **hodnota**

Časový limit v sekundách. Tato hodnota musí být celé kladné číslo větší než nula. Pokud tuto hodnotu vynecháte, zobrazí klient aktuální hodnotu časového limitu.

Například:

```
DEBUG T1 900
```

Tato hodnota nastaví časovou prodlevu klienta pro odpovědi serveru na 900 sekund.

#### **Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## **DELETE (Vymazat soubor ve vzdáleném systému)**

### **Podpříkaz FTP klienta**

Podpříkaz DELETE použijte, chcete-li vymazat soubor nebo člen databázového souboru ze vzdáleného systému. Formát podpříkazu FTP klienta DELETE je:

```
DELeTe vzdálenýsoubor
```

#### **vzdálenýsoubor**

Soubor ve vzdáleném systému, který chcete vymazat. Informace o tom, jak zadat jméno souboru, je-li vzdáleným systémem server iSeries<sup>(R)</sup>, najdete v popisu podpříkazu NAMEFMT (Vybrat formát pojmenování souboru).

Vzdálený systém může požádat o oprávnění k výmazu souboru. Na tento požadavek odpovězte podpříkazem ACCT (Odeslat informace o účtu).

Můžete se také podívat na:

- MDELETE (Vymazat více souborů ze vzdáleného systému).
- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## **DIR (Vypsát seznam položek adresáře, knihoven nebo souborů)**

### **Podpříkaz FTP klienta**

Podpříkaz FTP klienta DIR zobrazí knihovny a jejich obsah nebo seznam adresářů a položek adresářů ve vzdáleném systému. Použijte podpříkaz DIR (Directory) v tomto formátu:

```
Dlr [jméno] [(Disk]
```

#### **jméno**

Jméno adresáře nebo knihovny. Předvolba je celý aktuální adresář nebo knihovna. Aktuální knihovnu nebo adresář můžete nastavit pomocí podpříkazu CD (Změnit pracovní adresář). Způsob zadání skupiny vzdálených souborů závisí na systému. Většina systémů povoluje generickou hvězdičku \*. Je-li vzdáleným systémem například server iSeries<sup>(R)</sup>, pak: DIR MYLIB/MYFILE.\* zobrazí seznam všech členů MYFILE v knihovně MYLIB.

Povoleny jsou dva formáty jména souboru. Zde uvedený příklad používá NAMEFMT 0. Informace o pojmenování souborů v rámci FTP najdete v popisu podpříkazu NAMEFMT (Vybrat formát pojmenování souboru).

#### ( Disk

Uloží výsledky podpříkazu DIR do souboru \*CURLIB/DIROUTPUT.DIROUTPUT, místo aby je zobrazil na obrazovce.

Je-li vzdáleným systémem server iSeries, tyto informace obsahují:

- Pro databázové soubory: objekty \*FILE a členy.
- Pro soubory hierarchického systému souborů (HFS):
  - Všechny pořadače knihovny dokumentů (QDLS) a jejich obsah, kterým mohou být další pořadače nebo dokumenty.
  - Všechny optické nosiče (QOPT) a jejich obsah, kterým mohou být adresáře nebo soubory.

S podpříkazem DIR zacházejte opatrně. Zadáte-li podpříkaz DIR bez jakéhokoli parametru, server vytvoří výpis všech souborů aktuálního adresáře. To může být mnohem delší seznam, než byste chtěli.

Chcete-li získat pouze seznam jmen souborů v adresáři, použijte podpříkaz LS (Vypsat seznam jmen vzdálených souborů).

#### Jděte na:

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## EBCDIC (Změnit typ souboru na EBCDIC)

### Podpříkaz FTP klienta

```
EBcdic
```

Typ přenosu EBCDIC je užitečný, když se přenášejí soubory mezi systémy EBCDIC. Díky této možnosti se vyhnete nutnosti v obou systémech konvertovat data z EBCDIC na ASCII a naopak.

#### Jděte na:

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## GET (Zkopírovat soubor ze vzdáleného systému do lokálního systému)

### Podpříkaz FTP klienta

```
Get vzdálenýsoubor [lokálnísoubor]  
[(Replace)]
```

#### vzdálenýsoubor

Soubor, který chcete získat ze vzdáleného systému.

#### lokálnísoubor

Člen lokálního souboru, dokument nebo jiný soubor, který chcete vytvořit. Pokud nezadáte jméno lokálního souboru, FTP klient použije předvolené jméno. Informace o předvolených jménech najdete v tématu Předvolená jména souborů v podpříkazech klienta pro přenos.

### ( Replace

Přepíše lokální soubor, pokud již existuje. Pokud ne zadáte parametr (Replace, server lokální soubor nepřepíše.

Formát jména souboru, který použijete v příkazu GET, je závislý na systému souborů, v němž se soubor nachází.

- Jestliže se tento soubor nenachází v systému souborů knihoven (QSYS.LIB), musíte použít příkaz GET s formátem NAMEFMT 1:

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

- Jestliže se soubor nachází v systému souborů knihoven, použijte příkaz GET a parametr NAMEFMT nastavte na 0:

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

Za předpokladu, že vzdálený systém je server iSeries<sup>TM</sup>, získáte tímto příkazem člen YOURMBR souboru YOURFILE v knihovně YOURLIB a tento člen se uloží jako YOURMBR souboru YOURFILE do aktuálního adresáře vašeho lokálního systému. Více informací o definování aktuálního adresáře najdete v popisu příkazu LCD (Změnit pracovní knihovnu nebo adresář v lokálním systému).

### Poznámka:

Jsou-li součástí jména vzdáleného souboru apostrofy, ohraničte jméno souboru dalšími dvěma páry apostrofů. Následující příklad ukazuje, jak získat člen 'MEMBER.ONE' ze vzdáleného hostitelského systému. Další informace najdete v tématu Ohraničení parametrů příkazu.

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

### Jděte na:

- Příkazy FTP klienta.
- Konvence syntaxe FTP klienta.
- MGET (Zkopírovat více souborů ze vzdáleného systému do lokálního systému).
- PUT (Zkopírovat člen souboru z lokálního systému do souboru ve vzdáleném systému).
- MPUT (Odeslat více členů souboru z lokálního systému do vzdáleného systému).

## HELP (Získání nápovědy k příkazům FTP)

Příkaz HELP poskytuje informace o příkazech FTP, které používá jak lokální, tak vzdálený systém.

### Nápověda k příkazům FTP klienta

Chcete-li získat informace o příkazech FTP, které používá lokální systém, použijte příkaz HELP v tomto formátu:

```
Help [* | ALL | příkaz ]
```

#### \* nebo ALL

Zobrazí seznam všech příkazů FTP klienta.

#### příkaz

Poskytne podrobnou nápovědu pro zadaný příkaz klienta. Například HELP GET vám ukáže, jak přenést soubor ze vzdáleného systému do lokálního systému. Příkazy můžete zkrátit na smysluplnou předponu.

Pokud použijete příkaz HELP bez jakéhokoliv parametru, zobrazí se seznam příkazů a všeobecný popis, jaké informace nápověda obsahuje. Chcete-li získat kontextovou nápovědu, přesuňte na obrazovce nápovědy kurzor na požadovaný příkaz a stiskněte klávesu **Enter**.

Chcete-li získat seznam lokálních podpříkazů na serveru iSeries<sup>(TM)</sup>, zadejte:

HELP

Informace o nápovědě získáte podpříkazem ?.

### Nápověda k podpříkazům FTP serveru

Chcete-li získat informace o podpříkazech FTP ve vzdáleném systému, použijte podpříkaz HELP v tomto formátu:

```
Help SERVER [podpříkaz]
```

#### SERVER

Poskytne nápovědu, kterou vzdálený systém nabízí k podpříkazům FTP serveru. Je to obdobné, jako použití podpříkazu QUOTE s parametrem HELP. Podpříkaz QUOTE HELP zobrazí seznam FTP podpříkazů podporovaných vzdáleným systémem.

#### podpříkaz

Jméno podpříkazu serveru, o kterém chcete získat informace. Například podpříkazem HELP SERVER STOR požádáte server o nápovědu k podpříkazu STOR.

#### Poznámka:

Synonymem k HELP SERVER je RHELP. Například podpříkazy HELP SERVER SITE a RHELP SITE jsou rovnocenné.

Další informace najdete v popisu podpříkazu QUOTE (Odeslat podpříkaz na FTP server).

#### Jděte na:

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## LCD (Změnit pracovní knihovnu nebo adresář v lokálním systému)

### Podpříkaz FTP klienta

```
LCd jménocesty
```

#### jménocesty

Jméno knihovny, pořadače nebo adresáře v lokálním systému.

#### Poznámky:

1. Podpříkaz LCD nezmění aktuální položku knihovny ze seznamu knihoven.
2. Jestliže chcete pomocí podpříkazu CD (nebo LCD) změnit jeden systém souborů na jiný, musíte zadat "kořenový" adresář, například /QDLS nebo /QOPT.

#### Jděte na:

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## LOCSITE (Zadat informace pro lokální systém)

### Podpříkaz FTP klienta



Chcete-li zadat informace, které použije FTP klient k poskytování služeb pro něj specifických, použijte podpříkaz FTP klienta LOCSITE:

LOCSITE [ parametry ]

FTP klient iSeries<sup>(TM)</sup> podporuje následující parametry a volby parametrů podpříkazu LOCSITE:

#### **CRTCCSID \*CALC**

Nové databázové soubory, které byly vytvořeny během přenosů souborů ASCII, budou používat předvolený CCSID EBCDIC, který odpovídá CCSID použitému při přenosu souborů ASCII. Je to předvolená hodnota.

#### **CRTCCSID \*USER**

Nové databázové soubory, které byly vytvořeny během přenosů souborů ASCII, budou používat CCSID aktuální úlohy. Je-li však tento CCSID roven 65535, potom předvolený CCSID se určí podle ID jazyka aktuální úlohy.

#### **CRTCCSID \*SYSVAL**

Nové databázové soubory, které byly vytvořeny během přenosů souborů ASCII, budou používat CCSID určený systémovou hodnotou QCCSID.

#### **CRTCCSID [CCSID-číslo]**

Zadejte, jaký CCSID chcete používat při vytváření databázových souborů na klientu během přenosů souborů ASCII. Server ověří tuto hodnotu.

#### **CRTCCSID**

Zobrazí zprávu, která obsahuje aktuální nastavení CRTCCSID FTP klienta.

#### **TRIM 0**

Nastaví volbu Trim na OFF. Server bude odesílat koncové mezery databázových záznamů.

#### **TRIM 1**

Nastaví volbu Trim na ON. Server nebude odesílat koncové mezery databázových záznamů při přenosu databázových souborů, které používají souborové členění a proudový režim. Je to předvolená hodnota.

#### **TRIM 2**

Při tomto nastavení volby Trim server nebude odesílat koncové mezery databázových záznamů při žádných přenosech, tzn. ani při záznamovém členění a blokovém režimu.

#### **TRIM**

Zobrazí zprávu, která obsahuje aktuální nastavení volby TRIM FTP klienta.

#### **Poznámky:**

1. Před vznikem tohoto podpříkazu byly koncové mezery záznamů v systému souborů QSYS.LIB vždy před přenosem souboru do systému serveru odstraněny.
2. Na nastavení parametru TRIM se nebere ohled při binárním přenosu souborů (TYPE I). Při přenosu souborů typu TYPE I se mezery nikdy neořezávají, bez ohledu na nastavení parametru TRIM.



#### **DTAPROT C**

Nastaví proměnnou pro ochranu dat na C (Clear). Tato proměnná slouží k nastavení ochrany dat při

otevření zabezpečeného řídicího připojení. Podrobnější informace o tom, jak nastavit zabezpečení ochrany dat najdete v popisu podpříkazů SECDATA a SECOPEN.

### **DTAPROT P**

Nastaví proměnnou pro ochranu dat na P (Private). Tato proměnná slouží k nastavení ochrany dat při otevření zabezpečeného řídicího připojení.

### **DTAPROT**

Zobrazí zprávu, která obsahuje aktuální nastavení volby TRIM the klienta.



#### **Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

### **LOCSTAT (Zobrazit informace o lokálním stavu) Podpříkaz FTP klienta**

```
LOCStat
```

Zobrazí informace o stavu lokálního systému, které zahrnují:

- Aktuální nastavení podpříkazu SENDSITE.
- Aktuální nastavení podpříkazu SENDPORT.
- Jméno vzdáleného systému, číslo portu a stav přihlášení.
- Typ dat a režim přenosu.
- Hodnotu formátu jmen pro klienta i pro server.
- Nastavení hodnot pro režim VERBOSE.
- Nastavení hodnot pro režim DEBUG.

#### **Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

### **LS (Vypsat seznam jmen vzdálených souborů) Podpříkaz FTP klienta**

```
LS [jméno] [(Disk)]
```

#### **jméno**

Vzdálený adresář, soubor nebo knihovna, jejichž seznam chcete procházet. Je-li vzdáleným systémem server iSeries<sup>(TM)</sup>, server vypíše seznam souborů a jejich členů. Jako předvolba se vypíše seznam celého aktuálního adresáře, knihovny nebo pořadače. Chcete-li změnit aktuální adresář, knihovnu nebo pořadač, použijte podpříkaz CD (Změnit pracovní adresář nebo knihovnu). Specifikace vzdáleného souboru je závislá na systému.

#### **(Disk**

Uloží výsledky podpříkazu LS do souboru \*CURLIB/LSOUTPUT.LSOUTPUT, místo aby je zobrazil na obrazovce. Pokaždé, když zadáte parametr (Disk se stejnou \*CURLIB, server změní obsah členu LSOUTPUT.LSOUTPUT.

**Poznámka:**

Vrátí-li FTP server kód negativní odpovědi (550), nevytvorí se žádný člen LSOUTPUT. Vrátí-li FTP server kód kladné odpovědi (150) bez uvedení jména souboru, potom nebude člen LSOUTPUT obsahovat žádné záznamy.

Podpříkaz LS vypíše pouze seznam jmen souborů. Chcete-li získat úplný seznam položek adresáře i s dalšími informacemi o souborech, použijte podpříkaz DIR (Vypsat seznam položek adresáře, knihoven nebo souborů).

**Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

**LTYPE (Lokální typ)****Podpříkaz FTP klienta**

Chcete-li zadat typ přenosu souborů nebo reprezentaci dat pro přenos souborů, který bude platit pro lokální systém, použijte podpříkaz FTP klienta LTYPE v tomto formátu:

```
LType C ccsid#
```

**C**

Typ CCSID. Kódem této hodnoty je C.

**ccsid#**

Hodnota CCSID. Kódem této hodnoty je číslo CCSID v rozsahu 1 až 65533.

**Poznámka:**

Podpříkaz LTYPE je podobný podpříkazu TYPE (Zadat typ přenosu souboru). Podpříkaz LTYPE změní typ reprezentace dat pouze na straně klienta. Podpříkaz TYPE změní typ reprezentace dat jak na klientu, tak na serveru.

**Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

**MDELETE (Vymazat více souborů ze vzdáleného systému)****Podpříkaz FTP klienta**

```
MDelete {vzdálenýsoubor [vzdálenýsoubor...]}
```

**vzdálenýsoubor**

Soubor nebo soubory v systému serveru, které chcete vymazat.

**Poznámka:**

Je-li vzdáleným souborem soubor systému QSYS.LIB na serveru iSeries<sup>(TM)</sup>, server vymaže všechny členy tohoto fyzického souboru iSeries. Vlastní soubor zůstane.

Zde je příklad pro systém souborů knihoven ve formátu NAMEFMT 0:

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

Tento příkaz vymaže člen MBRA v souboru FILE1 v knihovně MYLIB a člen MBRB v souboru FILE2 v knihovně YOURLIB na vzdáleném serveru iSeries. Tentýž příklad ve formátu NAMEFMT 1:

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

Následuje příklad pro systém knihoven dokumentů ve formátu NAMEFMT 1:

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

Tento příkaz vymaže dokument PCSMENU.EXE v pořadači QIWSOS2 v knihovně služeb knihovny dokumentů, a také vymaže PCSFILE.EXE v pořadači PCSDIR v knihovně QDLS na serveru iSeries.

Chcete-li genericky vymazat více souborů, můžete použít hvězdičku (\*). Například v případě formátu NAMEFMT 0, je-li vzdáleným systémem server iSeries, zadejte:

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

Tento příklad vymaže všechny členy souboru MYFILE v knihovně MYLIB. Použití hvězdičky je platné pouze na konci znakového řetězce.

#### **Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## **MGET (Zkopírovat více souborů ze vzdáleného systému do lokálního systému)**

### **Podpříkaz FTP klienta**

#### **Jak MGET přenáší soubory:**

Chcete-li zkopírovat jeden nebo více vzdálených souborů, použijte podpříkaz FTP klienta MGET v následujícím formátu. Pro každý vzdálený soubor, který chcete přenést, se provede samostatný podpříkaz GET. Server automaticky vytvoří jméno odpovídajícího lokálního souboru podle předvolených pravidel pro pojmenování.

K určení, kam uložit soubory, používá podpříkaz FTP klienta MGET následující metodu:

- Podpříkaz MGET vždy umístí soubory do aktuální knihovny nebo adresáře.
- Pokud uživatel vydal podpříkaz LCD, server použije tuto knihovnu nebo adresář.
- Pokud uživatel nevydal podpříkaz LCD, server nastaví aktuální adresář takto:
  - Má-li úloha uživatele nastavenou aktuální knihovnu, použije se tato knihovna jako aktuální adresář pro FTP.
  - Nemá-li úloha uživatele nastavenou aktuální knihovnu, server použije QGPL jako aktuální adresář.

```
MGet {vzdálenýsoubor  
[vzdálenýsoubor...]}[(Replace)]
```

#### **vzdálenýsoubor**

Soubor nebo soubory, které chcete získat ze vzdáleného systému.

#### **( Replace**

Přepíše stávající soubor na vašem lokálním systému. Pokud soubor v lokálním systému již existuje a nepoužijete volbu Replace, stávající soubor se nepřepíše. Jméno lokálního souboru, do kterého je vzdálený soubor zkopírován, se vytvoří automaticky. Další informace najdete v popisu podpříkazu GET (Zkopírovat soubor ze vzdáleného systému do lokálního systému).

Chcete-li zkopírovat všechny členy souboru do své aktuální knihovny nebo adresáře, můžete použít hvězdičku (\*). Například, je-li vzdáleným systémem server iSeries<sup>(TM)</sup>:

- MGET MYLIB/MYFILE.\* zkopíruje všechny členy souboru MYFILE v knihovně MYLIB ve vzdáleném systému do vaší aktuální knihovny v lokálním systému.
- MGET /QSYS.LIB/MYLIB.LIB/MYFILE.FILE/\*.MBR by byl obdobou tohoto příkazu ve formátu NAMEFMT 1.
- MGET /QOPT/PICTURES/IMAGES/\*. zkopíruje všechny soubory v adresáři IMAGES na optickém nosiči PICTURES do vaší aktuální knihovny (nebo adresáře) v lokálním systému.
- MGET TESTFILE.A\* zkopíruje všechny členy ze souboru TESTFILE, které začínají písmenem A.
- MGET /QDLS/QISSOS2/A\* zkopíruje všechny dokumenty z pořadače QISSOS2, které začínají písmenem A.

#### **Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

### **MKDIR (Vytvořit adresář)**

#### **Podpříkaz FTP klienta**

```
MKdir jménocesty
```

#### **jménocesty**

Jméno adresáře souborů, knihovny nebo jiného systémově závislého označovače skupiny souborů ve vzdáleném systému.

#### **Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

### **MODE (Zadat režim přenosu dat)**

#### **Podpříkaz FTP klienta**

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

#### **B**

Určuje blokový režim. V tomto režimu server přenáší data jako sérii datových bloků, začínající jedním nebo více bajty hlavičky. Pokud přenášíte data v blokovém režimu, reprezentace dat musí být EBCDIC.

#### **S**

Určuje proudový režim. V tomto režimu server přenáší data jako proud bajtů. V proudovém režimu můžete použít libovolnou reprezentaci dat.

#### **Poznámky:**

1. Proudový režim je předvolený režim přenosu, který používá FTP. Některé systémy nepodporují blokový režim.
2. Pokud nepovinný parametr vynecháte, klient zobrazí stávající hodnotu režimu.

#### **Jděte na:**

- Podpříkazy FTP klienta.

- Konvence syntaxe FTP klienta.

## **MPUT (Odeslat více členů souboru z lokálního systému do vzdáleného systému)** **Podpříkaz FTP klienta**

Chcete-li odeslat jeden nebo více lokálních souborů do vzdáleného systému, použijte podpříkaz FTP klienta MPUT. Pro každý lokální soubor, který chcete přenést, klient provede samostatný podpříkaz PUT. Jméno příslušného vzdáleného souboru se vytvoří podle předvolených pravidel pro pojmenování.

```
Mput {lokální_soubor [lokální_soubor...]}
```

### **lokální\_soubor**

Zadejte jeden nebo více členů souboru lokálního systému souborů knihoven nebo souborů jiného systému souborů podporovaného FTP, které chcete přenést do vzdáleného systému. Klient automaticky vygeneruje jméno, které bude přiděleno souboru ve vzdáleném systému.

### **Poznámka:**

Jestliže vzdálený soubor již existuje, jeho obsah se přepíše obsahem *lokálního\_souboru*, pokud není podpříkazem SUNIQUE (Řídit přepisování souborů) zadáno jinak.

Informace o tom, jak zadat jméno souboru, je-li vzdáleným systémem server iSeries<sup>(R)</sup>, najdete v popisu podpříkazu NAMEFMT (Vybrat formát pojmenování souboru). V tomto příkladu je použit formát NAMEFMT 0:

```
Mput MYLIB/FILE1.MBR1 MYLIB/FILE1.MBR2
```

Členy MBR1 a MBR2 souboru FILE1 v knihovně MYLIB se odešlou do vzdáleného systému.

V tomto příkladu je použit formát NAMEFMT 1:

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

Dokumenty PCSMENU.EXE a PCSMENU2.EXE z počítače QIWSOS2 se odešlou do vzdáleného systému.

Chcete-li odeslat všechny členy souboru, můžete použít hvězdičku (\*). Například MPUT MYLIB/MYFILE.\* přenesou všechny členy souboru MYFILE v knihovně MYLIB. Další informace najdete v popisu podpříkazu MGET (Zkopírovat více souborů ze vzdáleného systému do lokálního systému).

### **Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.
- PUT (Zkopírovat člen souboru z lokálního systému do souboru ve vzdáleném systému).

## **NAMEFMT (Vybrat formát pojmenování souboru)** **Podpříkaz FTP klienta**

Chcete-li zvolit, který formát jmen souborů se bude používat v lokálním a vzdáleném systému (jedná-li se o server iSeries<sup>(TM)</sup>), použijte podpříkaz FTP klienta NAMEFMT:

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

### **0**

Formát jména určený pouze pro databázové soubory systému souborů knihoven. Obecný formát je: [jménoknihovny/]jménosouboru[.jménočlenu]

## 1

Formát jména pro všechny systémy souborů podporované protokolem FTP, včetně systému souborů knihoven. Chcete-li pracovat se všemi systémy souborů iSeries, musíte nastavit formát jména na '1'.

Soubory systému souborů knihoven v tomto formátu jména mají tvar:

```
[/QSYS.LIB/][jménoknihovny.LIB/]jménosouboru.FILE[/jměnočlenu.MBR]
```

Pro soubory typu save můžete také použít formát:

```
/QSYS.LIB/jménoknihovny.LIB/jménosouboru.SAVF
```

Soubory systému souborů služeb knihovny dokumentů mají formát:

```
[/QDLS/][{jménopořadače[.ext]/}]jménosouboru[.ext]
```

Optické nosiče mají formát:

```
/QOPT/jménonosiče/jménoadresáře/jménosouboru.ext
```

### Poznámky:

1. Formát jmen můžete nastavit na 0 pouze tehdy, je-li pracovním adresářem databázová knihovna.
2. Zadáte-li příkaz NAMEFMT bez jakéhokoli parametru, klient zobrazí aktuální formát jmen.

Můžete se také podívat na:

- Systémy souborů a konvence pojmenování podporované FTP.
- Příkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## NULLFLDS (Povolit přenos souborů s poli NULL)

### Podpříkaz FTP klienta

Pomocí tohoto příkazu můžete povolit či zakázat přenos databázových souborů v lokálním i vzdáleném systému iSeries<sup>™</sup>, které obsahují pole s hodnotou NULL.

NULLflds [ 0   1 ]
--------------------

Zadáte-li parametr, platné hodnoty jsou:

## 0

Nepovolit přenos databázových souborů, které obsahují pole NULL. Je to předvolená hodnota.

## 1

Povolit přenos databázových souborů, které obsahují pole NULL.

### Poznámky:

1. Přenos souborů, které obsahují pole NULL, vyžaduje, aby toto nastavení bylo povoleno jak u klienta, tak u serveru. Cílový soubor musí existovat dříve, než dojde k přenosu souboru. Kromě toho musí mít cílový soubor stejnou definici souboru jako soubor zdrojový.
2. Když přenášíte soubor, který obsahuje pole NULL do jiného systému než iSeries, nebo když typ přenosu dat vede ke konverzi kódové stránky dat, může dojít k nepředvídatelným výsledkům.
3. Zadáte-li příkaz NULLFLDS bez parametru, klient zobrazí aktuální nastavení.

**Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

**OPEN (Navázat spojení s FTP serverem ve vzdáleném systému)****Podpříkaz FTP klienta**

```
Open jménosystému [čísloportu]
```

**jménosystému**

Jméno nebo internetová adresa vzdáleného systému.

**čísloportu**

Číslo portu, které se bude v této relaci používat, dokud server neukončí spojení. Tato volba je volitelná. Jestliže nezadáte číslo portu, server si nějaké zvolí sám.

Jakmile otevřete spojení se vzdáleným systémem, nemůžete se připojit k jinému systému, dokud neuzavřete aktuální relaci.

**Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

**PASS (Odeslat heslo)****Podpříkaz FTP klienta**

```
PAss heslo
```

**heslo**

Řetězec, který představuje vaše heslo.

Tomuto podpříkazu musejí předcházet podpříkazy OPEN a USER. U některých systémů tento podpříkaz dokončuje vaši identifikaci při řízení přístupu. Tento podpříkaz nepotřebujete, pokud budete k zadání hesla vyzváni během připojování nebo přihlašování se k serveru.

**Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

**PUT (Zkopírovat člen souboru z lokálního systému do souboru ve vzdáleném systému)****Podpříkaz FTP klienta**

```
PUt lokální soubor [vzdálený soubor]
```

**lokální soubor**

Jméno lokálního členu souboru systému souborů knihoven, souboru typu save, dokumentu nebo jiného souboru.

**vzdálený soubor**

Jméno doručeného souboru ve vzdáleném systému. Pokud nezadáte jméno vzdáleného souboru, FTP server dodá předvolené jméno. Informace o předvolených jménech najdete v tématu Předvolená



jména souborů v podpříkazech klienta pro přenos. Jestliže vzdálený soubor již existuje, jeho obsah se přepíše obsahem lokálního souboru, pokud není podpříkazem SUNIQUE (Řídit přepisování souborů) určeno jinak.

Chcete-li odeslat soubor do vzdáleného systému, musíte mít nadefinován aktuální pracovní adresář s právem zápisu.

V tomto příkladu je podpříkaz PUT použit k přenosu členu souboru:

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

Člen MYMBR souboru MYFILE v knihovně MYLIB se odešle do vzdáleného systému.

V tomto příkladu se odešle dokument PCSMENU.EXE z pořadače QIWSOS2 v systému souborů služeb knihovny dokumentů do vzdáleného systému:

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

**Poznámka:**

Jsou-li součástí jména vzdáleného souboru apostrofy, ohraničte jméno souboru dalšími dvěma páry apostrofů. V následujícím příkladu se odešle 'MEMBER.ONE' jako jméno souboru do vzdáleného systému. Další informace najdete v tématu Ohraničení parametrů podpříkazu.

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

**Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.
- MPUT (Odeslat více členů souboru z lokálního systému do vzdáleného systému).
- GET (Zkopírovat soubor ze vzdáleného systému do lokálního systému).
- MGET (Zkopírovat více souborů ze vzdáleného systému do lokálního systému).

## **PWD (Zobrazit aktuální adresář, pořadač nebo knihovnu)**

### **Podpříkaz FTP klienta**

Chcete-li zobrazit aktuální adresář nebo knihovnu ze vzdáleného systému, použijte podpříkaz FTP klienta PWD:

```
PWd
```

Je-li vzdáleným systémem server iSeries<sup>(TM)</sup>, server zobrazí vaši aktuální knihovnu nebo adresář systému souborů ve vzdáleném systému. Server zobrazí také pracovní adresář v uvozovkách. Chcete-li změnit aktuální knihovnu nebo adresář ve vzdáleném systému, použijte podpříkaz CD (Změnit pracovní adresář).

**Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## **QUOTE (Odeslat podpříkaz na FTP server)**

### **Podpříkaz FTP klienta**

```
QUOTE řetězec
```

## řetězec

Podpříkaz serveru, který chcete odeslat a zpracovat na vzdáleném FTP serveru. FTP server odešle řetězec v přesně stejném znění na vzdálený FTP server.

### Poznámky:

1. Klient potřebuje podpříkaz QUOTE ke spuštění speciálního podpříkazu FTP serveru iSeries<sup>(TM)</sup> RCMD (Odeslat CL příkaz do systému FTP serveru). Chcete-li například zapsat protokol úlohy serveru do souboru pro souběžný tisk, zadejte:

```
QUOTE RCMD DSPJOBLOG
```

K práci s protokolem úlohy můžete použít příkaz WRKSPLF. Pamatujte si, že musíte zadat profil uživatele, který se přihlásil k FTP serveru, spouštíte-li příkaz WRKSPLF z jiného uživatelského profilu.

2. FTP server iSeries omezuje řetězec na 1000 znaků.
3. Vše, co do podpříkazu QUOTE zadáte, se přenese na server. Zadáte-li například:

```
QUOTE CWD 'SYS1'
```

server obdrží

```
CWD 'SYS1'
```

Tímto způsobem můžete například získat nápovědu ze vzdáleného systému. Zadejte:

```
QUOTE HELP
```

Server odešle podpříkaz HELP do vzdáleného hostitelského systému, který vrátí zobrazení všech podpříkazů, které podporuje. Zobrazené informace se liší v závislosti na typu vzdáleného hostitelského systému.

Je nutné poznamenat, že podpříkazy serveru zadané prostřednictvím podpříkazu QUOTE, ovlivňují pouze server, ale podobné podpříkazy klienta mohou ovlivnit jak klienta, tak server. Například podpříkaz klienta REIN odešle na server podpříkaz serveru REIN a současně znovu inicializuje některé stavové proměnné klienta. Podpříkaz QUOTE REIN odešle pouze podpříkaz REIN na server, ale nezmění žádnou stavovou proměnnou klienta.

### Poznámka:

Buďte opatrní, když používáte podpříkaz QUOTE k přímému zadávání podpříkazů serveru, aby nedošlo k neočekávaným výsledkům. Používejte podpříkaz QUOTE pouze ve zvláštních situacích, kdy nelze použít jiné podpříkazy klienta. Příkladem je situace, kdy potřebujete použít některý ze zvláštních podpříkazů serveru iSeries, jako je například CTRL.

### Jděte na:

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## REINITIALIZE (Znovu nastavit relaci mezi systémy)

### Podpříkaz FTP klienta

```
REInitialize
```

Pokud server podporuje podpříkaz REINITIALIZE, relace uživatele se serverem se ukončí. Server bude ve stejném stavu, v jakém byl v okamžiku navázání spojení, a uživatel se musí znovu přihlásit, aby mohl pokračovat.

Přenosy dat, které již probíhají, se mohou dokončit, ještě než se ukončí relace uživatele.

**Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## **RENAME (Přejmenovat soubor ve vzdáleném systému)**

### **Podpříkaz FTP klienta**

```
REname původníjméno novějmeno
```

**původníjméno**

Současné jméno vzdáleného souboru.

**novějmeno**

Nové jméno vzdáleného souboru. Pokud soubor zadaný v parametru *novějmeno* již existuje, nový soubor ho nahradí.

V tomto příkladu se soubor SPORTSCAR.BMP v adresáři IMAGES na optickém nosiči PICTURES přejmenuje na CAR.BMP:

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

**Poznámka:** Na serveru iSeries<sup>(TM)</sup> nemůžete přejmenovat soubor na odlišný systém souborů.

**Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## **RESET (Obnovit)**

### **Podpříkaz FTP klienta**

Chcete-li vyčistit frontu odpovědí serveru, použijte podpříkaz FTP klienta RESET:

```
REset
```

Tento podpříkaz resynchronizuje pořadí podpříkazů serveru a odpovědí se vzdáleným FTP serverem. Obnovení synchronizace může být nutné po porušení FTP protokolu vzdáleným serverem.

**Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## **RMDIR (Odstranit adresář)**

### **Podpříkaz FTP klienta**

```
RMdir jménocesty
```

## **jménocesty**

Jméno adresáře souborů, knihovny nebo jiného systémově závislého označovače skupiny souborů ve vzdáleném systému. Můžete odstranit pouze prázdné adresáře hierarchického systému souborů (HFS). Knihovny server vymaže bez dalších podmínek.

### **Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## **SECData (Nastavit zabezpečení ochrany dat)**

### **Podpříkaz FTP klienta**

Chcete-li zadat úroveň zabezpečení pro datové připojení, když je již navázáno řídicí připojení ke vzdálenému systému, použijte příkaz SECData v tomto tvaru:

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

**Poznámka:** Synonymem k tomuto podpříkazu je SData.

### **C**

Úroveň ochrany datového kanálu je nastavena na 'clear' (čitelný). Toto připojení **není** zabezpečené. Lze je použít k přenosu předem zašifrovaných dat nebo dat, která nejsou důvěrná.

### **P**

Úroveň ochrany datového kanálu je nastavena na 'private' (soukromý). Toto připojení je zabezpečené. Než se přes něj mohou přenášet data, je třeba mezi klientem a serverem vyjednat TLS.

1. Nezadáte-li žádný parametr, podpříkaz SECData zobrazí aktuální hodnotu použité úrovně zabezpečení ochrany dat.
2. Úroveň ochrany je na počátku nastavena na hodnotu, která byla uvedena v parametru DTAPROT CL příkazu STRTCPFTP při vytvoření řídicího připojení.
3. Podpříkaz SECData lze použít pouze tehdy, je-li zabezpečeno řídicí připojení.
4. Při každém úspěšném nastavení úrovně ochrany dat podpříkazem SECData je vydán podpříkaz serveru PROT.
5. Když nastavuje úroveň ochrany dat, posílá podpříkaz SECData na server podpříkazy PBSZ a PROT. Podpříkaz SECData také pro každý úspěšný podpříkaz PROT nastaví proměnnou klienta. Tato proměnná představuje poslední úroveň ochrany dat (C nebo P) přijatou serverem. Tato proměnná slouží k nastavení úrovně ochrany dat, když podpříkaz SECOpen otvírá zabezpečené řídicí připojení. Tuto proměnnou můžete změnit pomocí volby LOCSITE DTAPROT (See 65).
6. Parametry 'C' a 'P' v podpříkazu SECData jsou stejné jako parametry používané v podpříkazu serveru PROT.

### **Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## **SECOpen (Nastavení zabezpečení ochrany dat)**

### **Podpříkaz FTP klienta**

Podpříkaz FTP klienta SECOpen otevře zabezpečené řídicí připojení k FTP serveru s použitím zadané volby zabezpečení. Syntaxe tohoto podpříkazu je:

```
SECOpen jménosystému [čísloportu] [volba_zabezpečení]
```

**Poznámka:** SOpen je synonymem k SECOPEN.

### **jménosystému**

Zadejte jméno nebo internetovou adresu vzdáleného systému.

### **čísloportu**

Zadejte číslo portu pro toto připojení.

### **Poznámky:**

- Je-li tento parametr vynechán a je zadána volba (SSL, použije se port číslo 21.
- Je-li tento parametr vynechán a je zadána volba (IMPLICIT, použije se port číslo 990.
- Jsou-li vynechány parametry čísloportu i volba\_zabezpečení, předpokládá se port číslo 21 a volba (SSL.

### **volba\_zabezpečení**

Zadejte typ zabezpečení, který se má použít.

#### **(SSL**

Připojení k FTP serveru bude zabezpečené pomocí SSL. Při vytváření připojení se použije podpříkaz serveru AUTH (Authorization).

#### **(IMPLICIT**

Připojení k FTP serveru bude zabezpečené "implicitně" pomocí SSL/TLS. "Implicitní" připojení SSL se vytvoří, aniž by se na server odeslaly podpříkazy serveru AUTH, PBSZ a PROT. V tom případě musí být server nakonfigurován tak, aby očekával vyjednávání SSL/TLS na zadaném čísle portu.

V případě "implicitního" SSL se bude server chovat tak, jako kdyby klient poslal tyto podpříkazy a parametry:

- AUTH SSL
- PBSZ 0
- PROT P

### **Poznámka:**

Nezadáte-li parametr volba\_zabezpečení, předpokládá se volba (SSL. Při použití portu číslo 990 se předpokládá volba (IMPLICIT.

### **Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## **SENDPASV (Určit, zda se má odeslat podpříkaz PASV)**

### **Podpříkaz FTP klienta**

Chcete-li určit, zda se má či nemá odeslat podpříkaz PASV na FTP server během přenosu dat nebo při zadávání podpříkazů DIR a LS, zadejte podpříkaz FTP klienta SENDPASV:

```
SENDPASv [ 0 | 1 ]
```

Pokud nezadáte žádný parametr, funguje SENDPASV jako přepínač. Hodnota SENDPASV se přepíná z 1 (zapnuto) na 0 (vypnuto) nebo z 0 na 1.

Pokud zadáváte i parametr, platné hodnoty jsou:

**0**

Podpříkaz PASV se neodesílá.

**1**

Podpříkaz PASV se odesílá. Je to předvolená hodnota.

Předvolba v systému iSeries<sup>(TM)</sup> je "zapnuto", tzn. podpříkaz PASV se odesílá. Je-li SENDPASV "vypnuto", server neodesílá podpříkaz PASV.

**Poznámky:**

1. Tento podpříkaz podporuje standard RFC 1579, "Firewall-Friendly FTP". Použití podpříkazu PASV k navázání datového připojení je vhodnější v případě, že se přenos dat musí uskutečnit přes ochrannou bariéru (firewall). V některých případech není přenos dat přes ochrannou bariéru bez použití podpříkazu PASV možný.
2. Některé FTP servery nemusejí podporovat podpříkaz PASV. Pokud se tak stane a SENDPASV je nastaven na "zapnuto", FTP klient zobrazí zprávu, že server nepodporuje PASV. Systém se potom pokusí navázat datové připojení bez odeslání podpříkazu PASV.
3. Je-li SENDPASV nastaven na "vypnuto" nebo není povolen, server odesílá podpříkaz PORT, pokud SENDPORT je nastaveno na zapnuto. Další informace najdete v popisu podpříkazu SENDPORT (Určit, zda se má odeslat podpříkaz PORT).
4. FTP servery, které nepodporují PASV neodpovídají standardu RFC 1123.

Omezení
<p>Jste-li připojeni k FTP serveru přes SOCKS server, můžete použít podpříkaz SENDPASV pouze předtím, než vydáte jakýkoli podpříkaz pro přenos dat nebo podpříkaz pro výpis adresářů. Pokud zadáte SENDPASV až po některém z těchto podpříkazů, klient nebude schopen vytvořit datové připojení k FTP serveru.</p> <p>Pokud již klient vydal podpříkaz pro přenos dat nebo výpis adresářů, uzavřete nejprve spojení s FTP serverem přes SOCKS server, a teprve potom můžete znovu vydat podpříkaz SENDPASV.</p> <p>Podpříkaz SENDPASV můžete použít, když je FTP klient odpojen od FTP serveru.</p>

**Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

**SENDPORT (Určit, zda se má odeslat podpříkaz PORT)**

**Podpříkaz FTP klienta**

Chcete-li určit, zda se má či nemá odeslat podpříkaz PORT na FTP server během přenosu dat nebo při zadávání podpříkazů DIR a LS, zadejte podpříkaz FTP klienta SENDPORT:

SENDPOrt [ 0   1 ]
--------------------

Pokud nezadáte žádný parametr, funguje SENDPORT jako přepínač. Hodnota SENDPORT se přepíná z 1 (zapnuto) na 0 (vypnuto) nebo z 0 na 1.

Pokud zadáváte i parametr, platné hodnoty jsou:

**0**

Podpříkaz PORT se neodesílá.

**1**

Podpříkaz PORT se odesílá. Je to předvolená hodnota.

**Poznámky:**

1. Podpříkaz SENDPORT použijte pouze tehdy, když nemůžete navázat spojení se serverem bez něho. Neuvážené používání podpříkazu SENDPORT může vést k chybám.
2. Můžete zjistit, že je vhodné neodesílat podpříkaz PORT těm systémům, které podpříkazy PORT ignorují, protože indikují, že příkaz přijaly.
3. Server neodesílá podpříkaz PORT, je-li podpříkaz SENDPASV nastaven na "zapnuto". Další informace najdete v popisu příkazu SENDPASV (Určit, zda se má odeslat podpříkaz PASV).

**Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## **SENDSITE (Určit, zda se má odeslat podpříkaz SITE)**

### **Podpříkaz FTP klienta**

Chcete-li určit, zda se má či nemá během provádění operace PUT nebo MPUT automaticky odeslat podpříkaz SITE, obsahující informaci o formátu záznamu, zadejte podpříkaz FTP klienta SENDSITE v tomto formátu:

SENDSite [ 0   1 ]
--------------------

Pokud nezadáte žádný parametr, funguje SENDSITE jako přepínač. Hodnota SENDSITE se přepíná z 1 (zapnuto) na 0 (vypnuto) nebo z 0 na 1.

Pokud zadáváte i parametr, platné hodnoty jsou:

**0**

Podpříkaz SITE se odesílá. Je to předvolená hodnota.

**1**

Podpříkaz SITE (obsahující informaci o formátu záznamu) se odesílá dříve než podpříkazy PUT a MPUT. Toto nastavení použijte tehdy, když přenášíte soubory na IBM<sup>(R)</sup> VM (Virtual Machine) server, který používá informaci o formátu záznamu, která se posílá s podpříkazem SITE.

**Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## **SITE (Odeslat informace používané vzdáleným systémem)**

### **Podpříkaz FTP klienta**

Chcete-li odeslat informace, které použije vzdálený systém k poskytování služeb pro něj specifických, použijte podpříkaz FTP klienta SITE v tomto formátu:

```
Site [parametry]
```

### **parametry**

Závisejí na vzdáleném systému.

Abyste zjistili povahu těchto parametrů a jejich syntaxi, zadejte podpříkaz HELP SERVER SITE. Některé FTP servery podpříkaz SITE nepodporují .

#### **Poznámka:**

Podpříkaz SITE se používá s podpříkazy PUT a MPUT a označuje formát a délku záznamů. Předvolba je, že podpříkaz PUT odesílá podpříkaz SITE automaticky. Podpříkaz NAMEFMT používá podpříkaz SITE, aby sdělil serveru, zda jsou jména ve formátu NAMEFMT 0 nebo NAMEFMT 1.

Další informace najdete v popisu příkazu SENDSITE (Určit, zda se má odeslat podpříkaz SITE).

#### **Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## **STATUS (Získat informace o stavu ze vzdáleného systému)**

### **Podpříkaz FTP klienta**

```
STAtus [jméno]
```

### **jméno**

Jméno vzdáleného adresáře nebo souboru, u něhož požadujete informace o stavu. Tento parametr není povinný.

**Poznámka:** Aplikace FTP serveru iSeries<sup>(TM)</sup> parametr jméno nepodporuje.

Pokud nezadáte žádný parametr, server vrátí všeobecné informace o stavu FTP serveru. Tyto informace zahrnují aktuální hodnoty všech parametrů pro přenos a stav připojení. Vrácená informace o stavu závisí na konkrétní implementaci serveru.

#### **Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## **STRUCT (Zadat členění souboru)**

### **Podpříkaz FTP klienta**

Chcete-li zadat členění dat posílaných do souboru, použijte podpříkaz FTP klienta STRUCT v tomto formátu:

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

**F** Souborové členění. Soubor se přenáší jako souvislá řada datových bajtů.



**R** Záznamové členění. Soubor se přenáší jako řada po sobě jdoucích záznamů.

Členění souboru ovlivňuje režim přenosu i interpretaci a uložení souboru.

**Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## **SUNIQUE (Řídit přepisování souborů)**

### **Podpříkaz FTP klienta**

Chcete-li řídit, zda se při provádění podpříkazů PUT nebo MPUT soubor přepíše, zadejte podpříkaz FTP klienta SUNIQUE:

SUnique [ 0 | 1 ]

Pokud ne zadáte žádný parametr, funguje SUNIQUE jako přepínač. Hodnota SUNIQUE se přepíná z 0 (vypnuto) na 1 (zapnuto) nebo z 1 na 0.

Pokud zadáváte i parametr, platné hodnoty jsou:

**0**

Přepíše soubor, pokud existuje. Je to předvolená hodnota.

**1**

Namísto přepsání existujícího souboru vytvoří ve vzdáleném systému nový soubor, který bude mít jedinečné jméno. FTP server odešle jméno vytvořeného souboru ze vzdáleného systému zpět uživateli.

**Poznámka:**

Je-li vzdáleným systémem server iSeries<sup>(TM)</sup>, server tvoří jména File.Mbr tak, že přidá číslo na konec jména *lokálního souboru*, které bylo zadáno v podpříkazu PUT nebo MPUT. To znamená, že pokud již ve vzdáleném systému existuje soubor *NEWFILE.NEWMBR*, vzdálený systém iSeries vytvoří soubor *NEWFILE.NEWMBR1*, do něhož zapíše data.

Jména souborů pro jiné systémy souborů, jako např. HFS, jsou vytvářena na stejném principu. Pokud jméno již existuje, vytvoří se nový soubor, jehož jméno se skládá ze zadaného jména souboru a číselné přípony. To znamená, že pokud již ve vzdáleném systému existuje soubor *xfsname*, vzdálený systém iSeries vytvoří soubor *xfsname1*, do něhož zapíše data.

**Jděte na:**

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## **SYSCMD (Předat CL příkaz iSeries<sup>(TM)</sup> lokálnímu systému iSeries)**

### **Podpříkaz FTP klienta**

Chcete-li spustit CL příkaz na svém lokálním serveru iSeries, aniž byste opustili prostředí FTP, použijte podpříkaz FTP klienta SYSCMD v tomto formátu:

SYSCmd *příkazovýřádek*

### příkazovýřádek

CL příkaz iSeries. Před příkazem můžete uvést ?, abyste získali náznak pro CL příkaz. Zadáte-li například:

```
SYSCMD ? SNDBRKMSG
```

objeví se obrazovka příkazu SNDBRKMSG (Odeslání přerušující zprávy).

Chcete-li si vidět zprávy nízké úrovně, které jsou výsledkem vašeho CL příkazu, nebo chcete-li zadat více CL příkazů, než se vrátíte do prostředí FTP, použijte iSeries příkaz CALL QCMD.

Chcete-li se například dostat na obrazovku systému iSeries Zadání příkazů, zadejte:

```
SYSCMD CALL QCMD
```

Z obrazovky Zadání příkazů můžete pak volat svoje aplikační programy nebo zadávat CL příkazy. Po dokončení vašeho aplikačního programu nebo CL příkazu se vrátíte na obrazovku Zadání příkazů. Odtud můžete prohlížet zprávy, spouštět další práci v systému, nebo se stisknutím klávesy (Konec) nebo klávesy F12 (Zrušení) vrátit do FTP.

CL příkaz systému iSeries můžete také zadat po stisknutí klávesy F21 (příkazová řádka CL) na hlavní obrazovce FTP. Server nepovolí klávesu F21, jestliže je k výstupnímu bodu Ověření platnosti požadavku na FTP klientu přiřazen program.

#### Poznámky:

1. Většina serverů má nastavenou časovou prodlevu, která ukončí relaci, pokud se během této doby neprovede žádná činnost. Pokud probíhá zpracování příkazu déle, než je nastavená doba časové prodlevy, server ukončí spojení s klientem.
2. Server iSeries podporuje vykřičník (!) jako synonymum k podpříkazu SYSCMD.
3. Podpříkaz SYSCMD předá systému iSeries přesně to, co uživatel zadal, jako CL příkaz.

#### Jděte na:

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## TYPE (Zadat typ přenosu souboru)

### Podpříkaz FTP klienta

Chcete-li zadat typ přenosu souborů nebo reprezentace dat pro přenos souborů, použijte podpříkaz FTP klienta TYPE v tomto formátu:

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

#### A

Nastaví typ přenosu jako předvolený (ASCII) typ přenosu. Má stejný účinek, jako podpříkaz ASCII. Server k souboru nepřirazuje řízení vertikálního formátu. Pro ASCII podporuje pouze předvolený formát NON PRINT. Používejte vždy typ přenosu ASCII nebo přenos textových souborů, kromě případů, kdy oba systémy používají reprezentaci dat EBCDIC.

Předvolený CCSID pro TYPE A (ASCII) je identifikátor kódové sady znaků, který je uveden v parametru CCSID příkazu STRTCPFTP nebo podpříkazu FTP.

## **B**

Shift JIS Kanji (CCSID 932)

### **B 1**

Shift JIS Kanji (CCSID 932)

### **B 2**

Kanji - rozšířený UNIX kód (CCSID 5050)

### **B 3**

JIS 1983 používající ASCII shift-in escape sekvenci (CCSID 5054)

### **B 3 A**

JIS 1983 používající ASCII shift-in escape sekvenci (CCSID 5054)

### **B 3 R**

JIS 1983 používající JISROMAN shift-in escape sekvenci (CCSID 5052)

### **B 4**

JIS 1978 používající ASCII shift-in escape sekvenci (CCSID 5055)

### **B 4 A**

JIS 1978 používající ASCII shift-in escape sekvenci (CCSID 5055)

### **B 4 R**

JIS 1978 používající JISROMAN shift-in escape sekvenci (CCSID 5053)

## **B 5**

Hangeul (CCSID 934)

## **B 6**

Korejština - standardní kód KSC-5601, verze 1989 (CCSID 949)

## **B 7**

Tradiční čínština (5550) (CCSID 938)

## **C ccsid#**

Nastaví typ přenosu na jakýkoli CCSID (identifikátor kódové sady znaků), který je v systému nainstalován. Po písmenu C následuje číslo CCSID.

## **E**

Nastaví typ přenosu na EBCDIC. Má stejný účinek jako podpříkaz EBCDIC. Server k souboru nepřirazuje řízení vertikálního formátu. Pro EBCDIC podporuje pouze předvolený formát NON PRINT. Typ přenosu EBCDIC je určen pro výkonný přenos dat mezi systémy, které používají EBCDIC jako svou interní reprezentaci dat.

## **F**

IBM EBCDIC Kanji (CCSID 5035)

### **F 1**

IBM EBCDIC Kanji (CCSID 5035)

## I

Nastaví typ přenosu na obraz. Má stejný účinek jako podpříkaz BINARY. Je-li zadán typ přenosu obraz, data jsou reprezentována jako řetězec bitů, pakovaný do bajtů po 8 bitech. Typ přenosu obraz je účinný při ukládání a načítání souborů a při přenosu binárních dat, jako např. kódu objektu. Data jsou přenášena tak, jak jsou; neprovádí se žádná konverze.

Pokud nezadáte žádné parametry, server zobrazí aktuální nastavení pro podpříkaz TYPE.

### Jděte na:

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## USER (Odeslat vaše uživatelské ID do vzdáleného systému)

### Podpříkaz FTP klienta

User *uživatelskéid* [*heslo*]

#### **uživatelskéid**

Vaše přihlašovací jméno ve vzdáleném systému.

#### **heslo**

Vaše heslo ve vzdáleném systému. Zadání hesla je volitelné. Pokud nezadáte heslo během volání podpříkazu USER, budete vyzváni k zadání hesla v případě, že vzdálený systém přihlašovací heslo vyžaduje.

### Jděte na:

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## VERBOSE (Řídit textové zobrazení chybových zpráv)

### Podpříkaz FTP klienta

Chcete-li řídit zobrazování odpovědí FTP serveru, použijte podpříkaz FTP klienta VERBOSE. Podpříkaz VERBOSE přepíná režim detailního výpisu mezi "zapnuto" a "vypnuto". Je-li režim detailního výpisu "zapnut", zobrazují se všechny odpovědi serveru, včetně kódů odpovědí. Je-li "vypnutý", některé odpovědi serveru a kódy odpovědí jsou vyřazeny a nezobrazí se.

Verbose

### Jděte na:

- Podpříkazy FTP klienta.
- Konvence syntaxe FTP klienta.

## Programy výstupních bodů FTP



FTP klient i FTP server komunikují s každým takovýmto programem prostřednictvím určitého výstupního bodu. Mezi serverem a programem výstupního bodu se předávají parametry. Formát předávaných informací je určen formátem výstupního bodu.

FTP používá následující výstupní body. V těchto tématech najdete další informace včetně popisu parametrů a ukázek kódu:

- Výstupní bod Ověření platnosti požadavku: klient a server.

- Výstupní bod Přihlášení k serveru.

Aby vaše programy výstupních bodů správně fungovaly, je třeba provést Instalaci a registraci programů výstupních bodů. Jestliže tyto programy již nebudete potřebovat, je třeba zabránit jejich opětovnému použití tak, že provedete Odstranění programů výstupních bodů.

### Výstupní body a formáty výstupních bodů u TCP/IP

V následující tabulce je uveden přehled výstupních bodů pro různé TCP/IP aplikace a příslušné formáty výstupních bodů.

Výstupní body TCP/IP	Aplikace	VLRQ0100	TCPL0100	TCPL0200	TCPL0300
QIBM_QTMF_CLIENT_REQ	FTP	X			
QIBM_QTMF_SERVER_REQ	FTP	X			
QIBM_QTMF_SVR_LOGON <sup>1</sup>	FTP		X	X	X <sup>2</sup>
QIBM_QTMX_SERVER_REQ	REXEC	X			
QIBM_QTMX_SVR_LOGON <sup>1</sup>	REXEC		X		X <sup>2</sup>
QIBM_QTOD_SERVER_REQ	TFTP	X			

<sup>1</sup> -Výstupní bod může mít více než jeden formát, ale program může být registrován pouze pro jeden formát výstupního bodu. Vyzkoušejte si všechny tyto formáty a potom si vyberte ten, který nejvíce odpovídá vašemu systému.

<sup>2</sup> - Tento formát je k dispozici od verze V5R1.



### Výstupní bod Ověření platnosti požadavku: klient a server

Výstupní body Ověření platnosti požadavku slouží k omezení činností, které mohou uživatelé FTP vykonávat. Výstupní body Ověření platnosti požadavku obsahuje jak FTP server, tak FTP klient. Chcete-li tedy omezit přístup k FTP serveru i FTP klientu, musíte k oběma těmto výstupním bodům připojit programy.

#### Poznámka:

Protože výstupní body FTP klienta i serveru mají tentýž formát, stačí pro oba tyto výstupní body napsat jeden program.

Jestliže implementujete anonymní FTP, napište svůj program výstupního bodu Ověření platnosti požadavku na FTP serveru tak, abyste omezili anonymní FTP uživatele pouze na podpříkazy FTP pro načítání. Nikdy nepovolujte anonymním uživatelům spouštět CL příkazy.

#### Co by měl váš program ještě obsahovat:

- Ošetření výjimek.
- Ladění.
- Protokolování.

#### Povolení a zamítnutí příkazů

Program výstupního bodu Ověření platnosti požadavku vám umožňuje řídit, zda danou operaci povolit nebo zamítnout. Rozhodnutí, která provedou programy výstupních bodů, jsou prováděna navíc k ověření prováděným aplikací FTP serveru. Aplikace FTP serveru nebo klienta volá program výstupního bodu, který je pro tuto aplikaci registrován, pokaždé, když zpracovává některý z těchto požadavků:

1. vytvoření adresáře/knihovny

2. vymazání adresáře/knihovny
3. nastavení aktuálního adresáře
4. výpis jmen souborů
5. vymazání souboru
6. odeslání souboru
7. získání souboru
8. přejmenování souboru
9. provedení CL příkazu

Jestliže chcete příkaz vždy a bezpodmínečně zamítnout, nastavte parametr 8 (Allow operation - Povolit operaci) ve formátu výstupního bodu VRLQ0100 na hodnotu -1.

### Existuje u programu výstupního bodu nějaká časová prodleva?

U programů výstupních bodů FTP se nepoužívá žádná časová prodleva. Pokud je v programu výstupního bodu chyba nebo výjimka, kterou nelze zpracovat, FTP server relaci předčasně ukončí.

### Příklady programů

Příklady programů, které jsou k dispozici, vám pomohou nastavit na serveru anonymní FTP. Tyto příklady slouží pouze pro ilustraci. Neobsahují dostatek funkcí na to, aby mohly být používány v provozním systému v tom stavu, v jakém jsou. Můžete z nich vycházet při vytváření vlastních programů. Okopírované části kódu ze vzorových programů můžete přidat do programů, které jste sami napsali. Doporučujeme, abyste vzorové programy spouštěli na jiném než provozním systému.

Příklad: Program výstupního bodu Ověření platnosti požadavku na FTP klientu nebo FTP serveru v jazyce CL

Příklad: Program výstupního bodu Ověření platnosti požadavku na FTP serveru v jazyce ILE RPG

### Prohlášení k příkladům kódu

IBM vám uděluje nevýhradní copyright licenci k používání všech ukázek programového kódu, ze kterých můžete generovat podobné funkce přizpůsobené svým vlastním potřebám.

Všechny ukázky kódu dodané IBM slouží pouze pro ilustrativní účely. Tyto ukázky nebyly náležitě otestovány pro všechny podmínky. Proto IBM nemůže zaručit ani naznačit spolehlivost, provozuschopnost ani funkčnost těchto programů.

Veškeré programy zde obsažené jsou vám k dispozici tak, "jak jsou", bez jakýchkoli záruk. Výslovně se popírají jakékoli naznačené záruky týkající se zákonnosti, obchodovatelnosti a vhodnosti ke konkrétním účelům.

**Příklad: Program výstupního bodu Ověření platnosti požadavku na FTP klientu nebo FTP serveru v jazyce CL:** Toto je příklad jednoduchého programu pro výstupní bod Ověření platnosti požadavku FTP. Je napsán v příkazovém jazyce iSeries<sup>TM</sup> CL (Command Language). Tento kód není úplný, může však být výchozím bodem pro vytváření vašich vlastních programů pro výstupní body serveru nebo klienta.

**Poznámka:** Důležité informace o souvisejících právech si můžete přečíst v Prohlášení k příkladům programového kódu.

(Formátovaný text v následujícím příkladu bude přesahovat mimo rámeček.)

```

/*****/
/*                                          */
/*  Sample FTP server request validation exit program for anonymous FTP.  */
/*  Note: This program is a sample only and has NOT undergone any formal */
/*        review or testing.                                          */
/*                                          */
/*****/

```

```

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

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

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

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

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

/* Declare request control block for QLGCNVCS (convert case) API: */
/* convert to uppercase based on job CCSID */
DCL          VAR(&CASEREQ) TYPE(*CHAR) LEN(22) +
                          VALUE(X'00000001000000000000000000000000+
                          00000000')
DCL          VAR(&ERROR)   TYPE(*CHAR) LEN(4) +
                          VALUE(X'00000000')

/* Assign input parameters to local copies */
CHGVAR      VAR(&APPID)  VALUE(%BINARY(&APPIDIN))
CHGVAR      VAR(&OPID)   VALUE(%BINARY(&OPIDIN))
CHGVAR      VAR(&IPLLEN) VALUE(%BINARY(&IPLLENIN))
CHGVAR      VAR(&IPADDR) VALUE(%SUBSTRING(&IPADDRIN 1 &IPLLEN))
CHGVAR      VAR(&OPLLEN) VALUE(%BINARY(&OPLLENIN))

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

/* Operation id 0 (incoming connection): reject if connection is coming */
/* through interface 9.8.7.6, accept otherwise. (The address is just an */
/* example.) This capability could be used to only allow incoming connections */
/* from an internal network and reject them from the "real" Internet, if */

```

```

/* the connection to the Internet were through a separate IP interface. */
/* NOTE: For FTP server, operation 0 is ALWAYS under QTCP profile. */
    IF      COND(&OPID = 0) THEN(DO)
    IF      COND(&OPINFO = '9.8.7.6') THEN(CHGVAR +
        VAR(%BINARY(&ALLOWOP)) VALUE(&NOALLOW))
    ELSE    CMD(CHGVAR VAR(%BINARY(&ALLOWOP)) +
        VALUE(&ALLOW))
    GOTO    CMDLBL(END)
ENDDO

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

END:      ENDPGM

```

**Příklad: Program výstupního bodu Ověření platnosti požadavku na FTP serveru v jazyce ILE RPG:**

Toto je příklad jednoduchého programu pro výstupní bod Ověření platnosti požadavku na FTP serveru. Je napsán v programovacím jazyce ILE RPG. Tento kód není úplný, může však být výchozím bodem pro vytváření vašich vlastních programů.

**Poznámka:** Důležité informace o souvisejících právech si můžete přečíst v Prohlášení k příkladům programového kódu.

(Formátovaný text v následujícím příkladu bude přesahovat mimo rámeček.)

```

* 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
*                                     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('abcdefghijklmnopqrstuvwxyz')
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
D APPIDds        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
*

```

a \*

```

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
C    Library      OREQ    PublicLib
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
C      Library        OREQ          PublicLib
C                      Z-ADD          Allow          ALLOWOP
C                      ELSE
C                      Z-ADD          DontAllow      ALLOWOP
C                      ENDIF
*
C          OTHER
C          Z-ADD      DontAllow      ALLOWOP
C          ENDSL
*
C          ELSE
*
* Any Other User: Allow All FTP Operations
*
C      OPIDIN          IFEQ          6          Send Files to Client
*
* If client issued GET for save file HESSU in library HESSU then we refresh the contents
*
*
C      LW:UP          XLATE          OPINFOIN      OPINFO

```

```

C          Z-ADD      0          i 3 0
C    Savetti  SCAN      OPINFO:1
i *
C    i IFGT      0
*
* We assume that the save file exists and here clear the save file
*
C          MOVE(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(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

```

**Formát výstupního bodu VLRQ0100:** Výstupní bod Ověření platnosti požadavku na FTP serveru je:

QIBM\_QTMF\_SERVER\_REQ

Výstupní bod Ověření platnosti požadavku na FTP klientu je:

QIBM\_QTMF\_CLIENT\_REQ

Rozhraní, které ovládá formát parametrů výstupního bodu, je:

VLRQ0100

Následující tabulka uvádí parametry a jejich formát pro rozhraní VLRQ0100.

**Požadovaný formát parametrů pro rozhraní výstupního bodu VLRQ0100**

Parametr	Popis	Input / Output	Typ a délka
1	Application identifier (Identifikátor aplikace)	Input	Binary (4)
2	Operation identifier (Identifikátor operace)	Input	Binary (4)
3	User profile (Uživatelský profil)	Input	Char (10)
4	Remote IP address (Vzdálená IP adresa)	Input	Char (*)
5	Length of remote IP address (Délka vzdálené IP adresy)	Input	Binary (4)
6	Operation-specific information (Informace specifická pro operaci)	Input	Char (*)

Parametr	Popis	Input / Output	Typ a délka
7	Length of operation-specific information (Délka informace specifické pro operaci)	Input	Binary (4)
8	Allow operation (Povolit operaci)	Output	Binary (4)

Následuje popis parametrů:

**VLRQ0100 - parametr 1:**

Application identifier (Identifikátor aplikace).

**INPUT; BINARY(4)**

Identifikuje aplikační program TCP/IP, který zadává požadavek. Rozhraní VLRQ0100 mohou sdílet čtyři různé aplikace TCP/IP. První parametr určuje, která z těchto aplikací volá program výstupního bodu. Možné hodnoty jsou:

0	program FTP klienta
1	program FTP serveru
2	program REXEC serveru
3	program TFTP serveru

**VLRQ0100 - parametr 2:**

Operation identifier (Identifikátor operace)

**Input; Binary(4)**

Identifikuje operaci (příkaz), který chce (požaduje) uživatel FTP provést.

Jestliže identifikátor aplikace (parametr 1) uvádí program FTP klienta nebo FTP serveru, možné hodnoty identifikátoru operace jsou:

Hodnota	ID operace	Podpříkaz klienta	Podpříkaz serveru
0	Start session (Spuštění relace)	Open, SECOpen	Nové připojení
1	Create directory/library (Vytvoření adresáře/knihovny)	*	MKD, XMDK
2	Delete directory/library (Vymazání adresáře/knihovny)	*	RMD, XRMD
3	Set current directory/library (Nastavení aktuálního adresáře/knihovny)	LCD	CWD, CDUP, XCWD, XCUP
4	List files (Výpis seznamu souborů)	*	LIST, NLIST
5	Delete file (Vymazání souboru)	*	DELE
6	Send file (Odeslání souboru)	APPEND, PUT, MPUT	RETR

Hodnota	ID operace	Podpříkaz klienta	Podpříkaz serveru
7	Receive file (Získání souboru)	GET, MGET	APPE, STOR, STOU
8	Rename file (Přejmenování souboru)	*	RNFR, RNT0
9	Execute CL command (Provedení CL příkazu)	SYSCMD	RCMD, ADDm, ADDV, CRTL, CRTP, CRTS, DLTF, DLTl

**Poznámka:**

Symbol \* představuje řídicí operace, které výstupní bod FTP klienta nerozpoznává. Tyto operace může klient provádět pouze pomocí CL příkazů zadaných v podpříkazu FTP klienta SYSCMD. Provádění CL příkazů řídí identifikátor operace 9.

**VLRQ0100 - parametr 3:**

User profile (Uživatelský profil)

**INPUT; CHAR(10)**

Uživatelský profil pro relaci FTP.

**VLRQ0100 - parametr 4:**

Remote IP address (Vzdálená IP adresa)

**INPUT; CHAR(\*)**

IP (Internet Protocol) adresa vzdáleného hostitelského systému. Formát tohoto řetězce je tečková konvence (123.45.67.89) se zarovnáním vlevo. Vzdáleným hostitelem může být klient nebo server, podle hodnoty identifikátoru aplikace (parametr 1).

**VLRQ0100 - parametr 5:**

Length of the remote IP address (Délka vzdálené IP adresy)

**INPUT; BINARY(4)**

Délka (v bajtech) IP adresy vzdáleného systému (parametr 4).

**VLRQ0100 - parametr 6:**

Operation-specific information (Informace specifická pro operaci)

**INPUT; CHAR(\*)**

Informace, která popisuje požadovanou operaci. Obsah tohoto pole závisí na hodnotě identifikátoru operace (parametr 2) a identifikátoru aplikace (parametr 1). Například:

**Pro identifikátor operace 0 a identifikátor aplikace 0:**

K požadované operaci neexistuje žádná informace. Toto pole je prázdné.

**Pro identifikátor operace 0 a identifikátor aplikace 1:**

Informace specifická pro operaci obsahuje IP adresu rozhraní TCP/IP, které se při této relaci připojí k lokálnímu hostiteli (FTP serveru). Formát tohoto řetězce je tečková konvence (123.45.67.89) se zarovnáním vlevo.

**Pro identifikátory operace 1 až 3:**

Informace specifická pro operaci obsahuje jméno adresáře nebo knihovny, v nichž se tato operace provede. Formát jména adresáře nebo knihovny je jméno jejich absolutní cesty.

**Pro identifikátory operace 4 až 8:**

Informace specifická pro operaci obsahuje jméno souboru, se kterým se operace provede. Formát jména souboru je jméno absolutní cesty.

**Pro identifikátor operace 9:**

Informace specifická pro operaci obsahuje příkaz CL (iSeries<sup>TM</sup> Control Language), který uživatel požaduje.

Můžete se také podívat na Poznámky k použití formátu výstupního bodu VLRQ0100.

**VLRQ0100 - parametr 7:**

Length of operation-specific information (Délka informace specifické pro operaci)

**INPUT; BINARY(4)**

Udává délku informace specifické pro operaci (parametr 6). Když výstupní bod nepodává informaci specifickou pro operaci, je hodnota tohoto parametru 0.

**VLRQ0100 - parametr 8:**

Allow operation (Povolit operaci)

**OUTPUT; BINARY(4)**

Určuje, zda se má požadovaná operace povolit nebo zamítnout. Možné hodnoty jsou:

-1	<p><i>Nikdy</i> nepovolit tento identifikátor operace:</p> <p>Tento identifikátor operace bude bezvýhradně zamítán po celý zbytek aktuální relace.</p> <p>Tento identifikátor operace nebude již znovu volat program výstupního bodu.</p>
0	Zamítnout operaci.
1	Povolit operaci.
2	<p><i>Vždy</i> povolit tento identifikátor operace:</p> <p>Tento identifikátor operace bude bezvýhradně povolován po celý zbytek aktuální relace.</p> <p>Tento identifikátor operace nebude již znovu volat program výstupního bodu.</p>

*Poznámky k použití formátu výstupního bodu VLRQ0100:* VLRQ0100 je formát výstupního bodu, který se používá jak pro výstupní bod Ověření platnosti požadavku na FTP klientu, tak i pro výstupní bod Ověření platnosti požadavku na FTP serveru.

**Neplatný výstup parametrů**

Jestliže výstup vrácený pro parametr 8 (Povolit operaci) není platný, FTP server zamítne požadovanou operaci a do protokolu úlohy zapíše tuto zprávu:

*Data from exit program for exit point &1 is missing or not valid. (Data z programu pro výstupní bod &1 chybějí nebo nejsou platná.)*

**Výjimky**

Jestliže FTP server narazí při volání programu výstupního bodu na výjimku, zapíše do protokolu úlohy tuto zprávu:



*Exception encountered for FTP exit program &1 in library &2 for exit point &3. (Byla zjištěna výjimka pro program FTP &1 v knihovně &2 pro výstupní bod &3.)*

### Přehled: Informace specifická pro operaci

Tato tabulka podává přehled o parametru Informace specifická pro operaci (VLRQ0100 - parametr 6), který je povinný pro každý Identifikátor operace (VLRQ0100 - parametr 2).

Identifikátor operace (VLRQ0100 - parm 2)	Informace specifická pro operaci (VLRQ0100 - parm 6)
0	Žádná, jestliže ID aplikace = 0 (parametr 1).
0	IP adresa klientského hostitele ve formátu tečkové konvence, jestliže ID aplikace = 1 nebo 2 (parametr 1).
1-3	Absolutní jméno cesty knihovny nebo adresáře. Příklady: /QSYS.LIB/QGPL.LIB <sup>(a)</sup> /QOpenSys/DirA/DirAB/DirABC <sup>(b)</sup>
4-8	Absolutní jméno cesty souboru. Příklady: /QSYS.LIB/MYLIB.LIB/MYFILE.FILE/MYMEMB.MBR <sup>(a)</sup> /QOpenSys/DirA/DirAB/DirABC/FileA1 <sup>(b)</sup>

#### Poznámky:

<sup>(a)</sup> - Jména cest v systému souborů QSYS.LIB se vždy uvádějí velkými písmeny.

<sup>(b)</sup> - Jména cest v systému souborů QOpenSys rozlišují velká a malá písmena a mohou obsahovat oba typy písmen.

### Výstupní bod Přihlášení k serveru

Pomocí výstupního bodu Přihlášení k aplikačnímu serveru TCP/IP můžete řídit autentizaci uživatelů na aplikačním serveru TCP/IP. Tento výstupní bod umožňuje přístup k serveru na základě adresy, z níž relace vychází. Také umožňuje zadat výchozí pracovní adresář, který je odlišný od adresáře uvedeného v uživatelském profilu.

Když přiřadíte tomuto výstupnímu bodu svůj program, bude server volat tento program výstupního bodu pro přihlášení při každém pokusu uživatele o přihlášení. Program výstupního bodu nastaví do výstupního parametru návratový kód, kterým určí, zda server bude či nebude pokračovat v operaci přihlášení. Další nastavení návratového kódu řídí zpracování přihlášení a informace o výchozím adresáři.

Výstupní bod Přihlášení k FTP serveru iSeries<sup>(TM)</sup> je:

QIBM\_QTMF\_SVR\_LOGON

Tento výstupní bod může mít tři formáty:

- Formát výstupního bodu TCPL0100 umožňuje tyto základní operace řízení přihlášení:
  - Možnost potvrdit nebo zamítnout přihlášení.
  - Možnost řídit uživatelský profil, heslo a aktuální knihovnu.
- Formát výstupního bodu TCPL0200 poskytuje další parametry pro řízení procesu přihlášení. Patří sem:
  - Možnost nastavit pracovní adresář na jakýkoliv adresář v systému.
  - Možnost vrátit informace specifické pro aplikaci.
  - Možnost řídit šifrování odesílaných a přijímaných dat u FTP klienta.

- Formát výstupního bodu TCPL0300 rozšiřuje formát TCPL0200, aby byla umožněna podpora rozšířeného hesla OS/400<sup>(R)</sup>, a nabízí další parametry, které umožňují zpracování CCSID v poli pro heslo a jméno adresáře. Navíc, je-li uživatel pro danou relaci autentizován certifikátem klienta, je tento certifikát klienta předán programu výstupního bodu.

#### Poznámky:

1. Pro výstupní bod Přihlášení k FTP serveru může být registrován pouze jeden program. Musíte se rozhodnout, který ze tří formátů výstupního bodu budete používat.
2. Tento výstupní bod umožňuje FTP aplikacím implementaci Anonymního FTP včetně informací nutných k protokolování a řízení přístupu.
3. Pro všechny znakové parametry ve formátech výstupního bodu TCPL0100 a TCPL0200, a pro všechny znakové parametry bez přiřazeného CCSID ve formátu výstupního bodu TCPL0200 platí, že znaková data předaná programu výstupního bodu mají CCSID stejný jako daná úloha. Jestliže je CCSID úlohy 65535, potom mají znaková data předvolený CCSID této úlohy. U veškerých znakových dat, která jsou vrácena programem výstupního bodu v těchto parametrech, se předpokládá stejný CCSID.

#### Pro anonymní FTP vytvořte program výstupního bodu Přihlášení k FTP serveru, který:

1. Potvrdí přihlášení uživatelského ID ANONYMOUS.
2. Požaduje jako heslo adresu elektronické pošty. Je zvykem požadovat 'platnou adresu elektronické pošty' místo hesla. Tento termín je zavádějící, neboť program výstupního bodu ověřuje pouze, je-li uprostřed řetězce alfanumerických znaků znak '@'. To je také obvyklé. Proto je velmi důležité zaprotokolovat IP adresu uživatele.
3. Zkontroluje znak @ v řetězci hesla.
4. Povolí uživateli ANONYMOUS přístup pouze ke knihovně s veřejným přístupem. Bližší informace najdete u návratového kódu 3 parametru 8 pro Formát TCPL0200.

#### Co by měl váš program ještě obsahovat:

- Ošetření výjimek.
- Ladění.
- Protokolování
  - zaznamenává se IP adresa a adresa elektronické pošty (posílaná jako heslo) žadatele služeb FTP.

#### Existuje u programu výstupního bodu nějaká časová prodleva?

U programů výstupních bodů FTP se nepoužívá žádná časová prodleva. Pokud je v programu výstupního bodu chyba nebo výjimek, kterou nelze zpracovat, FTP server relaci předčasně ukončí.

#### QTCP potřebuje oprávnění

Když aplikace zavolá program výstupního bodu Přihlášení k FTP serveru, úloha FTP serveru se zpracovává pod uživatelským profilem QTCP.

Ujistěte se, že profil QTCP má dostatečná oprávnění k přístupu a zápisu ke všem souborům protokolu nebo satelitním souborům, které jsou s programem výstupního bodu spojeny.

#### Příklady programů

Příklady programů, které jsou k dispozici, vám pomohou nastavit na serveru anonymní FTP. Tyto příklady slouží pouze pro ilustraci. Neobsahují dostatek funkcí na to, aby mohly být používány v provozním systému v tom stavu, v jakém jsou. Můžete z nich vycházet při vytváření vlastních programů. Okopírované části

kódu ze vzorových programů můžete přidat do programů, které jste sami napsali. Doporučujeme, abyste vzorové programy spouštěli na jiném než provozním systému.

Příklad: Program výstupního bodu Přihlášení k FTP serveru v jazyce CL.

Příklad: Program výstupního bodu Přihlášení k FTP serveru v jazyce C.

Příklad: Program výstupního bodu Přihlášení k FTP serveru v jazyce ILE RPG.

## Prohlášení k příkladům kódu

IBM vám uděluje nevýhradní copyright licenci k používání všech ukázek programového kódu, ze kterých můžete generovat podobné funkce přizpůsobené svým vlastním potřebám.

Všechny ukázky kódu dodané IBM slouží pouze pro ilustrativní účely. Tyto ukázky nebyly náležitě otestovány pro všechny podmínky. Proto IBM nemůže zaručit ani naznačit spolehlivost, provozuschopnost ani funkčnost těchto programů.

Veškeré programy zde obsažené jsou vám k dispozici tak, "jak jsou", bez jakýchkoli záruk. Výslovně se popírají jakékoli naznačené záruky týkající se zákonnosti, obchodovatelnosti a vhodnosti ke konkrétním účelům.

**Příklad: Program výstupního bodu Přihlášení k FTP serveru v jazyce CL:** Toto je příklad jednoduchého programu pro výstupní bod Přihlášení k FTP serveru. Je napsán v příkazovém jazyce iSeries<sup>(TM)</sup> CL (Command Language). Tento kód není úplný, může však být výchozím bodem pro vytváření vašich vlastních programů.

**Poznámka:** Důležité informace o souvisejících právech si můžete přečíst v Prohlášení k příkladům programového kódu.

(Formátovaný text v následujícím příkladu bude přesahovat mimo rámeček.)

```
/* **** */
/*
/* 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(&USRLLEN)   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(&USRLLEN) VALUE(%BINARY(&USRLLENIN))
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(&USRLLEN = 9) THEN(DO)
  IF      COND(%SST(&USRIN 1 9) = 'ANONYMOUS') THEN(DO)
/* For anonymous user: want to force user profile ANONYMOUS current library to PUBLIC. */
  CHGVAR VAR(%BINARY(&RETCDOU)) VALUE(6)
  CHGVAR VAR(&USRPRFOUT) VALUE('ANONYMOUS ')
  CHGVAR VAR(&CURLIBOUT) VALUE('PUBLIC  ')
  ENDDO
/* Any other user: proceed with normal logon processing. */
  ELSE      CMD(CHGVAR VAR(%BINARY(&RETCDOU)) VALUE(1))
  ENDDO
  ELSE      CMD(CHGVAR VAR(%BINARY(&RETCDOU)) VALUE(1))

END:      ENDPGM

```

**Příklad: Program výstupního bodu Přihlášení k FTP serveru v jazyce C:** Toto je příklad jednoduchého programu pro výstupní bod Přihlášení k FTP serveru. Je napsán v programovacím jazyce C. Tento kód není úplný, může však být výchozím bodem pro vytváření vašich vlastních programů.

**Poznámka:** Důležité informace o souvisejících právech si můžete přečíst v Prohlášení k příkladům programového kódu.

(Formátovaný text v následujícím příkladu bude přesahovat mimo rámeček.)

```

/* Module Description *****/
/* *****/
/* Note: This program is a sample only and has NOT undergone any
/ formal review or testing. */
/* *****/
/* Source File Name: qtmfsvrln.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. */
/* qtmfsvrln - FTP Server Logon exit function. */

```

```

/*          CheckClientAddress - Check originating sessions IP */
/*                                address.                        */
/*                                                                */
/* End Module Description *****/
#define _QTMFSVRLGN_C

/*****/
/* All file scoped includes go here                               */
/*****/
#ifndef __stdio_h
#include <stdio.h>
#endif

#ifndef __ctype_h
#include <ctype.h>
#endif

#ifndef __string_h
#include <string.h>
#endif

#ifndef __stdlib_h
#include <stdlib.h>
#endif

#include "qusec.h"          /* Include for API error code structure */
#include "qsyrusri.h"      /* Include for User Information API    */

/*****/
/* All file scoped Constants go here                               */
/*****/
#define EQ      ==
#define NEQ    !=
#define BLANK   ' '
#define FWIDTH 128        /* Width of one database file record */
#define FNAME   21        /* Qualified database file name width */

/* Valid characters for Client IP address. The CheckClientAddress()
/* function will check the Client IP address input argument
/* (ClientIPAddr_p) to ensure it is in valid dotted-decimal format.
/* This is one example of an input validity check.
const char ValidChars[] = "0123456789.";
/*****/
/* All file scoped type declarations go here                       */
/*****/

/*****/
/* All file-scoped macro calls go here                             */
/*****/

/*****/
/* All internal function prototypes go here                        */
/*****/
static void qtmfsvrlgn
    (int,char *,int,char *,int,char *,int,int *,char *,char *,char *);

static int CheckClientAddress(char *, int);

/*****/
/* All file scoped variable declarations go here                   */
/*****/

/*****/
/* ** NOTE **                                                      */
/*****/

```

```

/* The following client IP address are for example purposes only. Any */
/* resemblance to actual system IP addresses is purely coincidental. */
/*****
/* EXCLUSIVE system lists, ie - Logon attempts from any client IP */
/* addresses NOT in one of these lists */
/* are allowed to continue. */
/* Reject server logon attempts of users attempting to log in from */
/* these client systems (return code = 0) */
char Reject[] = "1.2.3.4 5.6.7.8";
/* Limit logon abilities of users attempting to log in as ANONYMOUS */
/* from these client systems (return code = 6). */
/* In this example program, the initial current library is set and */
/* returned as an output parameter for users attempting to log in */
/* as ANONYMOUS from these specific client systems. */
char Limit[] = "9.8.7.6 4.3.2.1 8.7.6.5";

/* Function Specification *****/
/*
/* Function Name: Main */
/*
/* Descriptive Name: FTP Server Logon exit program main. */
/*
/* This example exit program allows access to a TCP/IP server to */
/* be controlled by the address of the originating session, gives */
/* additional control over the initial current library to a user, */
/* and provides the capability to implement "anonymous" FTP. */
/*
/* Notes: */
/*
/* Dependencies: */
/* FTP Server Logon exit point QIBM_QTMF_SVR_LOGON was registered */
/* during FTP product installation. */
/*
/* Restrictions: */
/*
/* None */
/*
/* Messages: */
/*
/* None */
/*
/* Side Effects: */
/*
/* None */
/*
/* Functions/Macros called: */
/*
/* qtmfsvrlgn - Server Logon exit function. */
/*

/* Input: */
/* int * argv[1] - Identifies requesting application */
/* (FTP Client =0, FTP Server = 1). */
/* char * argv[2] - User identifier from client program. */
/* (For FTP server, this is user CMD data */
/* int * argv[3] - Length (in bytes) of User ID string. */
/* char * argv[4] - Authentication string from client. */
/* (For FTP server, this is the password) */
/* int * argv[5] - Length (bytes) Authentication string. */
/* char * argv[6] - Internet Protocol address from which */
/* the session originates. */
/* int * argv[7] - Length (in bytes) of IP address. */
/* int * argv[8] - Return code (received as 0). */
/* char * argv[9] - User profile (received as blanks). */
/* char * argv[10] - Password (received as blanks). */
/* char * argv[11] - Initial current library (received as blanks)*/

```

```

/*
/* Exit Normal: Return Return Code, User Profile, Password, Initial
/* Current Library to server application.
/*
/* Exit Error: None
/*
/* End Function Specification *****/
void main(int argc, char *argv[])
{
/* *****/
/* Code
/* *****/

/* *****/
/* Collect input arguments and call function to determine if client
/* should be allowed to log in to an FTP server application.
/* *****/
qtmfsvr1gn(*(int *) (argv[1])), /* Application Identifier
(Input) */
    argv[2], /* User Identifier (Input) */
    (*(int *) (argv[3])), /* Length User of
Identifier(Input) */
    argv[4], /* Authentication String (Input) */
    (*(int *) (argv[5])), /* Length of Authentication string
(Input) */
    argv[6], /* Client IP Address (Input) */
    (*(int *) (argv[7])), /* Length of Client IP Address
(Input) */
    (int *) (argv[8]), /* Return Code (Output)*/
    argv[9], /* User Profile (Output)*/
    argv[10], /* Password (Output)*/
    argv[11]; /* Initial Current Library (Output)*/
    return;
}

/* Function Specification *****/
/*
/* Function Name: qtmfsvr1gn
/*
/* Descriptive Name: Server Logon exit function.
/*
/* This exit function provides control over user authentication to
/* an FTP server.
/*
/* Notes:
/*
/* Dependencies:
/*
/* FTP Server Logon exit point QIBM_QTMF_SVR_LOGON was
/* registered during FTP product installation.
/*
/* Restrictions:
/*
/* None
/*
/* Messages:
/*
/* None
/*
/* Side Effects:
/*
/* None
/*
/* Functions/Macros called:
/*

```

```

/*      CheckClientAddress - Check the ClientIPAddr_p input argument.*/
/*      memcpy - Copy bytes from source to destination.          */
/*      memset - Set bytes to value.                             */
/*      strstr - Locate first occurrence of substring.           */
/*      sprintf - Formatted print to buffer.                     */
/*                                                              */
/* Input:                                                         */
/* int    ApplId          - Application Identifier (Server = 1).  */
/* char *  UserId_p       - User identifier from client program. */
/*                               (For FTP server, USER subcommand data)*/
/* int    Lgth_UserId     - Length (in bytes) of user ID string. */
/* char *  AuthStr_p      - Authentication string from client.   */
/*                               (For FTP server, this is the password)*/
/* int    Lgth_AuthStr    - Length (bytes) Authentication string.*/
/* char *  ClientIPAddr_p - Internet Protocol address from which */
/*                               the session originates.         */
/* int *   Lgth_ClientIPAddr - Length (in bytes) of IP address.  */
/*                                                              */

/* Output:                                                         */
/* int *  ReturnCode: Indicates degree of success of operation: */
/*      ReturnCode = 0 - Reject logon.                          */
/*      ReturnCode = 1 - Continue logon; use initial current library*/
/*      ReturnCode = 2 - Continue logon; override initial current */
/*      library                                                  */
/*      ReturnCode = 3 - Continue logon; override user, password */
/*      ReturnCode = 4 - Continue logon; override user, password, */
/*      current library                                          */
/*      ReturnCode = 5 - Accept logon; override user profile   */
/*      ReturnCode = 6 - Accept logon; override user profile,  */
/*      current library                                          */
/* char *  UserProfile - User profile to use for this session  */
/* char *  Password    - Password to use for this session      */
/* char *  Init_Cur_Lib - Initial current library for this session */
/*                                                              */
/* Exit Normal: (See OUTPUT)                                     */
/*                                                              */
/* Exit Error:  None                                           */
/*                                                              */
/* End Function Specification *****/
static void 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 FTPUSR1 or FTPUSR2 will be
    /* allowed, and this example exit will return the initial current
    /* library as an output parameter along with a return code of 2.
    /******
    /* Continue the logon operation, Return Code = 1
    */

```



```

char Return1[] = "FTPUSR10 ";
/* Continue the logon operation, Return Code = 2 */
char Return2[] = "FTPUSR11 FTPUSR2 ";
/* Continue the logon operation, Return Code = 3 */
char Return3[] = "FTPUSR12 FTPUSR3 FTPUSR23 ";
/* Continue the logon operation, Return Code = 4 */
char Return4[] = "FTPUSER FTPUSR4 FTPUSR24 FTPUSR94 ";
int rc; /* Results of server logon request */
Qsy_USRI0300_T Receiver_var; /* QSYRUSRI API Receiver variable */
int Lgth_Receiver_var; /* Receiver variable length */
char Format_Name[8]; /* Format name buffer */
char User_Id[10]; /* User Identifier buffer */
Qus_EC_t error_code = /* QSYRUSRI API error code structure: */
{
    sizeof(Qus_EC_t), /* Set bytes provided */
    0, /* Initialize bytes available */
    ' ',' ',' ',' ',' ',' ',' ',' ',' ', /* Initialize Exception Id */
};
char *pcTest_p; /* Upper-case User Identifier pointer*/
int i; /* "For" loop counter variable */

/*****
/* Code */
*****/

/* Test validity of application ID input argument. */
if(1 NEQ ApplId)
{
    /* ERROR - Not FTP server application. */
    /* Return Code of 0 is used here to indicate */
    /* that an incorrect input argument was received. */
    /* The server logon operation will be rejected. */
    rc = 0; /* Application ID not valid */
} /* End If the application identifier is NOT for FTP server */
else /* FTP server application identifier */
{
    /* Validate the client IP address input argument. */
    rc = CheckClientAddress(ClientIPAddr_p,
        Lgth_ClientIPAddr);
    if(0 NEQ rc) /* Valid, acceptable client address */
    {
        /* Initialize User_Id; used to hold upper-cased user identifier */
        memset(User_Id, BLANK, sizeof(User_Id));

        /* Initialize pcTest_p to point to UserId_p input argument. */
        pcTest_p = UserId_p;

        /* Uppercase all of the user ID to compare for ANONYMOUS user. */
        for(i = 0; i < Lgth_UserId; i++)
        {
            User_Id[i] = (char)toupper(*pcTest_p);
            pcTest_p += 1;
        }

        /* If user has logged in as ANONYMOUS. */
        if(0 == memcmp("ANONYMOUS ", User_Id, 10))
        {
            /* Determine how to continue with ANONYMOUS logon attempt. */
            if(NULL NEQ strstr(Limit, ClientIPAddr_p))
            {
                /* If users system IP address is found in the "Limit" list, */
                /* return ReturnCode of 6, user profile and initial */
                /* current library values as output parameters. */
                memcpy(UserProfile_p, "USERA1 ", 10);
                memcpy(InitCurrLib_p, "PUBLIC ", 10);
            }
        }
    }
}

```

```

        rc = 6;
    }
else
    {
        /* Users system IP address is NOT found in the "Limit" list,*/
        /* return ReturnCode of 5, user profile output parameter; */
        /* use the initial current library that is specified by the */
        /* user profile information. */
        memcpy(UserProfile_p, "USERA1  ", 10);
        rc = 5;
    }
} /* End If USER is ANONYMOUS */

else /* Else USER is not ANONYMOUS */
    {
        /* Set receiver variable length. */
        Lgth_Receiver_var = sizeof(Qsy_USRI0300_T);
        /* Set return information format. */
        memcpy(Format_Name, "USRI0300", sizeof(Format_Name));
        /* Set user identifier passed in. */
        memset(User_Id, BLANK, sizeof(User_Id));
        memcpy(User_Id, UserId_p, Lgth_UserId);
        /* Call QSYRUSRI - Retrieve User Information API */
        QSYRUSRI(&Receiver_var, /* Return Information receiver var */
                Lgth_Receiver_var, /* Receiver variable length */
                Format_Name, /* Return information format name */
                User_Id, /* User ID seeking information */
                &error_code); /* Error return information */
        /* Check if an error occurred (byte_available not equal 0) */
        if(0 NEQ error_code.Bytes_Available)
            {
                /* Return ReturnCode of 0 only (Reject logon); */
                rc = 0; /* Reject the logon operation */
                *ReturnCode = rc; /* Assign result to ReturnCode */
            }
        else /* No error occurred from Retrieve User Info */
            {
                /* (Bytes_Available = 0) */
                /* Set current library for user profile. */
                memcpy(InitCurrLib_p, Receiver_var.Current_Library, 10);
                if(NULL NEQ strstr("CRTDFT ",
                                Receiver_var.Current_Library))
                    {
                        memcpy(InitCurrLib_p, "FTPDEFAULT", 10);
                    }
            }
        else
            {
                if(NULL NEQ strstr(Return1, UserId_p))
                    {
                        /* Return ReturnCode of 1 (Continue logon); */
                        /* Also return user profile and password output */
                        /* parameters to endure they are ignored by the server.*/
                        memcpy(UserProfile_p, UserId_p, Lgth_UserId);
                        memcpy>Password_p, AuthStr_p, Lgth_AuthStr);
                        rc = 1; /* Continue the logon operation */
                    }
            }
        else
            {
                if(NULL NEQ strstr(Return2, UserId_p))
                    {
                        /* Return ReturnCode of 2, and initial current library*/
                        /* Also return user profile and password values */
                        /* even though they will be ignored by the server. */
                        memcpy(UserProfile_p, UserId_p, Lgth_UserId);
                        memcpy>Password_p, AuthStr_p, Lgth_AuthStr);
                        memcpy(InitCurrLib_p, "FTPEXT2",

```



```

/*      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 */
        (strstr(ClientIPAddr_p, ValidChars) < Lgth_ClientIPAddr) ||
        (strstr(ClientIPAddr_p, ".") EQ 1) || /* Leading '.' in IP */
        (strstr(ClientIPAddr_p, " ") EQ 1)) /* Leading blank in IP */
    {
        /* Client's IP address not valid, or contains an incorrect character */
        rc = 0; /* Client IP address input argument not valid */
    }
    else
    {
        /* Is client system allowed to log in to FTP server? */
        if(NULL NEQ strstr(Reject, ClientIPAddr_p))
        {
            /* Return code = 0 - Reject the server logon operation, as the */
            /* client IP address is found in the global */
            /* "Reject" list. */
            rc = 0; /* Reject the logon operation */
        }
    }
}

```

```

    }
    else
    {
        /* Continue the server logon operation checks.          */
        rc = 1;          /* Continue the logon operation */
    }
}
return(rc);
}

#undef _QTMFSVRLGN_C

```

**Příklad: Program výstupního bodu Přihlášení k FTP serveru v jazyce ILE RPG:** Toto je příklad jednoduchého programu pro výstupní bod Přihlášení k FTP serveru. Je napsán v programovacím jazyce ILE RPG. Tento kód není úplný, může však být výchozím bodem pro vytváření vašich vlastních programů.

**Poznámka:** Důležité informace o souvisejících právech si můžete přečíst v Prohlášení k příkladům programového kódu.

(Formátovaný text v následujícím příkladu bude přesahovat mimo rámeček.)

```

* 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    RETCDOUT
*
* 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                RETCDOUT              Return Code (Out)
*                                     possible values: 0 = Reject Logon
*                                     1 = Continue Logon
*                                     2 = Continue Logon,
*                                     override current
*                                     library
*                                     3 = Continue Logon,
*                                     override user prf,
*                                     password
*                                     4 = Continue Logon,
*                                     override user prf,
*                                     password, current
*                                     library
*                                     5 = Accept logon with
*                                     user prf returned
*                                     6 = Accept logon with
*                                     user prf returned,
*                                     override current
*                                     library
C           PARM                USRPRFOUT            10                User Profile (Out)
C           PARM                PASSWDOUT            10                Password (Out)
C           PARM                CURLIBOUT            10                Current Lib. (Out)
C/EJECT
*****
* THE MAIN PROGRAM *
*****
*
* Check for ANONYMOUS user
*
C   1
C   USRLENIN        SUBST(P)  USRIN:1      User          10
C   User           IFEQ      Anonym
C   User           MOVEL     Anonym      USRPRFOUT
*
* Check if the user entered something as a e-mail address
*
C   AUTLENIN        IFGT      *ZERO                E-mail addr. entered
*
* Check if the E-mail address is a valid one
*
C   Z-ADD          0          i 3 0
C   '@'           SCAN      AUTIN:1      i Valid E-mail address
*
*                                     contains @ character

```

```

*
C   i IFGT      0                               Found a '@'
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      Invalid E-mail addr
C               ENDIF      Reject Logon attempt
*
C               ELSE
C               Z-ADD    0             RETCDOUT      No E-mail address
C               ENDIF      Reject Logon attempt
*
C               ELSE
*
* Any Other User: Proceed with Normal Logon Processing, but the Client address must not belong
* to network 10.xxx.xxx.xxx
*
C   3           SUBST    IPADDRIN:1   TheNet      3
C   TheNet      IFEQ     InvalidNet
C               Z-ADD    0             RETCDOUT      Wrong Net
C               ELSE
C               Z-ADD    1             RETCDOUT      Reject Logon attempt
C               ENDIF      Right Net
C               ENDIF      Continue with Logon
*
C               ENDIF
*
C               EVAL     *INLR = *ON
C               RETURN

```

**Formát výstupního bodu TCPL0100:**

Jméno formátu výstupního bodu: TCPL0100

Jméno výstupního bodu: QIBM\_QTMF\_SVR\_LOGON

Jméno výstupního bodu: QIBM\_QTMX\_SVR\_LOGON

Toto je požadovaná skupina parametrů:

1	Application identifier (Identifikátor aplikace)	Input	Binary(4)
2	User identifier (Identifikátor uživatele)	Input	Char(*)
3	Length of user identifier (Délka identifikátoru uživatele)	Input	Binary(4)
4	Authentication string (Autentizační řetězec)	Input	Char(*)
5	Length of authentication string (Délka autentizačního řetězce)	Input	Binary(4)
6	Client IP address (IP adresa klienta)	Input	Char(*)
7	Length of client IP address (Délka IP adresy klienta)	Input	Binary(4)

8	Return code (Návratový kód)	Output	Binary(4)
9	User profile (Uživatelský profil)	Output	Char(10)
10	Password (Heslo)	Output	Char(10)
11	Initial current library (Výchozí aktuální knihovna)	Output	Char(10)

## Popis parametrů

### Application identifier (Identifikátor aplikace)

INPUT; BINARY(4) Udává požadovaný aplikační server. Platné hodnoty jsou:

**1**

program FTP serveru

**2**

program REXEC serveru

### User identifier (Identifikátor uživatele)

INPUT; CHAR(\*) Identifikace uživatele získaná z programu klienta. V případě FTP serveru tento parametr obsahuje datové pole z podpříkazu USER.

### Length of user identifier (Délka identifikátoru uživatele)

INPUT; BINARY(4) Délka řetězce identifikátoru uživatele (v bajtech).

### Authentication string (Autentizační řetězec)

INPUT; CHAR(\*) Řetězec získaný z programu klienta (např. heslo).

V případě FTP serveru tento parametr obsahuje datové pole z podpříkazu PASS (Password). Od verze V5R1 platí, že když je uživatel autentizován podle certifikátu klienta, nepředají se do tohoto parametru žádná data.

### Length of authentication string (Délka autentizačního řetězce)

INPUT; BINARY(4) Délka autentizačního řetězce (v bajtech).

### Poznámka:

Pro FTP server platí, že je-li uživatel autentizován podle certifikátu klienta, obsahuje tento parametr hodnotu 0.

### Client IP address (IP adresa klienta)

INPUT; CHAR(\*) IP (Internet Protocol) adresa, ze které byla tato relace zahájena. Formát tohoto řetězce je tečková konvence se zarovnáním vlevo.

### Length of client IP address (Délka IP adresy klienta)

INPUT; BINARY(4) Udává délku IP adresy klienta (v bajtech).

### Return code (Návratový kód)

OUTPUT; BINARY(4) Udává, zda se má operace přihlášení potvrdit nebo zamítnout, zda se má provést ověření hesla a zda se má změnit výchozí aktuální knihovna. Platné hodnoty jsou:

**0**

Operace přihlášení bude zamítnuta. Parametry pro uživatelský profil, heslo a výchozí aktuální knihovnu budou ignorovány.



## 1

Operace přihlášení bude pokračovat. Použije se zadaný identifikátor a autentizační řetězec daného uživatele a jím zadaná výchozí aktuální knihovna. Identifikátor uživatele bude jeho uživatelský profil a autentizační řetězec bude heslo. Výstupní parametry pro uživatelský profil, heslo a výchozí aktuální knihovnu budou ignorovány.

### Poznámka:

Aby bylo přihlášení úspěšné, musí se autentizační řetězec shodovat s heslem nastaveným v uživatelském profilu.

## 2

Operace přihlášení bude pokračovat. Použije se identifikátor uživatele a autentizační řetězec, ale výchozí aktuální knihovna se změní na hodnotu uvedenou v parametru pro výchozí aktuální knihovnu. Identifikátor uživatele bude uživatelský profil. Autentizační řetězec bude heslo. Je třeba nastavit parametr pro výchozí aktuální knihovnu. Výstupní parametry pro uživatelský profil a heslo budou ignorovány.

### Poznámka:

Aby bylo přihlášení úspěšné, musí se autentizační řetězec shodovat s heslem nastaveným v uživatelském profilu.

## 3

Operace přihlášení bude pokračovat. Uživatelský profil a heslo se změní na hodnoty získané z výstupních parametrů vašeho programu výstupního bodu. Použije se výchozí aktuální knihovna z uživatelského profilu, který vrátí váš program výstupního bodu. Výstupní parametr pro výchozí aktuální knihovnu bude ignorován.

### Poznámka:

Aby mohlo být přihlášení úspěšně dokončeno, výstupní parametr pro heslo se musí shodovat s heslem uživatelského profilu. **Upozornění!** IBM důrazně doporučuje, abyste **nikdy** nezadávali heslo do programu výstupního bodu přímo. Šifrování například umožňuje algoritmické určení hesla.

## 4

Operace přihlášení bude pokračovat a uživatelský profil, heslo i výchozí aktuální knihovna se změní na hodnoty uvedené ve výstupních parametrech vašeho programu výstupního bodu.

### Poznámka:

Aby mohlo být přihlášení úspěšně dokončeno, výstupní parametr pro heslo se musí shodovat s heslem uživatelského profilu. **Upozornění!** IBM důrazně doporučuje, abyste **nikdy** nezadávali heslo do programu výstupního bodu přímo. Šifrování například umožňuje algoritmické určení hesla.

## 5

Operace přihlášení bude potvrzena. Uživatelský profil se změní na hodnotu, kterou váš program výstupního bodu vrátí ve výstupním parametru pro uživatelský profil. Použije se výchozí aktuální knihovna, která je uvedena v uživatelském profilu vráceném tímto programem výstupního bodu. Výstupní parametry pro výchozí aktuální knihovnu a heslo budou ignorovány.

### Poznámka:

Zadání této hodnoty potlačí normální zpracování hesla OS/400<sup>(R)</sup>. Toto bude jediná autentizace hesla.

## 6

Operace přihlášení bude potvrzena. Uživatelský profil a výchozí aktuální knihovna se změní na

hodnoty, které váš program výstupního bodu vrátí ve výstupních parametrech pro uživatelský profil a výchozí aktuální knihovnu. Výstupní parametr pro heslo bude ignorován.

**Poznámka:**

Zadání této hodnoty potlačí normální zpracování hesla OS/400. Toto bude jediná autentizace hesla.

**User profile (Uživatelský profil)**

OUTPUT; CHAR(10) Uživatelský profil pro tuto relaci. Tento parametr musí být zarovnán vlevo a doplněn mezerami.

**Password (Heslo)**

OUTPUT; CHAR(10) Heslo pro tuto relaci. Tento parametr musí být zarovnán vlevo a doplněn mezerami.

**Initial current library (Výchozí aktuální knihovna)**

OUTPUT; CHAR(10) Výchozí aktuální knihovna, která má být nastavena pro tuto relaci. Tento parametr musí být zarovnán vlevo a doplněn mezerami.

**Jděte na:**

- Poznámky k použití formátu výstupního bodu VLRQ0100.

**Můžete se také podívat na:**

- Formát TCPL0200: Tento výstupní bod obsahuje další parametry pro řízení procesu přihlášení.
- Formát TCPL0300: Tento výstupní bod označuje aplikační server, který vnesl požadavek.
- Výstupní bod Přihlášení k serveru: Pomocí výstupního bodu Přihlášení k aplikačnímu serveru TCP/IP můžete řídit autentizaci uživatelů na aplikačním serveru TCP/IP.

*Poznámky k použití formátu TCPL0100:* Pro FTP platí, že pokud není některý z vrácených výstupních parametrů platný, FTP server nepovolí operaci. V tom případě FTP server zapíše do protokolu úlohy zprávu Data from exit program for exit point &1 is missing or not valid (Data z programu pro výstupní bod &1 chybějí nebo nejsou platná).

Pro FTP dále platí, že pokud při volání programu výstupního bodu narazíte na nějakou výjimku, FTP server vydá tuto zprávu: Exception encountered for FTP exit program &1 in library &2 for exit point &3 (Byla zjištěna výjimka pro ukončovací program FTP &1 v knihovně &2 pro výstupní bod &3).

Následující tabulka shrnuje, co FTP server udělá v závislosti na hodnotě návratového kódu (parametr 8), která se vrátí FTP serveru z programu výstupního bodu.

**Poznámka:**

'Vrácená hodnota' znamená, že program výstupního bodu musí vrátit příslušnou hodnotu pro daný výstupní parametr. Tuto hodnotu potom použije FTP server ke zpracování požadavku na přihlášení.

Návratový kód	Uživatelský profil (9)	Heslo (10)	Výchozí knih. (11)
0	Ignorován	Ignorován	Ignorován
1	(Identifikátor uživatele, parametr 2)	(Heslo, parametr 4)	(Z uživatelského profilu)
2	(Identifikátor uživatele, parametr 2)	(Heslo, parametr 4)	Vrácená hodnota
3	Vrácená hodnota	Vrácená hodnota	(Z uživatelského profilu)

Návratový kód	Uživatelský profil (9)	Heslo (10)	Výchozí knih. (11)
4	Vrácená hodnota	Vrácená hodnota	Vrácená hodnota
5	Vrácená hodnota	Ignorován	(Z uživatelského profilu)
6	Vrácená hodnota	Ignorován	Vrácená hodnota

Hodnoty v závorkách udávají, kterou informaci aplikace TCP/IP použije, jestliže ignoruje výstupní hodnotu. Položka Ignorován znamená, že se nepoužije žádná hodnota; pro daný návratový kód se tedy nic nevrací.

Pro FTP server (výstupní bod QIBM\_QTMF\_SVR\_LOGON, identifikátor aplikace 1): Je-li identifikátor uživatele ANONYMOUS a tento výstupní bod volá program výstupního bodu, potom server při požadavku na heslo vydá tuto speciální odpověď: 331 Guest logon in process, send complete e-mail address as password (Probíhá přihlášení uživatele Guest, jako heslo zadejte úplnou e\_mailovou adresu). Aplikace vydá tuto zprávu, ještě než zavolá program výstupního bodu.

Když aplikace potvrdí přihlášení k serveru, FTP server vydá tuto odpověď: 230 Guest logon accepted, access restrictions apply (Přihlášení uživatele Guest potvrzeno, platí omezený přístup).

#### Pro REXEC server (identifikátor aplikace 2):

- Není-li vrácená hodnota výstupního parametru Povolit operaci platná, REXEC server nepovolí operaci. REXEC server zapíše do protokolu úlohy zprávu "Data from exit program for exit point &1 is missing or not valid" (Data z ukončovacího programu pro výstupní bod &1 chybějí nebo nejsou platná).
- Jestliže se REXEC server setká při volání programu výstupního bodu s nějakou výjimkou, nepovolí operaci. Do protokolu úlohy zapíše zprávu "Exception encountered for REXEC exit program &1 in library &2 for exit point &3" (Byla zjištěna výjimka pro ukončovací program FTP &1 v knihovně &2 pro výstupní bod &3.).

#### Formát výstupního bodu TCPL0200:

Jméno formátu výstupního bodu: TCPL0200

Jméno výstupního bodu: QIBM\_QTMF\_SVR\_LOGON

Toto je požadovaná skupina parametrů:

1	Application identifier (Identifikátor aplikace)	Input	Binary(4)
2	User identifier (Identifikátor uživatele)	Input	Char(*)
3	Length of user identifier (Délka identifikátoru uživatele)	Input	Binary(4)
4	Authentication string (Autentizační řetězec)	Input	Char(*)
5	Length of authentication string (Délka autentizačního řetězce)	Input	Binary(4)
6	Client IP address (IP adresa klienta)	Input	Char(*)
7	Length of client IP address (Délka IP adresy klienta)	Input	Binary(4)
8	Allow logon (Povolit přihlášení)	Output	Binary(4)

9	User profile (Uživatelský profil)	Output	Char(10)
10	Password (Heslo)	Output	Char(10)
11	Initial current library (Výchozí aktuální knihovna)	Input/Output	Char(10)
12	Initial home directory (Výchozí domovský adresář)	Output	Char(*)
13	Length of initial home directory (Délka výchozího domovského adresáře)	Input/Output	Binary(4)
14	Application-specific information (Informace specifická pro aplikaci)	Input/Output	Char(*)
15	Length of application-specific information (Délka informace specifické pro aplikaci)	Input	Binary(4)

## Popis parametrů

### Application identifier (Identifikátor aplikace)

INPUT; BINARY(4) Udává aplikační server, ze kterého vyšel požadavek. Platné hodnoty jsou:

**1**

program FTP serveru

### User identifier (Identifikátor uživatele)

INPUT; CHAR(\*) Identifikace uživatele získaná z programu klienta. V případě FTP serveru tento parametr obsahuje datové pole z podpříkazu USER.

### Length of user identifier (Délka identifikátoru uživatele)

INPUT; BINARY(4) Délka řetězce identifikátoru uživatele (v bajtech).

### Authentication string (Autentizační řetězec)

INPUT; CHAR(\*) Řetězec získaný z programu klienta (např. heslo).

V případě FTP serveru tento parametr obsahuje datové pole z podpříkazu PASS (Password). Od verze V5R1 platí, že když je uživatel autentizován podle certifikátu klienta, nepředají se do tohoto parametru žádná data.

### Length of authentication string (Délka autentizačního řetězce)

INPUT; BINARY(4) Délka autentizačního řetězce (v bajtech).

## Poznámka:

Pro FTP server platí, že je-li uživatel autentizován podle certifikátu klienta, obsahuje tento parametr hodnotu 0.

### Client IP address (IP adresa klienta)

INPUT; CHAR(\*) IP (Internet Protocol) adresa, ze které byla tato relace zahájena. Formát tohoto řetězce je tečková konvence se zarovnáním vlevo.

### Length of client IP address (Délka IP adresy klienta)

INPUT; BINARY(4) Udává délku IP adresy klienta (v bajtech).

### Allow logon (Povolit přihlášení)

OUTPUT; BINARY(4) Udává, zda operace přihlášení bude potvrzena nebo zamítnuta, a jak se provede autentizace hesla. Platné hodnoty jsou:

**0**

Operace přihlášení bude zamítnuta. Všechny ostatní výstupní parametry budou ignorovány.

**1**

Operace přihlášení bude pokračovat. Použije se zadaný identifikátor a autentizační řetězec daného uživatele. Identifikátor uživatele bude uživatelský profil a autentizační řetězec bude heslo. Aktuální knihovna a pracovní adresář se budou řídit podle nastavení výstupních parametrů. Výstupní parametry pro uživatelský profil a heslo budou ignorovány.

#### Poznámka:

Aby bylo přihlášení úspěšné, musí se autentizační řetězec shodovat s heslem nastaveným v uživatelském profilu.

**2**

Operace přihlášení bude pokračovat. Uživatelský profil a heslo se změní na hodnoty získané z výstupních parametrů vašeho programu výstupního bodu. Aplikace inicializuje aktuální knihovnu a pracovní adresář na základě nastavení výstupních parametrů.

#### Poznámka:

Aby mohlo být přihlášení úspěšně dokončeno, výstupní parametr pro heslo se musí shodovat s heslem uživatelského profilu. **Upozornění!** IBM důrazně doporučuje, abyste **nikdy** nezadávali heslo do programu výstupního bodu přímo. Šifrování například umožňuje algoritmické určení hesla.

**3**

Operace přihlášení bude potvrzena. Uživatelský profil se změní na hodnotu, kterou váš program výstupního bodu vrátí ve výstupním parametru pro uživatelský profil. Program inicializuje aktuální knihovnu a pracovní adresář na základě nastavení výstupních parametrů. Výstupní parametr pro heslo bude ignorován.

#### Poznámka:

Je-li váš systém nastaven na úroveň zabezpečení 20 a více, zadání této hodnoty potlačí normální zpracování hesla OS/400<sup>(R)</sup>. Toto bude jediná autentizace hesla.

### User profile (Uživatelský profil)

OUTPUT; CHAR(10) Uživatelský profil pro tuto relaci. Když je tento parametr požadován, musí být zarovnán vlevo a doplněn mezerami.

### Password (Heslo)

OUTPUT; CHAR(10) Heslo pro tuto relaci. Když je tento parametr požadován, musí být zarovnán vlevo a doplněn mezerami.

### Initial current library (Výchozí aktuální knihovna)

OUTPUT; CHAR(10) Výchozí aktuální knihovna, která má být nastavena pro tuto relaci. Když je tento parametr požadován, musí být zarovnán vlevo a doplněn mezerami. Tento parametr je při volání programu výstupního bodu nastaven na speciální hodnotu:

#### \*CURLIB

Použije se aktuální knihovna, kterou určuje uživatelský profil.

**Initial home directory (Výchozí domovský adresář)**

OUTPUT; CHAR(10) Výchozí domovský adresář, která má být nastaven pro tuto relaci. Je-li tento parametr zadán, musí obsahovat platné absolutní jméno cesty, a parametr pro délku výchozího domovského adresáře musí být nastaven na odpovídající hodnotu.

**Length of initial home directory (Délka výchozího domovského adresáře)**

INPUT/OUTPUT; BINARY(4) Délka parametru pro výchozí domovský adresář vrácená programem výstupního bodu. Při volání programu výstupního bodu je tento parametr nastaven na nulu. Jestliže program výstupního bodu hodnotu parametru nezmění, domovský adresář se nastaví na domovský adresář, který je uveden v uživatelském profilu.

**Application-specific information (Informace o aplikaci)**

INPUT/OUTPUT; CHAR(\*) Informace, která sděluje, jak je nastaveno přihlašování v souvislosti s aplikací. Informace o správném formátu najdete v tématu Formát parametru informace specifické pro aplikaci.

**Length of application-specific information (Délka informace specifické pro aplikaci INPUT; BINARY(4)** Délka informace, která se týká aplikace (v bajtech).

**Můžete se také podívat na:**

- Výstupní bod Přihlášení k serveru: Pomocí výstupního bodu Přihlášení k aplikačnímu serveru TCP/IP můžete řídit autentizaci uživatelů na aplikačním serveru TCP/IP.

*Formát parametru informace specifické pro aplikaci:* Udává-li identifikátor aplikace, že jde o program FTP serveru, pak má parametr pro informaci specifickou pro aplikaci tato pole:

Offset Dec	Offset Hex	Typ	Pole
0	0	BINARY(4)	Initial name format (Výchozí formát jména)
4	4	BINARY(4)	Initial current working directory (Výchozí aktuální pracovní adresář)
8	8	BINARY(4)	Initial file listing format (Výchozí formát výpisu souborů)
12	C	BINARY(4)	Control connection security mechanism (Mechanismus zabezpečení řídicího připojení)
16	10	BINARY(4)	Data connection encryption option (Volba šifrování datového připojení)
20	14	BINARY(2)	Control connection ciphersuite (Šifrovací sada pro řídicí připojení)
22	16	BINARY(2)	Data connection cyphersuite (Šifrovací sada pro datová připojení)

**Popis polí****Initial name format (Výchozí formát jména)**

Určuje výchozí nastavení formátu jmen souborů pro tuto relaci. Při volání programu výstupního bodu je

hodnota tohoto pole nastavena tak, že odpovídá hodnotě v konfiguračním souboru FTP serveru, která je uvedená v parametru NAMEFMT. Platné hodnoty jsou:

**0**

Použije se formát jména KNIHOVNA/SOUBOR.ČLEN. Toto nastavení odpovídá volbě NAMEFMT(\*LIB) v příkazu CHGFTP a je obdobou podpříkazu SITE NAMEFMT 0 pro FTP server.

**1**

Použije se formát jména cesty. Toto nastavení odpovídá volbě NAMEFMT(\*LIB) v příkazu CHGFTP a je obdobou podpříkazu SITE NAMEFMT 0 pro FTP server.

#### **Initial current working directory (Výchozí aktuální pracovní adresář)**

Udává výchozí nastavení aktuálního pracovního adresáře FTP serveru, což je předvolený adresář používaný při operacích se soubory a výpisy. Při volání programu výstupního bodu je hodnota tohoto pole nastavena tak, že odpovídá hodnotám konfigurace FTP serveru, uvedeným v parametru CURDIR. Platné hodnoty jsou:

**0**

Jako výchozí aktuální pracovní adresář FTP serveru se použije aktuální knihovna. Toto nastavení odpovídá volbě CURDIR(\*CURLIB) v příkazu CHGFTP.

**1**

Jako výchozí aktuální pracovní adresář FTP serveru se použije domovský adresář. Toto nastavení odpovídá volbě CURDIR(\*HOMEDIR) v příkazu CHGFTP.

**Poznámka:** Nastavíte-li toto pole na 1, musíte nastavit i pole pro výchozí formát jmen na 1.

#### **Initial file listing format (Výchozí formát výpisu souborů)**

Určuje výchozí nastavení formátu seznamu souborů pro tuto relaci. Při volání programu výstupního bodu je hodnota tohoto pole nastavena tak, že odpovídá hodnotě konfigurace FTP serveru, která je uvedená v parametru LISTFMT. Platné hodnoty jsou:

**0**

Použije se formát seznamu souborů serveru iSeries<sup>(TM)</sup>. Toto nastavení odpovídá volbě LISTFMT(\*DFT) v příkazu CHGFTP a je obdobou podpříkazu SITE LISTFMT 0 pro FTP server.

**1**

Použije se formát seznamu souborů systému UNIX<sup>(TM)</sup>. Toto nastavení odpovídá volbě LISTFMT(\*UNIX) v příkazu CHGFTP a je obdobou podpříkazu SITE LISTFMT 1 pro FTP server.

#### **Control connection security mechanism (Mechanismus zabezpečení řídicího připojení)**

Určuje mechanismus zabezpečení ochrany dat, který se použije pro řídicí připojení v této relaci FTP. Platné hodnoty jsou:

**0**

Řídicí připojení nebude zabezpečeno. **1**

Řídicí připojení bude zabezpečeno pomocí SSL (Secure Sockets Layer); mechanismus zabezpečení, který zadává FTP klient pomocí podpříkazu AUTH, je TLS-P nebo SSL.

**2**

Řídicí připojení bude zabezpečeno pomocí SSL; mechanismus zabezpečení, který zadává FTP klient pomocí podpříkazu AUTH, je TLS-C nebo TLS.

**Poznámky:**

- Toto pole je pouze vstupem do programu výstupního bodu. Změny provedené programem výstupního bodu jsou ignorovány.
- U relací připojících se přes zabezpečený FTP port je tato hodnota nastavena na 1. Relace se zabezpečeným FTP portem se chovají, jako kdyby byl na FTP server odeslán implicitní podpříkaz AUTH SSL.

**Data connection encryption option (Volba šifrování datového připojení)**

Udává, zda má být datové připojení v této relaci FTP šifrováno. Platné hodnoty jsou:

**-1**

Šifrování datového připojení FTP nebude v této relaci FTP povoleno.

**0**

Šifrování datového připojení FTP bude v této relaci FTP povoleno (ale není povinné).

**1**

Šifrování datového připojení FTP bude v této relaci FTP povinné.

**Poznámky:**

- Je-li hodnota mechanismu zabezpečení řídicího připojení nastavena na 1, nastavení volby šifrování datového připojení na -1 bude vyžadovat použití dalších podpříkazů FTP klienta, aby byl zajištěn úspěšný přenos dat. (Mechanismus zabezpečení pomocí TLS-P/SSL šifruje datová připojení standardně.)
- Je-li hodnota mechanismu zabezpečení řídicího připojení nastavena na 2, nastavení volby šifrování datového připojení na 1 bude vyžadovat použití dalších podpříkazů FTP klienta, aby byl zajištěn úspěšný přenos dat. (Mechanismus zabezpečení ochrany TLS-C/TLS nekóduje datová připojení standardně.)

**Control connection ciphersuite (Šifrovací sada pro řídicí připojení)**

Určuje šifrovací sadu SSL, používanou v této relaci FTP k šifrování řídicího připojení. Hodnoty šifrovací sady jsou definovány v API SSL (Secure Sockets Layer). Informace o těchto rozhraních API najdete pod heslem Secure Sockets Layer (SSL) APIs v tématu **Programming** v rámci aplikace iSeries Information Center.

**Poznámky:**

- Toto pole je pouze vstupem do programu výstupního bodu. Změny provedené programem výstupního bodu jsou ignorovány.
- Tato hodnota je platná pouze tehdy, když je hodnota mechanismu zabezpečení řídicího připojení nastavena na 1 nebo 2.

**Data connection ciphersuite (Šifrovací sada pro datové připojení)**

Určuje šifrovací sadu SSL, používanou k šifrování datového připojení v této relaci FTP. Při volání programu výstupního bodu, je tato hodnota nastavena na 0, což znamená, že podpora SSL může vyjednat šifrovací sadu, která se bude používat. Jestliže program výstupního bodu změní toto pole, musí být zadána platná hodnota šifrovací sady. Hodnoty šifrovací sady jsou definovány v API SSL (Secure Sockets Layer). Informace o těchto rozhraních API najdete pod heslem Secure Sockets Layer (SSL) APIs v tématu **Programming** v rámci aplikace iSeries Information Center.



**Poznámky:**

- Toto pole je ignorováno, jestliže hodnota mechanismu zabezpečení řídicího připojení je nastavena na 0 nebo volba šifrování datového připojení je nastavena na -1.
- Je-li toto pole nastaveno na jinou hodnotu než 0 nebo než je hodnota uvedenou v poli pro šifrovací soupravu řídicího připojení, může dojít k selhání při pokusu o navázání komunikace SSL mezi FTP serverem a FTP klientem, protože uvedená šifrovací souprava nemusí být FTP klientem podporovaná.

**Formát výstupního bodu TCPL0300:**

Jméno formátu výstupního bodu: TCPL0300

Jméno výstupního bodu: QIBM\_QTMF\_SVR\_LOGON

Jméno výstupního bodu: QIBM\_QTMX\_SVR\_LOGON

Toto je požadovaná skupina parametrů:

1	Application identifier (Identifikátor aplikace)	Input	Binary(4)
2	User identifier (Identifikátor uživatele)	Input	Char(*)
3	Length of user identifier (Délka identifikátoru uživatele)	Input	Binary(4)
4	Authentication string (Autentizační řetězec)	Input	Char(*)
5	Length of authentication string (Délka autentizačního řetězce)	Input	Binary(4)
6	CCSID of authentication string (CCSID autentizačního řetězce)	Input	Binary(4)
7	Client IP address (IP adresa klienta)	Input	Char(*)
8	Length of client IP address (Délka IP adresy klienta)	Input	Binary(4)
9	Allow logon (Povolit přihlášení)	Output	Binary(4)
10	User profile (Uživatelský profil)	Output	Char(10)
11	Password (Heslo)	Output	Char(*)
12	Length of password (Délka hesla)	Output	Binary(4)
13	CCSID of password (CCSID hesla)	Output	Binary(4)
14	Initial current library (Výchozí aktuální knihovna)	Input/Output	Char(10)
15	Initial home directory (Výchozí domovský adresář)	Output	Char(*)
16	Length of initial home directory (Délka výchozího domovského adresáře)	Input/Output	Binary(4)

17	CCSID of initial home directory (CCSID výchozího domovského adresáře)	Input/Output	Binary(4)
18	Application-specific information (Informace specifická pro aplikaci)	Input/Output	Char(*)
19	Length of application-specific information (Délka informace specifické pro aplikaci)	Input	Binary(4)

## Popis parametrů

### Application identifier (Identifikátor aplikace)

INPUT; BINARY(4) Udává aplikační server, ze kterého vyšel požadavek. Platné hodnoty jsou:

**1**  
program FTP serveru

**2**  
program REXEC serveru

### User identifier (Identifikátor uživatele)

INPUT; CHAR(\*) Identifikace uživatele získaná z programu klienta.

V případě FTP serveru tento parametr obsahuje datové pole z podpříkazu USER.

### Length of user identifier (Délka identifikátoru uživatele)

INPUT; BINARY(4) Délka řetězce identifikátoru uživatele (v bajtech).

### Authentication string (Autentizační řetězec)

INPUT; CHAR(\*) Řetězec získaný z programu klienta (např. heslo).

U FTP serveru tento parametr obsahuje datové pole z podpříkazu PASS (Password); pokud je však uživatel autentizován prostřednictvím certifikátu klienta, obsahuje tento parametr certifikát klienta.

### Length of authentication string (Délka autentizačního řetězce)

INPUT; BINARY(4) Délka autentizačního řetězce (v bajtech).

### CCSID of authentication string (CCSID autentizačního řetězce)

INPUT; BINARY(4) CCSID parametru autentizačního řetězce. Pro FTP server platí, že je-li uživatel autentizován podle certifikátu klienta, obsahuje tento parametr hodnotu -2.

### Client IP address (IP adresa klienta)

INPUT; CHAR(\*) IP (Internet Protocol) adresa, ze které byla tato relace zahájena. Formát tohoto řetězce je tečková konvence se zarovnáním vlevo.

### Length of client IP address (Délka IP adresy klienta)

INPUT; BINARY(4) Udává délku IP adresy klienta (v bajtech).

### Allow logon (Povolit přihlášení)

OUTPUT; BINARY(4) Udává, zda operace přihlášení bude potvrzena nebo zamítnuta, a jak se provede autentizace hesla. Platné hodnoty jsou:

**0** Operace přihlášení bude zamítnuta. Všechny ostatní výstupní parametry budou ignorovány.

**1** Operace přihlášení bude pokračovat. Použije se zadaný identifikátor a autentizační řetězec daného uživatele. Identifikátor uživatele bude uživatelský profil a autentizační řetězec bude heslo. Aktuální knihovna a pracovní adresář se budou řídit podle nastavení výstupních parametrů. Výstupní parametry pro uživatelský profil a heslo budou ignorovány.

**Poznámka:** Aby bylo přihlášení úspěšné, musí se autentizační řetězec shodovat s heslem nastaveným v uživatelském profilu.

**2**

Operace přihlášení bude pokračovat. Uživatelský profil a heslo se změní na hodnoty získané z výstupních parametrů vašeho programu výstupního bodu. Aplikace inicializuje aktuální knihovnu a pracovní adresář na základě nastavení výstupních parametrů.

**Poznámka:** Aby mohlo být přihlášení úspěšně dokončeno, výstupní parametr pro heslo se musí shodovat s heslem uživatelského profilu. **Upozornění!** IBM důrazně doporučuje, abyste **nikdy** nezadávali heslo do programu výstupního bodu přímo. Šifrování například umožňuje algoritmické určení hesla.

**3**

Operace přihlášení bude potvrzena. Uživatelský profil se změní na hodnotu, kterou váš program výstupního bodu vrátí ve výstupním parametru pro uživatelský profil. Program inicializuje aktuální knihovnu a pracovní adresář na základě nastavení výstupních parametrů. Výstupní parametr pro heslo bude ignorován.

**Poznámka:** Je-li váš systém nastaven na úroveň zabezpečení 20 a více, zadání této hodnoty potlačí normální zpracování hesla OS/400<sup>(R)</sup>. Toto bude jediná autentizace hesla.

### User profile (Uživatelský profil)

OUTPUT; CHAR(10) Uživatelský profil pro tuto relaci. Když je tento parametr požadován, musí být zarovnán vlevo a doplněn mezerami.

### Password (Heslo)

OUTPUT; CHAR(\*) Heslo pro tuto relaci. Pokud je požadováno, musíte také zadat parametry délka hesla a CCSID hesla, a tento parametr musí být zarovnán vlevo. Když je systémová hodnota QPWDVLV nastavena na 0 nebo 1, můžete zadat až 10 znaků; je-li systémová hodnota QPWDVLV nastavena na 2 nebo 3, můžete zadat až 128 znaků.

### Length of password (Délka hesla)

OUTPUT; BINARY(4) Délka hesla (v bajtech). Platný rozsah je 1 až 512 bajtů.

### CCSID of password (CCSID hesla)

OUTPUT; BINARY(4) CCSID hesla. Tento parametr musí být v programu výstupního bodu nastaven, když je zadán parametr pro heslo. Platné hodnoty jsou:

**0**

CCSID dat, která mají být konvertována, se určí podle CCSID úlohy. Je-li CCSID úlohy 65535, použije se CCSID z atributu pro předvolené CCSID (DFTCCSID) úlohy.

**1-65533**

Platný CCSID z tohoto rozsahu.

**Initial current library (Výchozí aktuální knihovna)**

OUTPUT; CHAR(10) Výchozí aktuální knihovna, která má být nastavena pro tuto relaci. Když je tento parametr požadován, musí být zarovnán vlevo a doplněn mezerami. Tento parametr je při volání programu výstupního bodu nastaven na speciální hodnotu \*CURLIB, při níž se použije aktuální knihovna uvedená v uživatelském profilu.

**Initial home directory (Výchozí domovský adresář)**

OUTPUT; CHAR(10) Výchozí domovský adresář, která má být nastaven pro tuto relaci. Je-li tento parametr zadán, musí obsahovat platné absolutní jméno cesty, a parametry pro délku výchozího domovského adresáře a CCSID výchozího domovského adresáře musí být nastaveny na odpovídající hodnotu.

**Length of initial home directory (Délka výchozího domovského adresáře)**

INPUT/OUTPUT; BINARY(4) Délka parametru pro výchozí domovský adresář vrácená programem výstupního bodu. Při volání programu výstupního bodu je tento parametr nastaven na nulu. Jestliže program výstupního bodu hodnotu parametru nezmění, domovský adresář se nastaví na domovský adresář, který je uveden v uživatelském profilu.

**CCSID of initial home directory (CCSID výchozího domovského adresáře)** OUTPUT; BINARY(4) CCSID výchozího domovského adresáře. Když je uveden výchozí domovský adresář, musí být tento parametr nastaven pomocí programu výstupního bodu. Platné hodnoty jsou:

**0**

CCSID dat, která mají být konvertována, se určí podle CCSID úlohy. Je-li CCSID úlohy 65535, použije se CCSID z atributu pro předvolené CCSID (DFTCCSID) úlohy.

**1-65533**

Platný CCSID z tohoto rozsahu.

**Application-specific information (Informace specifická pro aplikaci)**

INPUT/OUTPUT; CHAR(\*) Informace, která sděluje, jak je nastaveno přihlašování v souvislosti s aplikací. Informace o správném formátu najdete v tématu Formát parametru informace specifické pro aplikaci.

**Length of application-specific information (Délka informace specifické pro aplikaci)**

INPUT; BINARY(4) Délka informace specifické pro aplikaci (v bajtech).

**Můžete se také podívat na:**

- Výstupní bod Přihlášení k serveru: Pomocí výstupního bodu Přihlášení k aplikačnímu serveru TCP/IP můžete řídit autentizaci uživatelů na aplikačním serveru TCP/IP.

**Odstraňování programů výstupních bodů**

K odstranění nainstalovaného programu výstupního bodu použijte tento postup:

1. Na libovolný příkazový řádek iSeries<sup>(TM)</sup> zadejte **WRKREGINF**.
2. Tiskněte opakovaně klávesu Page Down, dokud se nezobrazí výstupní bod Přihlášení k FTP serveru:

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

3. Zadejte **8** do pole Vol vlevo od položky výstupního bodu a stiskněte klávesu **Enter**.
4. Na obrazovce Práce s ukončovacími programy zadejte **4** (Odstranit).
5. Do pole **Ukončovací program** zadejte jméno programu výstupního bodu.
6. Do pole Knihovna zadejte jméno knihovny, která obsahuje tento program výstupního bodu.
7. Stiskněte klávesu **Enter**.

8. Když je odstranění výstupních bodů dokončeno, zastavte a znovu spusťte FTP server.

## Metody přenosu dat

Než začnete přenášet soubory, je třeba vybrat odpovídající typ přenosu souborů. Můžete použít předvolený typ nebo ASCII, nebo zadat jiný typ. ASCII je standard pro kódování znaků v síti Internet. EBCDIC je standard pro systém iSeries<sup>(TM)</sup>. Vyberte si odpovídající typ podle následujících informací:

- ASCII použijte k přenosu souborů, které obsahují pouze text ("textové" soubory).
- EBCDIC použijte k přenosu EBCDIC dat mezi systémy v případě, kdy oba podporují EBCDIC. Tímto způsobem se vyhnete nutnosti konvertovat v obou systémech data z EBCDIC na ASCII a naopak.
- BINARY použijte k přenosu jiných než textových souborů, jako jsou binární numerická data, grafické soubory a soubory typu save systému iSeries.

Když zvolíte formát přenosu dat, jste připraveni na Přenos souborů prostřednictvím FTP.

Další informace o konkrétních typech souborů najdete v tématech:

- Přenos souborů, které obsahují pakovaná dekadická data, mezi servery iSeries
- Přenos souborů \*SAVF.
- Přenos dokumentů QDLS.
- Přenos souborů typu "kořenový", QOpenSys a QLANSrv.
- Přenos souborů pomocí QfileSvr.400.
- Přenos souborů QSYS.LIB.
- Pokyny k předběžnému vytváření souborů.
- Konverze CCSID.

## Přenos souborů, které obsahují pakovaná dekadická data, mezi servery iSeries<sup>(TM)</sup>



FTP nepodporuje konverzi speciálních číselných formátů, jako jsou dekadická data v pakovaném nebo zónovém tvaru.

Přenos pakovaných dekadických dat nebo dekadických dat v zónovém tvaru mezi servery iSeries je podporován, když použijete buď typ přenosu TYPE I (BINARY), nebo TYPE E (EBCDIC) s režimem přenosu BLOCK. Tyto typy přenosu posílají data tak, jak jsou, bez jakékoli konverze. U všech ostatních typů přenosu může dojít k nepředvídatelným výsledkům.

Když přenášíte data v pakovaném nebo zónovém tvaru v externě popsáném souboru QSYS.LIB, měl by být cílový soubor předem vytvořen stejným způsobem jako zdrojový soubor. Toto opatření se vztahuje na data, která obsahují některý speciální číselný formát, nebo když se vyžaduje klíčovaný přístup.

Když přenášíte data binárním typem přenosu (binary), musí být délka záznamu cílového souboru stejná jako délka záznamu zdrojového souboru.

Chcete-li přenést dekadická data v pakovaném nebo zónovém tvaru z nebo do systému s jinou architekturou (například S/390<sup>(R)</sup> nebo UNIX<sup>(R)</sup>), je třeba data konvertovat na tisknutelný tvar.



## Přenos souborů \*SAVF



Soubory \*SAVF musí být přenášeny jako obrazy, a proto je třeba před vydáním podpříkazu GET nebo PUT zadat příkaz FTP BINARY.

Když přenášíte soubor \*SAVF s použitím formátu jména 0, musí být soubor typu save předem vytvořen v cílovém systému. Předem vytvářet soubory se doporučuje i v jiných případech, z důvodu zachování výkonu a integrity.

Přenos souboru typu save - protože je to formát souboru typický pro iSeries - má smysl pouze tehdy, když oba servery, odesílatel i příjemce, jsou servery iSeries<sup>(TM)</sup>. Soubor typu save však může být odeslán na jiný server než iSeries a tam uložen, například za účelem zálohování. Tento soubor je pak možné později pomocí FTP přenést na server iSeries.

#### **Příklad: Přenesení souboru \*SAVF ze systému VM na server iSeries**

Následující příklad ukazuje, jak přenést soubor \*SAVF ze systému VM na server iSeries s použitím formátu jména NAMEFMT 0 a 1. Relace FTP byla již zahájena, byl vydán podpříkaz BINARY a zadán formát NAMEFMT 0.

Nejprve přeneste soubor P162484.SAVF310L z disku A systému VM na server iSeries. Protokol FTP v systému VM vyžaduje zadat soubor s tečkou mezi jménem a typem souboru. Na serveru iSeries jej zadejte jako soubor P162484 v knihovně P162484, a zadejte REPLACE, jako kdyby byl již dříve vytvořen, i když jste jej ještě nepoužili. To, že je vytvoření souboru předem povinné, voláte parametrem NAMEFMT 0.

Změňte hodnotu NAMEFMT na 1 a opakujte přenos souboru s tímto novým formátem jména. Opět zadejte REPLACE; tento soubor již existuje z předchozího kroku.

- Poznámky:**
- Kdybyste na serveru iSeries nevytvořili soubor ještě před přenosem ve formátu jména NAMEFMT 0, vypadalo by to, jako že se přenos úspěšně dokončil. Při prozkoumání souboru na serveru iSeries byste však zjistili, že byl vytvořen fyzický soubor (\*PF), a nikoli soubor typu save (\*SAVF).
  - V systému VM může být nutné provést určité předběžné zpracování, podle toho, jakým způsobem byl soubor \*SAVF do systému VM odeslán:
    - Jestliže byl soubor \*SAVF odeslán do VM pomocí FTP, stačí k jeho přenesení zpět na server iSeries zadat podpříkaz GET.
    - Jestliže byl soubor \*SAVF odeslán do VM pomocí příkazu SNDNETF (Odeslání síťového souboru), je nejprve třeba zkonvertovat tento soubor v systému VM z formátu záznamu (RECFM) typu proměnný na RECFM typu pevný, a teprve potom jej pomocí FTP přenést zpět na server iSeries. K tomu slouží příkaz COPYFILE systému VM. Například:  
COPYFILE P162484.SAVF310L A = = = (RECFM F REPLACE

**Obrázek 1.** Přenos souboru \*SAVF ze systému VM na server iSeries použitím formátu jména NAMEFMT 0 a NAMEFMT 1.

```
> 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
Enter an FTP subcommand.
====>
```



## Přenos dokumentů QDLS



Když přenášíte dokument QDLS, je atribut položky adresáře QDLS, který udává typ přenášeného dokumentu, na přijímajícím serveru iSeries<sup>(TM)</sup> standardně nastaven na typ dokumentu PCFILE pro všechny typy dokumentů s výjimkou dokumentů RFT (revisable-form text). Předvolený typ dokumentu pro dokumenty RFT je RFTDCA. Dokumenty typu RFTDCA můžete prohlížet a editovat pomocí CL příkazu WRKDOC. Dokumenty typu PCFILE nelze prohlížet ani editovat pomocí CL příkazu WRKDOC.



## Přenos souborů typu “kořenový”, QOpenSys, QLANSrv, QDLS a QOPT



Když přenášíte soubory v systému souborů “kořenovém”, QOpenSys, QLANSrv, QDLS a QOPT, musíte použít proudový režim (MODE S) a souborové členění (STRUCT F).

Soubory typu “kořenový”, QOpenSys, QDLS a QOPT mohou existovat v jakékoli platné kódové stránce. Soubory přenášené do systému souborů QLANSrv jsou označeny příznakem pro kódovou stránku definovanou pro popis síťového serveru, který odpovídá adresáři, kde se daný soubor nachází.

Konverze dat a přiřazení CCSID se mohou lišit podle použitého typu přenosu (TYPE). Můžete se podívat na téma Příznak kódové stránky CCSID u souborů iSeries<sup>(TM)</sup>. U systému souborů QLANSrv není podporován typ TYPE E.

Když přidáváte (append) data do existujícího souboru, příznak CCSID tohoto souboru se nemění. Když přidáváte data do existujícího souboru s použitím TYPE A, konvertují se tato data na kódovou stránku tohoto souboru.



## Přenos souborů pomocí QfileSvr.400



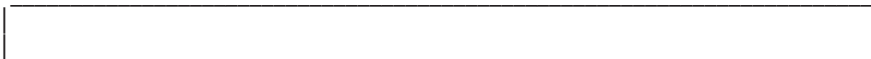
Systém souborů QFileSvr.400 umožňuje přístup k jiným systémům souborů na vzdálených serverech iSeries<sup>(TM)</sup>. Je podporován přenos v rámci systémů souborů “kořenový”, QOpenSys, QLANSrv, QDLS a QOPT. Přenos souborů v systému souborů QSYS.LIB není podporován.

Musíte použít proudový režim (MODE S) a souborové členění (STRUCT F). Například na Obrázku 9-30 (See 127), se přenáší soubor FILE.ABC do a z tří různých systémů souborů v systému AS012 pomocí QFileSvr.400 v systému AS009.

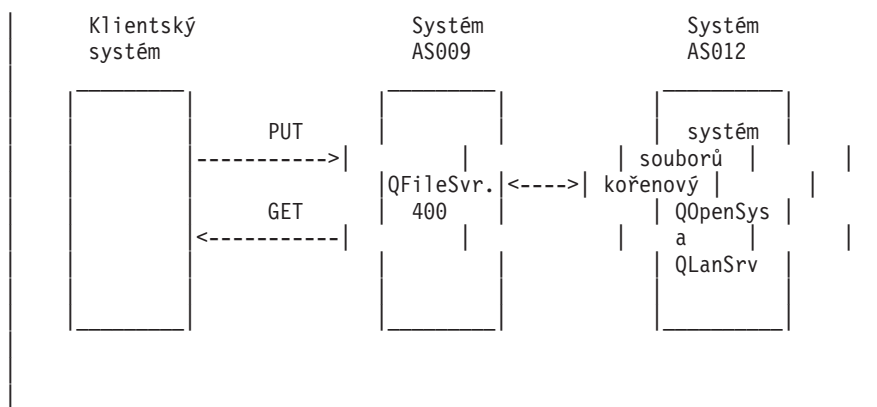
Po připojení k systému AS009 podpříkazy FTP klienta uvedené na Obrázku 9-31 (See 128) provedou přenos dat.

**Poznámka:** Uživatelské ID a heslo musí být v systému AS009 i AS012 stejné.

### Obrázek 9-30. Příklad systému souborů QFileSvr.400







**Obrázek 9-31.** Podpříkazy k přenosu dat s použitím systému souborů QFileSvr.400

```

NAMEFMT 1
LCD /CLIENTDIR1
CD /QFileSvr.400/AS012/FLSDIR
PUT FILE.ABC
GET FILE.ABC /CLIENTDIR2/FILE.ABC
CD /QFileSvr.400/AS012/QOpenSys/FLSDIR
PUT FILE.ABC
GET FILE.ABC /CLIENTDIR2/FILE.ABC (REPLACE
CD /QFileSvr.400/AS012/QLANSrv/NWS/LANSRV/DSK/K/FLSDIR
PUT FILE.ABC
GET FILE.ABC /CLIENTDIR2/FILE.ABC (REPLACE
SYSCMD RMVLNK '/CLIENTDIR2/FILE.ABC'
DELETE /QFileSvr.400/AS012/FLSDIR/FILE.ABC
DELETE /QFileSvr.400/AS012/QOpenSys/FLSDIR/FILE.ABC
DELETE /QFileSvr.400/AS012/QLANSrv/NWS/LANSRV/DSK/K/FLSDIR/FILE.ABC
QUIT
  
```



## Přenos souborů QSYS.LIB



Níže uvedené tabulky Tabulka 1 a Tabulka 2 uvádějí přehled operací FTP při proudovém režimu přenosu a při obrazovém typu přenosu u systému souborů QSYS.LIB. Až budete tyto tabulky používat, mějte na paměti následující skutečnosti:

### Kompatibilní délka záznamu a velikost souboru

Když posíláte data do již existujícího souboru, musí být délka záznamu a velikost souboru kompatibilní s posílaným souborem, jinak dojde k chybě přenosu. Velikost záznamu i souboru musí být větší nebo rovny velikosti záznamu a souboru, jakou má zdrojový soubor. Při zjišťování kompatibility s existujícím souborem musíte vzít v úvahu stávající počet záznamů, povolený počet přípon a maximální povolenou délkou záznamu. Tyto informace zjistíte zadáním příkazu iSeries<sup>(TM)</sup> DSPFD (Zobrazení popisu souboru).

### Automatické vytvoření souboru na serveru iSeries

Když server iSeries obdrží soubor, automaticky vytvoří fyzický soubor, pokud již takový soubor neexistuje. Doporučuje se však, abyste tento soubor na serveru iSeries předem vytvořili.



## Typ dat

Když přenášíte data s použitím TYPE I, data nejsou konvertována. Když soubor dosud neexistuje, je při vytvoření označen příznakem CCSID 65535.

### Poznámka:

Předem vytvořit soubor byste měli, když pomocí podpříkazů MGET a MPUT přenášíte soubory s více členy. Když soubor předem nevytvoříte, FTP vytvoří automaticky soubor, který má maximální délku záznamu rovnou nejdelšímu záznamu prvního zpracovávaného členu. Má-li některý člen souboru větší délku záznamu, dojde při přenosu tohoto členu k chybě zkrácení. Vytvoříte-li předem soubor s délkou záznamu, která vyhovuje pro všechny členy, pak k této chybě nedojde.

**Tabulka 1: Proudový režim přenosu u systému souborů QSYS.LIB**

Knihovna existuje	Soubor existuje	Člen existuje	Zvoleno přepsání	Kompatibilní délka záznamu	Kompatibilní velikost souboru	Výsledek
Ano	Ano	Ano	Ano	Ano	Ano	Data se zapíší do členu.
Ano	Ano	Ano	Ne	nerozhoduje	nerozhoduje	Přenos je zamítnut a vrátí se zpráva.
Ano	Ano	Ne	nerozhoduje	Ne	Ano	Přenos souboru se dokončí se zkrácením záznamů a vrátí se zpráva.
Ano	Ano	Ne	Ano	Ne	Ano	Přenos souboru se dokončí se zkrácením záznamů a vrátí se zpráva.
Ano	Ano	Ne	nerozhoduje	Ano	Ano	Vytvoří se člen a zapíší se do něj data.
Ano	Ano	Ne	Ne	nerozhoduje	Ne	Přenos je zamítnut a vrátí se zpráva.

Knihovna existuje	Soubor existuje	Člen existuje	Zvoleno přepsání	Kompatibilní délka záznamu	Kompatibilní velikost souboru	Výsledek
Ano	Ne	nerozhoduje	nerozhoduje	nerozhoduje	nerozhoduje	Vytvoří se soubor s délkou záznamu rovnou maximální délce záznamu příchozího souboru. Vytvoří se člen a zapíší se do něj data.
Ne	nerozhoduje	nerozhoduje	nerozhoduje	nerozhoduje	nerozhoduje	Přenos je zamítnut a vrátí se zpráva. Pomocí příkazu CRTLIB vytvořte knihovnu na vzdáleném serveru iSeries.

**Tabulka 2: Obrazový režim přenosu u systému souborů QSYS.LIB**

Knihovna existuje	Soubor existuje	Člen existuje	Zvoleno přepsání	Výsledek
Ano	Ano	Ano	Ano	Data se zapíší do členu.
Ano	Ano	Ano	Ne	Přenos je zamítnut a vrátí se zpráva.
Ano	Ano	Ne	nerozhoduje	Vytvoří se člen a data.
Ano	Ne	nerozhoduje	nerozhoduje	
Ne	nerozhoduje	nerozhoduje	nerozhoduje	



### **Příjem textových souborů u QSYS.LIB:**



Protože systém souborů iSeries<sup>(TM)</sup> QSYS.LIB interně podporuje záznamové členění, FTP na serveru iSeries konvertuje soubory přijaté serverem iSeries na záznamové členění a soubory odesílané serverem iSeries konvertuje na souborové členění FTP. Textové soubory přijaté serverem iSeries přes FTP jsou konvertovány na záznamové členění následujícím způsobem:

- Když FTP přijme soubor, který již na serveru iSeries existuje, použije se délka záznamu existujícího souboru.
- Když FTP vytvoří na serveru iSeries nový soubor, jako délku záznamu souboru použije délku (s vynecháním koncových mezer) nejdelšího řádku nebo záznamu v souboru.

Textové soubory odesílané ze serveru iSeries přes FTP jsou konvertovány na souborové členění tak, že se ze všech řádků nebo záznamů odstraní koncové mezery a odešle se tento zkrácený záznam.



## Pokyny k předběžnému vytváření souborů



Důrazně se doporučuje, abyste předem vytvořili veškeré soubory, které se chcete přenášet do systému souborů QSYS.LIB na serveru iSeries<sup>™</sup>. To je nejlepší způsob jako zajistit spolehlivý a účinný přenos dat se zachováním optimálního výkonu a integrity.

Dbejte na to, abyste alokovali dostatek záznamů, do kterých lze umístit celý přenášený soubor. U serveru iSeries k tomu slouží parametr SIZE příkazu CCRTPF (Vytvoření fyzického souboru).

Parametr RCDLEN příkazu CRTPF (Vytvoření fyzického souboru) musí odpovídat očekávané maximální délce záznamu.

### Poznámka:

V systému FTP serveru můžete předem vytvářet soubory pomocí příkazu QUOTE. V systému FTP klienta můžete předem vytvářet soubory pomocí příkazu SYSCMD.



## Konverze CCSID



Server iSeries<sup>™</sup> pomocí údaje o CCSID (identifikátor kódové sady znaků) interpretuje vstupní data a dodává výstupní data v odpovídajícím formátu pro zobrazení. Vstup může být ASCII nebo EBCDIC. Podrobné informace o konverzích CCSID najdete v těchto tématech:

- Specifikace mapovacích tabulek.
- Příznak kódové stránky CCSID u souborů iSeries.
- Pokyny k FTP týkající se NLS.



### **Specifikace mapovacích tabulek:**



U FTP klienta se mapovací tabulky specifikují příkazem FTP. U FTP serveru se to provádí příkazem CHGFTPA (Změna atributů FTP). Ke specifikaci mapovacích tabulek u FTP klienta použijte tento postup:

1. Zadejte příkaz FTP.
2. Stiskněte klávesu **PF4**. Objeví se obrazovka **Spuštění relace FTP TCP/IP**.
3. Stiskněte klávesu **F10**. Zobrazí se náznaky pro odchozí a příchozí tabulky ASCII/EBCDIC.

```
                Spuštění přenosu soub. TCP/IP (STRTCFTP)

Zapište volby, stiskněte Enter.

Vzdálený systém . . . . .

Internetová adresa . . . . .
Identifikátor kód. znak. sady      *DFT      1-65533, *DFT

                Další parametry
```

Odchozí tabulka EBCDIC/ASCII . .	*CCSID	Name, *CCSID, *DFT
Knihovna . . . . .		Name, *LIBL, *CURLIB
Příchozí tabulka ASCII/EBCDIC .	*CCSID	Name, *CCSID, *DFT
Knihovna . . . . .		Name, *LIBL, *CURLIB
Další		
F3=Konec	F4=Náznak	F5=Obnova
F12=Zrušení	F13=Jak používat tuto obrazovku	
F24=Další klávesy		

**Obrázek 1.** Nastavení mapovacích tabulek ASCII pomocí hodnoty \*CCSID.

Zadejte CCSID (a toho plynoucí mapovací tabulky), který se použije u FTP klienta. Nezměníte-li hodnotu \*DFT, použije se hodnota CCSID 00819 (ISO 8859-1 8 bitů ASCII). Můžete také zadat určitý CCSID pro příchozí a odchozí přenosy. Používání CCSID je popsáno v tématu Pokyny k FTP týkající se podpory národního jazyka (NLS).

**Poznámky:**

- Hodnoty CCSID pro dvoubajtovou znakovou sadu (DBCS) CCSID nejsou v parametru CCSID příkazu CHGFTPA povoleny. Hodnoty CCSID pro DBCS mohou být zadány pomocí podpříkazu TYPE (Zadat typ přenosu souboru).
- Aby byla zajištěna kompatibilita s verzemi nižšími než V3R1, začlenila IBM podporu mapování do FTP. U příchozích přenosů souborů typu TYPE A vede použití mapovacích tabulek ke ztrátě příznaku CCSID, když se musí vytvořit cílový soubor. IBM důrazně doporučuje používat u normálních operací podporu CCSID.



**Příznak kódové stránky CCSID u souborů iSeries<sup>(TM)</sup>:**



Když FTP vytvoří na serveru iSeries nový soubor, je tento soubor označen příznakem CCSID nebo kódové stránky tohoto CCSID, aby bylo možné identifikovat znaková data v tomto souboru. Když nahrazujete (replace) nebo přidáváte (append) data do existujícího souboru, tento příznak se v souboru nezmění. V následující tabulce je přehled o tom, jak FTP přiřazuje tyto hodnoty u různých systémů souborů a typů přenosu.

**Tabulka 1: Příznak kódové stránky CCSID u souborů systému iSeries**

Přijímající systém souborů	Typ přenosu A (ASCII)	Typ přenosu C (CCSID)	Typ přenosu E (EBCDIC)	Typ přenosu I (Image/Binary)
QSYS.LIB	CCSID určený podle identifikátoru kódové sady znaků EBCDIC pro nastavení nových databázových souborů (CRTCCSID).	'ccsid', jestliže se jedná o CCSID EBCDIC. Jde-li o CCSID ASCII, přiřadí se související předvolený CCSID EBCDIC.	65535	65535

Přijímající systém souborů	Typ přenosu A (ASCII)	Typ přenosu C (CCSID)	Typ přenosu E (EBCDIC)	Typ přenosu I (Image/Binary)
“kořenový”, QOpenSys, QDLS, QOPT	Předvolený CCSID ASCII.	Hodnota 'ccsid' udaná v podpříkazu TYPE C ccsid#.	CCSID úlohy, není-li to 65535. Je-li CCSID úlohy 65535, přiřadí se předvolený CCSID úlohy.	Předvolený CCSID ASCII.
QLanSrv	Kódová stránka ASCII z popisu síťového serveru pro daný adresář souboru.	Kódová stránka ASCII z popisu síťového serveru pro daný adresář souboru.	Nepodporován.	Kódová stránka ASCII z popisu síťového serveru pro daný adresář souboru.

**Poznámka:**

Předvolený CCSID ASCII je definován při spuštění úlohy FTP: U klienta je to v parametru CCSID příkazu STRTCPFTP (a FTP). U serveru je to v parametru CCSID atributů konfigurace FTP, které lze měnit příkazem CHGFTP. Přiřazení souborů u QFileSvr.400 závisí na přijímajícím systému souborů.



**Pokyny k FTP týkající se NLS:**



Používáte-li FTP v prostředí s různými primárními jazyky, mějte na paměti následující skutečnosti:

- Když přenášíte data s použitím typu E (neboli EBCDIC), data se ukládají tak, jak jsou a proto budou v kódové stránce EBCDIC souboru, z něhož pocházejí. To může vést k tomu, že uložený soubor bude označen neodpovídající hodnotou CCSID, pokud se u daných dvou serverů iSeries<sup>(TM)</sup> liší primární jazyk.

Jsou-li například data s kódovou stránkou 237 odeslána s použitím TYPE E do systému souborů QSYS.LIB na počítači, kde tento soubor neexistuje, data se uloží tak, jak jsou, do nového souboru označeného příznakem CCSID 65535. Pokud přijímající soubor již existuje, data budou přijata tak, jak jsou, a označena příznakem CCSID existujícího souboru, což nemusí být 237.

Abyste se vyhnuli nesprávnému přiřazení CCSID, můžete pomocí podpříkazu TYPE C CCSID (například TYPE C 237) zadat CCSID přenášených dat. Když je při přenosu zadán CCSID a data se zapíší do existujícího souboru, budou tato data konvertována na CCSID existujícího souboru. Jestliže před přenosem neexistuje žádný cílový soubor, vytvoří se soubor s příznakem tohoto zadaného CCSID.

Pokud v uvedeném příkladu nebude existovat cílový soubor, vytvoří se v přijímajícím systému soubor s CCSID 237. Bude-li již cílový soubor existovat, data se zkonvertují z CCSID 237 na CCSID cílového souboru.

- Při spouštění FTP klienta se může objevit zpráva TCP3C14: Unable to convert data from CCSID &1 to CCSID &2 (Nelze konvertovat data z CCSID &1 na CCSID &2). To se stává, když není k dispozici žádná konverze znaků mezi CCSID EBCDIC zadaným ve vaší úloze a CCSID ASCII zadaným pro tuto relaci FTP.

CCSID ASCII můžete změnit zadáním hodnoty do parametru CCSID v CL příkazu STRTCPFTP. CCSID 850, který obsahuje kódovou znakovou sadu IBM<sup>(R)</sup> Personal Computer Latin-1, je CCSID ASCII, pro který existují konverze znaků na všechny platné hodnoty CCSID úlohy.

- Když používáte FTP v režimu ASCII mezi dvěma systémy EBCDIC, data na odesílajícím systému se zkonvertují z kódové stránky EBCDIC, v níž jsou uložena, na ASCII, a potom zase z ASCII na kódovou stránku EBCDIC přijímajícího systému. Obvykle to nepředstavuje žádný problém, protože 7bitová kódová stránka ASCII používaná těmito dvěma systémy je stejná, pokud nejsou znaky EBCDIC v odesílajícím systému definovány v kódové stránce ASCII. Některé znaky v kódové stránce ASCII mohou být mapovány ve dvou různých kódových stránkách EBCDIC odlišně. To se může stát, jsou-li některé znaky

ASCII variantní (znak zaujímá v kódové stránce EBCDIC jinou pozici hexadecimálního kódu). Variantní znak může být v přijímajícím systému jinak interpretován, je-li jeho kódová stránka EBCDIC jiná než kódová stránka odesílajícího systému.



## Systémy souborů a konvence pojmenování

FTP server skládá informace o systému souborů do víceúrovňového členění, které má podobu stromu.

Systémy souborů OS/400<sup>(R)</sup>, které jsou podporovány FTP, se liší v závislosti na úrovni vydání serveru iSeries<sup>(R)</sup>. Systémy souborů v rámci OS/400 mohou používat odlišné podmínky pro data a jejich hierarchické seskupování.

### Konvence pojmenování

Každý systém souborů OS/400 má vlastní sadu pravidel pro pojmenování souborů. Formát použitý pro pojmenování souboru, se musí řídit konvencemi pojmenování v systému souborů, ve kterém je daný soubor uložen. Formáty a příklady jmen souborů v systémech souborů OS/400 podporovaných protokolem FTP jsou popsány v Integrovaném systému souborů (IFS). Další informace najdete v tématu Soubory a systémy souborů. Server může podávat informace o pojmenování souborů na jiných serverech než iSeries, pokud použijete příkaz QUOTE HELP.

### Formát jmen NAMEFMT pro FTP server

Při zahájení relace FTP serveru je formát jmen NAMEFMT nastaven na "0". Hodnotu NAMEFMT můžete změnit podpříkazem SITE.

Server automaticky změní předvolený formát NAMEFMT 0 na NAMEFMT 1 v případě, že 'první' parametr se souborem nebo cestou, který obdrží v podpříkazu:

- začíná znakem lomítka (/) nebo vlnovky (~) nebo
- je prázdný (s výjimkou podpříkazů LIST a NLST).

Žádný z dalších podpříkazů serveru, které by obsahovaly parametr se souborem nebo cestou, již na hodnotu NAMEFMT nemá vliv. Když se změní hodnota NAMEFMT, bude odpověď serveru na tento podpříkaz zahrnovat i informaci o tom, že se hodnota NAMEFMT změnila.

Hodnota NAMEFMT na serveru se například změní na "1", když první podpříkaz serveru obsahující jméno nebo cestu bude:

```
CWD /DIR1/DIR2A
```

Odpověď serveru bude:

```
250-NAMEFMT set to 1.  
250 Current directory changed to /DIR1/DIR2A.
```

### Poznámka:

Tato vlastnost umožňuje, aby běžný webový prohlížeč, který vyžaduje formát jmen NAMEFMT 1, mohl komunikovat s FTP servery iSeries, aniž by bylo nutné zadávat podpříkaz SITE NAMEFMT 1.

Více informací o formátu jmen NAMEFMT najdete v tématu Systémy souborů a konvence pojmenování.

Pokyny pro práci s formáty jmen souborů najdete v popisu podpříkazu klienta NAMEFMT (Vybrat formát pojmenování souboru).

## Systémy souborů OS/400<sup>(TM)</sup>, které jsou podporovány FTP

Systémy souborů, které jsou podporovány FTP, se liší v závislosti na úrovni vydání serveru iSeries<sup>(TM)</sup>.

### **Systém souborů knihoven QSYS.LIB - knihovny, soubory, členy**

FTP podporuje přenos souborů typu save a členů fyzických souborů, logických souborů, souborů DDM a zdrojových fyzických souborů. Pro fyzické soubory systému souborů QSYS.LIB platí, že přenášená data jsou členem souboru, který je uložen v paměti knihovny.

### **Služby knihovny dokumentů QDLS - pořadače a dokumenty**

Pro systém souborů služeb knihovny dokumentů (QDLS) platí, že přenášená data jsou dokumentem. Dokumenty QDLS jsou uloženy v adresářích, kterým se říká pořadače.

### **“kořenový” (root)**

Systém souborů /. Tento systém souborů má všechny výhody podpory proudových souborů a hierarchické struktury adresářů integrovaného systému souborů. Má vlastnosti systémů souborů DOS a OS/2<sup>(R)</sup>.

### **QOpenSys**

Systém souborů otevřených systémů. Tento systém souborů je kompatibilní se standardy otevřených systémů na bázi UNIX<sup>(R)</sup>, jako je POSIX a XPG. Podobně jako kořenový systém souborů, má výhody podpory proudových souborů a adresářů, které nabízí integrovaný systém souborů. Podporuje jména s rozlišením velkých a malých písmen.

### **QOPT**

QOPT je systém souborů optického média. Tento systém souborů poskytuje přístup k proudovým datům, která jsou uložena na optických médiích.

### **QFileSvr.400**

Systém souborů souborového serveru OS/400. Tento systém souborů umožňuje přístup k jiným systémům souborů, které se nacházejí na vzdálených serverech iSeries. FTP nepodporuje přístup do QSYS.LIB, QDLS a QOPT, které používají QFileSvr.400.

Přehled a podrobné informace o systémech souborů, které FTP podporuje, najdete v tématu Integrated File System.

## Stavové zprávy FTP serveru

Jak zadáváte během relace FTP klienta podpříkazy, objevují se na vaší obrazovce stavové zprávy v 3místném číselném kódu: xyz.

První číslo (x) udává, zda je odezva dobrá, špatná nebo neúplná. První číslo může mít pět hodnot:

- 1yz = Dobrá. Požadovaná akce byla zahájena; měla by následovat další odezva.
- 2yz = Dobrá. Požadovaná akce byla úspěšně dokončena; může být zahájen nový požadavek.
- 3yz = Neúplná. Podpříkaz byl přijat, ale požadovaná akce je pozastavena, dokud nebudou odeslány další informace.
- 4yz = Neúplná. Server podpříkaz nepotvrdil. Požadovaná akce nebude zpracována. Jedná se pouze o dočasnou chybu a můžete tedy znovu požadovat provedení akce.
- 5yz = Špatná. Podpříkaz nebyl potvrzen a požadovaná akce se neprovede.

Druhé číslo (y) představuje funkční kategorii odezvy.

- x0z=Syntaxe. Poukazuje na chyby syntaxe, příkazy, které neodpovídaly tomu, co jste se snažili udělat, a nadbytečné příkazy.
- x1z=Informace. Poukazuje na požadavky na informace, jako jsou stav nebo nápověda.
- x2z=Připojení. Poukazuje na řídicí nebo datová připojení.

- x3z=Autentizace. Poukazuje na proces přihlášení.
- x5z=System souborů. Poukazuje na stav serveru v souvislosti s požadavkem na přenos souborů.

Třetí číslo (z) ještě podrobněji rozlišuje údaj funkční kategorie.

V následující tabulce je přehled běžných kódů odpovědí a jejich význam. Text zprávy se může lišit pro různé systémy serverů.

Kód	Co znamená
110	Restart the marker reply (Odpověď - znovu spustit marker)
120	Service is ready in nnn minutes (Služba bude připravena za nnn minut)
125	Data connection is already open; transfer is starting (Datové propojení je již otevřeno; spouští se přenos)
150	File starting OK; about to open the data connection (Spuštění souboru OK; chystá se otevření datového připojení)
200	Command OK (Příkaz OK)
202	Command was not implemented; it is not used on this system (Příkaz nebyl implementován; není používán v tomto systému)
211	System status, or system help reply (Odpověď se stavem systému nebo systémovou nápovědou)
212	Directory status (Stav adresáře)
213	File status (Stav souboru)
214	Help message (Zpráva nápovědy)
220	Service is ready for a new user (Služba je připravena pro nového uživatele)
226	Closing the data connection; the requested file action was successful (Uzavírá se datové připojení; požadovaná akce se souborem byla úspěšná)
230	User is logged in (Uživatel je přihlášen)
250	Requested file action was okay; action is completed (Požadovaná akce se souborem byla OK; akce je dokončena)
257	Path name was created (Bylo vytvořeno jméno cesty)
331	Password is required (Požaduje se heslo)
332	Account is required (Požaduje se účet)
425	Cannot open the data connection (Nelze otevřít datové připojení)
426	Connection is closed; the transfer ended abnormally (Spojení je uzavřeno; přenos skončil abnormálně)
450	Requested file action was not taken; file busy (Požadovaná akce se souborem nebyla vykonána; soubor se používá)
451	Requested action ended abnormally; local error in processing (Požadovaná akce skončila abnormálně; lokální chyba při zpracování)



Kód	Co znamená
452	Requested action was not taken; insufficient storage exists in system (Požadovaná akce nebyla vykonána; systém nemá dostatek paměti)
500	Syntax error; command was unrecognized (Chyba syntaxe; příkaz nebyl rozpoznán)
501	Syntax error in the parameters or arguments (Chyba syntaxe v parametrech nebo argumentech)
502	Command was not implemented (Příkaz nebyl implementován)
503	Bad sequence of commands (Špatné pořadí příkazů)
504	Command was not implemented for that parameter (Pro tento parametr nebyl implementován příkaz)
530	Logon attempt was rejected (Pokus o přihlášení byl zamítnut)
532	Need an account for storing files (Pro uložení souborů je třeba účet)
550	Requested action was not taken; the file was not found (or no access) (Požadovaná akce nebyla vykonána; soubor nebyl nalezen (nebo k němu není přístup))
551	Requested action ended abnormally; the page type is unknown (Požadovaná akce skončila abnormálně; neznámý typ stránky)
552	Requested file action ended abnormally; storage allocation was exceeded (Požadovaná akce se souborem skončila abnormálně; byla překročena přidělená poaměť)
553	Requested action was not taken; the file name is not allowed (Požadovaná akce nebyla vykonána; jméno souboru není povoleno)

## Konvence syntaxe FTP serveru

Podpříkazy FTP serveru popsané v tomto tématu používají následující konvence syntaxe:

### Velká písmena

Písmena uvedená v definici syntaxe pro podpříkaz velkými písmeny musíte zadat přesně tak, jak jsou v definici zobrazena. Tato písmena však můžete zadat jak malými, tak i velkými písmeny.

### Slova malými písmeny nebo výrazy s rozdělovacími znaménky

Slova malými písmeny nebo výrazy s rozdělovacími znaménky, jako například vzdálenýsoubor nebo informace-o-účtu, představují proměnné, místo kterých musíte dosadit určitou informaci.

### Lomené závorky [ ]

Slova, symboly nebo fráze, které jsou umístěny v lomených závorkách, můžete považovat za volitelné.

### Levé závorky ( a hvězdičky \*

Levé závorky a hvězdičky musíte zadat přesně tak, jak se objevují v definicích syntaxe.

### Složené závorky { }

Složené závorky označují skupinu parametrů, hodnot nebo proměnných, kterou můžete opakovat.

### **Tři tečky ...**

Tři tečky znamenají, že můžete použít žádné nebo více opakování předchozí proměnné, uzavřené v lomených závorkách.

### **Svislý pruh |**

Svislý pruh mezi parametry nebo hodnotami znamená, že můžete zadat buď první, nebo druhou hodnotu (parametr), ale nikdy obě najednou. Svislé pruhy jsou uvnitř lomených nebo složených závorek.

## **Konvence syntaxe FTP klienta**

Podpříkazy FTP klienta, popsané v tomto tématu, používají následující konvence syntaxe:

### **Velká písmena**

Velká písmena vytištěná v definici syntaxe podpříkazu klienta, představují minimální počet písmen z příkazu, které musíte zadat. Podpříkazy FTP klienta můžete zadávat jak malými, tak i velkými písmeny.

### **Slova malými písmeny nebo výrazy s rozdělovacími znaménky**

Slova malými písmeny nebo výrazy s rozdělovacími znaménky, jako například vzdálenýsoubor nebo informace-o-úctu, představují proměnné, místo nichž je třeba dosadit určitou informaci.

### **Lomené závorky[ ]**

Slova, symboly nebo fráze, které jsou umístěny v lomených závorkách, můžete považovat za volitelné.

### **Levé závorky ( a hvězdičky \***

Levé závorky a hvězdičky musíte zadat přesně tak, jak se objevují v definicích syntaxe.

### **Složené závorky { }**

Složené závorky označují skupiny parametrů, hodnot nebo proměnných, které můžete opakovat.

### **Tři tečky ...**

Tři tečky znamenají, že můžete použít žádné nebo více opakování předchozí proměnné, uzavřené v lomených závorkách.

### **Svislý pruh |**

Svislý pruh mezi parametry nebo hodnotami znamená, že můžete zadat buď první, nebo druhou hodnotu (parametr), ale nikdy obě najednou. Svislé pruhy jsou uvnitř lomených nebo složených závorek.

### **Další podrobnosti k syntaxi:**

- Ohraničení parametrů podpříkazu: V tomto tématu se dozvíte, jak používat apostrof (') nebo uvozovky (") k ohraničení parametrů.
- Předvolená jména souborů v podpříkazech klienta pro přenos: V tomto tématu se dozvíte podrobnosti o předvolených hodnotách.
- Pojmenování souborů pro přenos: V tomto tématu se dozvíte podrobnosti o parametrech lokálního souboru a vzdáleného souboru.

### **Ohraničení parametrů podpříkazu**

K ohraničení parametrů podpříkazu se používají buď apostrofy ('), nebo uvozovky ("). Chcete-li zadat apostrof jako součást parametru, musíte jej zadat jako dva za sebou následující apostrofy (''), je-li celý parametr ohraničen apostrofy. Jinou možností je zadat jej jako jediný apostrof v parametru ohraničeném uvozovkami (").

Podobně, mají-li být součástí parametru uvozovky ("), musíte je zadat některým z těchto způsobů:

- Jednoduché uvozovky (") v parametru ohraničeném apostrofy.
- Dvoje za sebou následující uvozovky (""") v parametru ohraničeném uvozovkami.

Apostrofy nebo uvozovky se používají takto:

1. Jestliže jsou apostrofy nebo uvozovky uvnitř parametru shodné s počátečním a koncovým oddělovačem, musíte znak uvnitř parametru zdvojit. Například:

```
'ABCD' '12345'
  představuje ABCD'12345
"ABCD""12345"
  představuje ABCD"12345
```

2. Nejsou-li počáteční a koncový oddělovač shodné se znakem uvnitř parametru, znak uvnitř parametru se neopakuje. Například:

```
"ABCD'12345"
  představuje ABCD'12345
'ABCD"12345'
  představuje ABCD"12345
```

3. Jestliže jsou součástí parametru oba znaky - apostrof i uvozovky, jeden znak si musíte vybrat jako oddělovač. Například:

```
"ABC'12""345" nebo 'ABC'12"345'
  představuje ABC'12"345
```

#### Další podrobnosti k syntaxi:

- Konvence syntaxe FTP klienta.

### Jména souborů v podpříkazech klienta pro přenos

Vynecháte-li jméno cílového souboru v podpříkazech PUT, APPEND a GET, FTP klient doplní předvolené jméno souboru. Protože můžete zadat jména zdrojových souborů i v podpříkazech MPUT a MGET, FTP server generuje jména cílových souborů i pro MPUT a MGET. Informace o syntaxi těchto podpříkazů najdete v níže uvedené tabulce podpříkazů pro přenos dat. Ve sloupci s názvem *Cíl* je uveden parametr, pro který se dodává předvolené jméno.

Podpříkaz	Zdroj	Cíl	Další
APPEND	jméno lokálního souboru	[jméno souboru na serveru]	
PUT	jméno lokálního souboru	[jméno souboru na serveru]	
GET	jméno souboru na serveru	[jméno lokálního souboru]	[(Replace)]
MPUT	jméno lokálního souboru		
MGET	jméno souboru na serveru		[(Replace)]

### PUT a APPEND

U příkazů PUT a APPEND jsou pravidla pro tvorbu předvolených jmen rozdělena do dvou kategorií:

- Pro server iSeries<sup>(TM)</sup>
  - Je-li cílovým systémem souborů systém souborů knihoven nebo systém knihovny dokumentů, předvolené jméno se řídí pravidly pojmenování pro tyto systémy včetně formátu jména.
  - Není-li cílovým systémem souborů ani systém souborů knihoven, ani systém knihovny dokumentů, použije se jedna z těchto možností:
    - Předvolené jméno bude jméno uvedené za posledním lomítkem ve jméně zdrojového souboru.
    - Pokud jméno zdrojového souboru neobsahuje žádné lomítko, bude předvolené jméno *shodné* se jménem zdrojového souboru.
- Pro jiný server než iSeries

- Je-li zdrojovým souborem soubor systému souborů knihoven, předvolené jméno se skládá z těchto jmen: *jménosouboru.jménočlenu*. Pokud jméno členu neexistuje, pak je předvoleným jménem jméno souboru.
- Je-li zdrojovým souborem soubor služeb knihovny dokumentů, předvolené jméno se skládá ze jména souboru a přípony.
- Nemá-li zdrojovým souborem ani soubor systému souborů knihoven, ani soubor služeb knihoven dokumentů, předvoleným jménem bude jméno uvedené za posledním lomítkem jména zdroje. Jestliže jméno zdroje neobsahuje žádné lomítko, je předvolené jméno shodné se jménem zdroje.

Jestliže se jedná o server iSeries, pak server generuje předvolené jméno v těchto podpříkazech podle stejných pravidel, která platí pro podpříkaz PUT.

## GET a MGET

Jestliže se nejedná o server iSeries, vychází předvolené jméno pro podpříkazy GET a MGET z části jména zdroje, která následuje za posledním lomítkem. Neobsahuje-li jméno zdroje žádné lomítko, je předvoleným jménem celé jméno zdroje. Toto jsou pravidla vytváření předvolených jmen:

- Je-li systémem souborů klienta *systém souborů knihoven* (databáze iSeries), platí tato pravidla:
  - Obsahuje-li jméno vzdáleného souboru tečku (.), znaky před tečkou se oříznou na 10 znaků, které představují jméno lokálního souboru. Znaky za tečkou se oříznou na 10 znaků, které představují jméno členu.
  - Jestliže jméno vzdáleného souboru neobsahuje tečku, jméno souboru i jméno členu se nastaví podle jména vzdáleného souboru oříznutého na 10 znaků, které představují jméno lokálního souboru.
  - Je-li formát jména nastaven na 1, server přidá k oběma částem jména (jméno souboru a jméno členu) odpovídající přípony.
- Je-li systémem souborů klienta *systém souborů služby knihovny dokumentů*, platí tato pravidla:
  - Obsahuje-li jméno vzdáleného souboru tečku, znaky před tečkou se oříznou na 8 znaků. Znaky za tečkou se oříznou na 3 znaky.
  - Jestliže vzdálené jméno neobsahuje tečku, jméno je oříznuto na 8 znaků bez přípony.
- Pro ostatní systémy souborů platí, že předvolené jméno je jméno za posledním lomítkem jména vzdáleného souboru.

### Poznámky:

1. Soubory typu save nemají členy, takže předvolená jména pro záložní soubory neobsahují část pro člen.
2. Server zobrazuje předvolená jména, když je zapnutý režim DEBUG.

### Další podrobnosti k syntaxi:

- Konvence syntaxe FTP klienta.

## Pojmenování souborů pro přenos

Podpříkazy FTP klienta, které používáte pro přenos dat, mohou obsahovat parametr **lokální soubor** nebo **vzdálený soubor**, nebo oba najednou. Tyto parametry můžete použít k pojmenování dat, která chcete přenést. Podpříkazy pro přenos jsou:

```
APPEND
lokální soubor [vzdálený soubor]
```

```
DELETE
vzdálený soubor
```

GET  
vzdálenýsoubor [lokálnísoubor]

MDELETE  
vzdálenésoubory

MGET  
vzdálenésoubory

MPUT  
lokálnísoubory

PUT  
lokálnísoubor [vzdálenýsoubor]

Jména pro parametry lokálnísoubor a vzdálenýsoubor mohou být buď částečně nebo plně kvalifikovaná. Částečně kvalifikované jméno obsahuje jméno vlastních dat spolu s jedním nebo více jmény hierarchické posloupnosti nad daty. Plně kvalifikované jméno obsahuje všechna jména hierarchické posloupnosti nad daty.

V případě částečně kvalifikovaného jména se soubor, který se má zpracovat, určí podle aktuálního pracovního adresáře. Pracovní adresář můžete v lokálním systému klienta nastavit podpříkazem LCD. Pracovní adresář na vzdáleném serveru můžete nastavit podpříkazem CD.

Formát jména v parametru lokálnísoubor musí být v souladu s pravidly pojmenování souborů v systému iSeries. Jména pro vzdálenýsoubor se musí řídit pravidly pojmenování souborů ve vzdáleném systému.

#### **Další podrobnosti k syntaxi:**

- Ohraničení parametrů podpříkazu: K ohraničení parametrů podpříkazu můžete používat buď apostrofy (') nebo uvozovky (").
- Předvolená jména souborů v podpříkazech klienta pro přenos: Pod tímto odkazem najdete podrobnosti o předvolených jménech v podpříkazech klienta pro přenos.
- Konvence syntaxe FTP klienta: Podpříkazy FTP klienta používají tyto konvence syntaxe.

---

## **Odstraňování problémů FTP**

Toto téma podává základní informace týkající se odstraňování problémů FTP.

Určování problémů FTP

Přehled kroků, kterými zkontrolujete, zda protokol SMTP správně funguje.

Povinné podklady při nahlašování problémů FTP

Popisuje, jaké informace může požadovat servisní zástupce.

Trasování FTP serveru

Popisuje použití podpříkazu DEBUG ke sledování problémů u FTP serveru.

Trasování FTP klienta

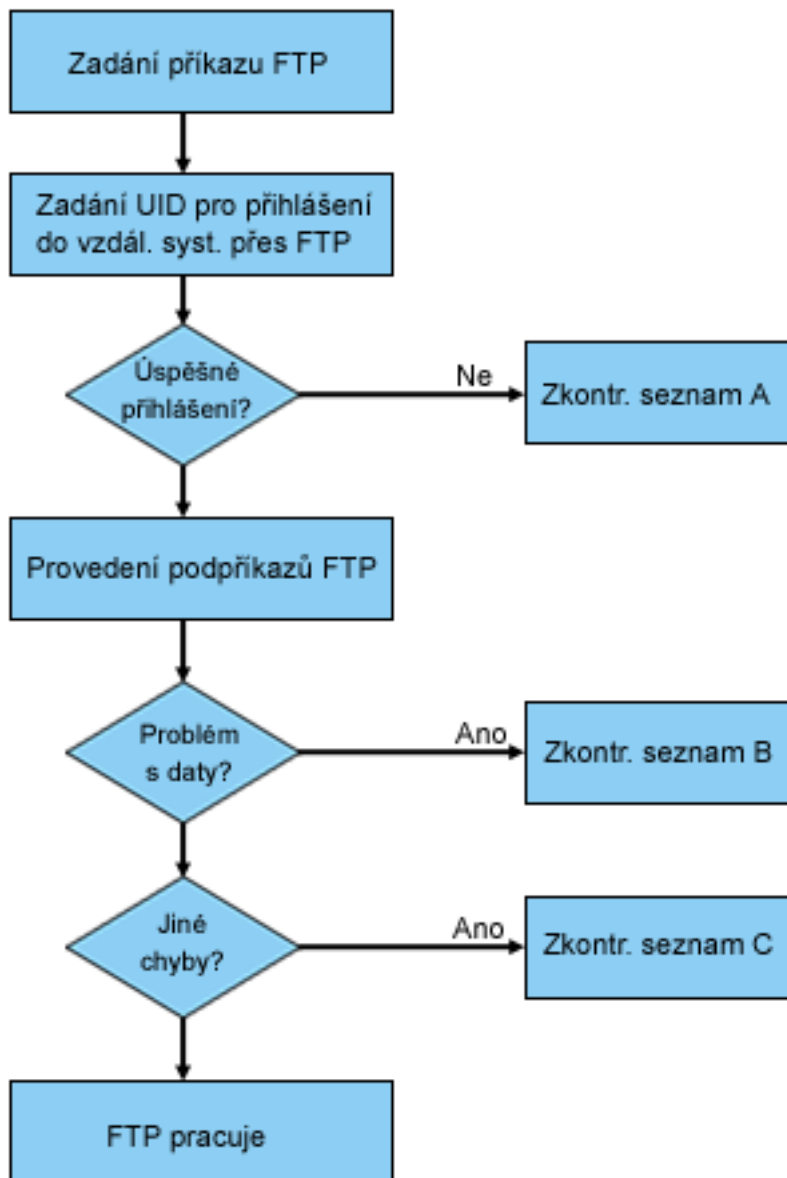
Popisuje použití podpříkazu DEBUG ke sledování problémů u FTP klienta.

Práce s protokoly úloh FTP serveru

Popisuje, jak získat soubor pro souběžný tisk z protokolu úlohy FTP serveru, z něhož můžete zjistit chyby.

## Určování problémů FTP

Narazíte-li při používání FTP na problém, zkuste zjistit příčinu pomocí následujícího vývojového diagramu. Předtím však použijte ještě vývojový diagram k určování obecných problémů TCP/IP. Seznam příčin, uvedený pod vývojovým diagramem, obsahuje kroky, které vám pomohou určit příčinu problému.



\*

Analýza problémů FTP

### Seznam příčin A

1. Je mezi připojením k FTP serveru iSeries<sup>™</sup> a zobrazením výzvy k zadání uživatelského ID dlouhá časová prodleva? Pokud ano, zkontrolujte konfiguraci serveru jmen domény (DNS) v systému iSeries. FTP server provede dotaz na DNS, jakmile je navázáno nové spojení. Problémy DNS mohou způsobit, že se server na několik minut zastaví, než obdrží odezvu.

2. Zkontrolujte, zda nebyl přidán nějaký program pro výstupní bod Přihlášení k FTP serveru. Viz téma Výstupní bod Přihlášení k serveru. Pokud byl, zkontrolujte, zda tento program výstupního bodu povoluje neúspěšné přihlášení.
3. Bylo-li požadováno heslo, zkontrolujte, zda vzdálené přihlášení vyžaduje heslo. Některé systémy vyžadují heslo, ale spojení může selhat když toto heslo není požadováno.
4. V případě potřeby nastavte heslo ve vzdáleném systému. Možná bude třeba znovu spustit systém, abyste změnili informace o zabezpečení v systému.
5. Zkontrolujte svoje uživatelské ID a heslo tak, že se pokusíte přihlásit na daný vzdálený systém. Pokud se nemůžete úspěšně přihlásit, obraťte se na vlastníka tohoto systému, aby zkontroloval správnost vašeho uživatelského ID a hesla.

### Seznam příčin B

1. Přenášíte-li binární soubory, zkontrolujte, zda je nastaven binární režim.
2. Zkontrolujte, zda mapovací tabulky v systémech serveru i klienta jsou kompatibilní. To je třeba udělat pouze v případě, že používáte vlastní mapovací tabulky.
3. Podívejte se, zda byl pro přenos zadán správný identifikátor CCSID. Pokud tomu tak není, nastavte před přenosem dat správnou hodnotu CCSID pomocí podpříkazu TYPE nebo LTYPE.
4. V systému, kam chcete data uložit, vytvořte soubor. Nastavte odpovídající délku záznamu, počet členů a počet přírůstků. Opakujte pokus s přenosem dat a zkontrolujte, zda byl tento přenos nyní úspěšný.
5. Zkontrolujte, zda máte oprávnění používat daný soubor a členy souboru.
6. Podívejte se, zda přenášený soubor neobsahuje pakovaná dekadická data nebo dekadická čísla v zónovém tvaru.
7. Přenášíte-li soubor typu save, zkontrolujte, zda používáte odpovídající způsob přenosu.

### Seznam příčin C

1. Zkontrolujte omezení velikosti souboru ve vzdáleném systému.
2. Podívejte se, zda nevypršel časový limit FTP serveru. Hodnota časové prodlevy serveru iSeries může být nastavena příkazem QUOTE TIME.
3. Pomocí příkazu NETSTAT si ověřte, zda je rozhraní \*LOOPBACK aktivní. Potom zopakujte problém provedením FTP LOOPBACK (iSeries-to-iSeries interně).
  - Pokud nemůžete problém zopakovat, jde pravděpodobně o problém vzdáleného systému.
  - Pokud se vám podařilo problém zopakovat, proveďte tyto kroky:
    - a. Jde-li o problém FTP serveru, spusťte trasování FTP serveru příkazem TRCTCPAPP.
    - b. Znovu zopakujte problém.
    - c. Ukončete připojení FTP. Viz téma Spuštění a zastavení FTP serveru.
    - d. Ukončete trasování FTP serveru příkazem TRCTCPAPP.
    - e. Vyhledejte soubor pro souběžný tisk, který má tyto vlastnosti:
      - Jméno souboru je QTMFFTRC.
      - Jméno uživatele asociované se souborem je jméno uživatele, který vydal příkaz TRCTCPAPP.

Záznam z trasování je uložen v souboru pro souběžný tisk v předvolené výstupní frontě systému, který je asociovaný s danou úlohou FTP serveru.
    - f. Odešlete tento soubor pro souběžný tisk.
    - g. Byl-li problém na straně FTP klienta iSeries, záznam z trasování lze získat pomocí podpříkazu klienta DEBUG 100.
    - h. Spouštíte-li FTP klienta interaktivně, pak k vytvoření souboru pro souběžný tisk, který obsahuje historii všech podpříkazů FTP klienta a asociovaných odpovědí FTP serveru, použijte klávesu F6 (Tisk). Spouštíte-li FTP klienta v dávkovém neobsluhovaném režimu, zapisuje se tato historie podpříkazů a odpovědí serveru do zadaného výstupního souboru. Podrobnější informace najdete v tématu "FTP jako dávková úloha".

## Povinné podklady při nahlašování problémů FTP

Když nahlašujete IBM nějaký problém FTP, mělo by toto hlášení obsahovat:

- Záznam z trasování komunikace od okamžiku poruchy (vyžadují se pouze data TCP/IP), který je dvakrát zformátován - jednou pro ASCII a jednou pro EBCDIC.
- Pokud FTP klient nebo server zaznamenal data o chybě softwaru, předejte i tato data.

### Poznámka:

Systémová hodnota QSFWERRLOG musí být nastavena na \*LOG, aby se mohl záznam chyb softwaru provádět. Pokud chyba nastane v době, kdy je systémová hodnota QSFWERRLOG je nastavena na \*NOLOG, změňte tuto hodnotu na \*LOG, zkuste chybu zopakovat a odešlete záznam dat o chybě softwaru. Když předáváte záznam dat o chybě softwaru, není nutné provádět trasování FTP.

- Protokoly úloh QTCPIP, FTP serveru nebo FTP klienta.
- Záznamy z ladicího trasování FTP klienta a FTP serveru.
- U problémů FTP klienta je nutný soubor pro souběžný tisk, obsahující relaci FTP klienta (který může být získán stisknutím klávesy F6 během relace FTP).
- Pokud je problém v integritě dat, měli byste odeslat soubor, člen nebo knihovnu, které působí problémy, plus kopii popisu tohoto souboru, členu nebo knihovny.

## Trasování FTP serveru

FTP server může být trasován z jakéhokoli serveru iSeries<sup>TM</sup> nebo jiného, na kterém je spuštěn TCP/IP. Existují dva způsoby, jak trasovat FTP server. Podpříkaz FTP serveru DBUG slouží k trasování v rámci relace FTP serveru. Příkaz TRCTCPAPP (Trasování aplikace TCP/IP) umožňuje rozsáhlé systémové trasování všech FTP serverů.

### Trasování FTP serveru pomocí podpříkazu DBUG

Zde je příklad použití podpříkazu FTP serveru DBUG:

```

                                     Protokol pro přenos souborů

Předchozí podpříkazy a zprávy FTP:
Připojení do vzdáleného hostitelského systému xxxxxxxx.xxxxxxx.xxx.xxx
na adrese n.nnn.nn.nnn pomocí portu 21.
220-QTCP na adrese xxxxxxxx.nnnnnnnn.nnn.nnn.
220 Připojení se ukončí, pokud nebude provoz více než 5 minut
215 OS/400 je vzdálený operační systém. Verze TCP/IP je
"V4R4M0".
>
331 Zadejte heslo.
230 TEST přihlášen.
250 Používáte formát pojmenování "0".
257 "QGPL" je aktuální knihovna

Zadejte podpříkaz FTP.
====> quote debug

F3=Konec      F6=Tisk      F9=Znovunastavení
F17=Začátek  F18=Konec   F21=Příkazová řádka CL
```



Spuštění trasování FTP serveru:

1. Napište QUOTE DEBUG. Tím spustíte trasování.

```
Protokol pro přenos souborů

Předchozí podpříkazy a zprávy FTP:
Připojení do vzdáleného hostitelského systému xxxxxxxx.xxxxxxx.xxx.xxx
na adrese n.nnn.nn.nnn pomocí portu 21.
220-QTCP na adrese xxxxxxxx.xxxxxxx.xxx.xxx.
220 Připojení se ukončí, pokud nebude provoz více než 5 minut
215 OS/400 je vzdálený operační systém. Verze TCP/IP je
"V4R4M0".
>
331 Zadejte heslo.
230 TEST přihlášen.
250 Používáte formát pojmenování "0".
257 "QGPL" je aktuální knihovna
> quote debug
250 Režim ladění je nyní ZAPNUTÝ
Zadejte podpříkaz FTP.
===> quote debug

F3=Konec      F6=Tisk      F9=Znovunastavení
F17=Začátek  F18=Konec   F21=Příkazová řádka CL
```

2. Proveďte operaci FTP, kterou chcete trasovat.
3. Zadejte znovu QUOTE DEBUG. Tím se trasování ukončí. Při trasování se vytvoří soubor pro souběžný tisk jménem QTMFFTRC. Tento soubor pro souběžný tisk se nachází v předvolené výstupní frontě. Jako uživatel je vždy uvedeno jméno uživatele, který byl přihlášen na FTP server, když bylo trasování ukončeno.
4. Zadejte QUIT. Tím se relace FTP ukončí.
5. Pomocí následujícího příkazu zjistíte výstupní frontu:  
DSPSYSVAL QPRTDEV  
Objeví se například následující obrazovka:

```
Zobrazení systémové hodnoty
Systémová hodnota. . . . : QPRTDEV
Popis . . . . . : Popis tiskového zařízení
Jméno tiskového zařízení . . . . : PRT01
```

Tiskové zařízení je také jméno předvolené systémové výstupní fronty.

6. Poznamenejte si jméno tiskového zařízení. V tomto příkladu je tiskovým zařízením PRT01.
7. Stiskněte klávesu F12 (Zrušení), abyste se vrátili na obrazovku, kde jste zadali příkaz DSPSYSVAL.
8. Zadejte příkaz:  
WRKOUTQ OUTQ(tiskové-zařízení)  
Za tiskové-zařízení dosadte zařízení zapsané z předchozí obrazovky. V tomto případě je výstupní frontou PRT01. Objeví se například následující obrazovka:

```
Work with Output Queue
Queue: PRT01      Library: QGPL      Status: RLS
```

```

Type options, press Enter.
  1=Send  2=Change  3=Hold  4=Delete  5=Display  6=Release  7=Messages
  8=Attributes  9=Work with printing status
Opt File      User      User Data  Sts  Pages  Copies  Form Type  Pty
-  QTCPPRT    QTCP      QTMSMTP   HLD  46     1         *STD      5
-  QTMFFTRC   QSECOFR   HLD      44     1         *STD      5

```

9. Když je na obrazovce poznámka "Další ...", přejděte stisknutím klávesy F18 (Konec) na konec seznamu souborů pro souběžný tisk.
10. Najděte poslední soubor jménem QTMFFTRC, který má jméno uživatele shodné se jménem uživatele, jenž byl přihlášen na FTP serveru, když byl vytvářen záznam z trasování.
11. Stisknutím klávesy F11 (Zobrazení 2) si můžete zobrazit datum a čas vytvoření souboru, s nímž chcete pracovat.
12. Ověřte si, že pracujete s nejnovější verzí souboru pro souběžný tisk QTMFFTRC.

Ve zprávě o problému uveďte, že jste se pokusili o trasování a že pokus nebyl úspěšný. Se zprávou o problému odešlete veškeré informace získané z trasování.

### Trasování FTP serveru pomocí příkazu TRCTCPAPP (Trasování aplikace TCP/IP)

Příkaz TRCTCPAPP (Trasování aplikace TCP/IP), který je nový od verze V4R4, umožňuje *rozsáhlé systémové* trasování *všech* FTP serverů.

Příkaz TRCTCPAPP je určen především pro vyškolené servisní a vývojové pracovníky. K používání tohoto příkazu je třeba mít zvláštní oprávnění \*SERVICE. Používejte příkaz TRCTCPAPP v situacích, kdy je zaznamenávání dat z trasování vyžadováno pro potřeby servisu nebo vývoje. Tento příkaz umožňuje zkušenému pracovníkovi dynamicky spouštět a zastavovat trasování aplikací.

Pomocí příkazu TRCTCPAPP je možné zachycovat informace z trasování FTP TCP/IP aplikace:

- Mohou být zachyceny informace z interního trasování FTP serveru iSeries. Informace, které budou pro FTP server zaznamenány, lze filtrovat pomocí vzdálené IP adresy a portu nebo uživatelského profilu iSeries. V systému může být v daném okamžiku aktivní pouze jedno trasování.

Zde uvádíme dva příklady použití příkazu TRCTCPAPP:

#### Příklad 1:

```
TRCTCPAPP APP(*FTP) SET(*ON)
```

Tento příkaz spustí trasování všech FTP serverů. Trasování ostatních TCP aplikací není ovlivněno.

#### Příklad 2:

```
TRCTCPAPP APP(*FTP) SET(*CHK)
```

Tento příkaz slouží ke kontrole stavu trasování úloh(y) FTP serveru. Předpokládáme, že poslední zadaný příkaz byl: >

```
TRCTCPAPP APP(*FTP) SET(*ON) USER(JOEC00L)
```

Odpovědi na tento příkaz může být sada zpráv, které by mohly vypadat podobně jako tyto zprávy:

```

TCP45B7 TRCTCPAPP APP(*FTP) SET(*ON) USER(JOEC00L)
      MAXSTG(*DFT) TRCFULL(*WRAP)
TCP45B1 Trasování je aktivní pro *FTP.
TCP45B2 Pro *FTP začalo zachycování dat.
TCP45B3 Data vyrovnávací paměti zalomena pro *FTP.

```

## Trasování FTP klienta

Chcete-li získat záznam z trasování FTP klienta nebo zobrazit podpříkazy odesílané na FTP server, použijte podpříkaz FTP klienta DEBUG. Podpříkaz DEBUG přepíná režim ladění programu. Zadáte-li volitelnou

hodnotu ladění, použije se tato hodnota k nastavení úrovně ladění programu. Je-li ladění programu zapnuto, zobrazí se každý podpříkaz odesílaný na server. Tyto podpříkazy jsou na začátku uvedeny řetězcem '>>>'. Abyste získali záznam z trasování FTP klienta, musíte nastavit hodnotu ladění na 100.

```
DEBug [hodnota-ladění]
```

### ***hodnota-ladění***

Je-li hodnota-ladění rovna 0, je ladění programu vypnuto. Je-li hodnota-ladění kladné celé číslo, je ladění programu zapnuto.

Nezadáte-li žádnou hodnotu, hodnota-ladění se přepne z nuly na jedničku, nebo z kladného celého čísla na nulu.

### **100**

Zahájí se trasování FTP klienta. Klient pokračuje v trasování, dokud se podpříkaz DEBUG nevypne nebo dokud se neukončí činnost FTP klienta. (Když je trasování ukončeno, může nastat delší prodleva, než se data z trasování zformátují.)

### **Poznámka:**

Trasování FTP klienta používejte pouze k nahlašování problémů softwaru IBM. Tato funkce může negativně ovlivnit výkon systému.

Ve verzi V4R4 byla přidána nová funkce FTP klienta pro ladění. Tato funkce je podobná výše popsané funkci DEBUG 100. Když je klient spuštěn, nejprve ověřuje existenci datové oblasti QTMFTPD100.

Datovou oblast QTMFTPD100 je třeba vytvořit v knihovně QTEMP pomocí příkazu:

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

Pokud již datová oblast QTMFTPD100 existuje, bude hodnota ladění nastavena na 100 a spustí se trasování FTP klienta. Účelem této funkce je umožnit ladění trasováním FTP klienta v situacích, kdy trasování FTP klienta *nelze* spustit podpříkazem DEBUG 100.

## **Práce s úlohami a s protokoly úloh FTP serveru**

Kopie protokolu úlohy FTP serveru může být užitečná k získání dalších informací o chybách, které se objevily na FTP serveru. FTP server automaticky zapíše protokol úlohy serveru do souboru pro souběžný tisk v případě, že je ukončen s chybou.

Protokol úlohy serveru lze zapsat do souboru pro souběžný tisk i bez ukončení serveru, zadáním následujícího podpříkazu FTP klienta:

```
QUOTE RCMD DSPJOBLOG
```

Chcete-li získat kopii chybových zpráv, zapsaných do protokolu úlohy serveru, musí být tento podpříkaz vydán poté, co se chyba objevila. Potom můžete zkontrolovat protokol úlohy pomocí příkazu WRKSPLF.

Tato metoda se doporučuje v případech, kdy zprávy vracené klientu ze serveru poskytují pouze minimum informací o vzniklé chybě. Tento způsob je například užitečný k získání podrobných informací o chybách vstupu/výstupu, které se na serveru objevily.

Pokud chyba zabraňuje tomu, aby byl protokol úlohy FTP serveru získán výše uvedenou cestou, zadejte následující příkaz, abyste vynutili vytvoření protokolu úlohy pro každou relaci FTP:

```
CHGJOB JOB(QUSRSYS/QTMFTPS) LOG(4 00 *SECLVL)
```

Potom zopakujte scénář, který vyvolává chybu. Abyste obnovili původní chování protokolu úlohy poté, co jste získali požadovaná data, zadejte příkaz:

```
CHGJOB JOB(QUSRSYS/QTMFTPS) LOG(4 00 *NOLOG)
```

Chcete-li, aby se soubor pro souběžný tisk z protokolu úlohy vytvářel vždy při ukončení relace FTP a při každém ukončení serveru (s chybou i bez), použijte příkaz CHGJOB (Změna popisu úlohy) takto:

```
CHGJOB JOB(QUSRSYS/QTMFTPS) LOG(4 00 *SECLVL)
```

Chcete-li, aby se soubor pro souběžný tisk z protokolu úlohy vytvářel pouze při ukončení serveru, zadejte příkaz CHGJOB takto:

```
CHGJOB JOB(QUSRSYS/QTMFTPS) LOG(4 00 *NOLIST)
```

### **Úlohy a jména úloh FTP serveru**

Úlohy FTP serveru se spustí, když se spustí příkaz STRTCP s parametrem AUTOSTART nastaveným na \*YES, nebo když se spustí příkaz STRTCP SVR s parametrem SERVER nastaveným na \*FTP nebo \*ALL. Tyto úlohy se spouštějí v podsystému QSYSWRK a jejich úkolem je monitorovat příchozí uživatele FTP. Formát jména těchto úloh je QTFTPnnnnn, kde nnnnn je číslo úlohy FTP serveru spuštěné na tomto serveru.

Chcete-li pracovat s úlohami FTP serveru, zadejte tento CL příkaz:

```
WRKACTJOB JOB(QTFTP*)
```

---

## Dodatek. Upozornění

Tyto informace platí pro produkty a služby nabízené v USA.

IBM nemusí nabízet produkty, služby a funkce popsané v tomto dokumentu v jiných zemích. Informace o produktech a službách, které jsou momentálně dostupné ve vaší oblasti, můžete získat od obchodního zástupce IBM. Žádný odkaz na produkt, program nebo službu IBM neznámá a ani z něj nelze vyvozovat, že smí být použit pouze uvedený produkt, program či služba IBM. Použit lze jakýkoli funkčně ekvivalentní produkt, program či službu neporušující práva IBM k duševnímu vlastnictví. Za vyhodnocení a ověření činnosti libovolného produktu, programu či služby jiného výrobce než IBM však odpovídá uživatel.

IBM může mít patenty nebo podané žádosti o patent, které zahrnují předmět tohoto dokumentu. Vlastnictví tohoto dokumentu vám nedává k těmto patentům žádná práva. Písemné dotazy ohledně licencí můžete zaslat na adresu:

IBM Director of Licensing  
IBM Corporation  
500 Columbus Avenue  
Thornwood, NY 10594-1785  
U.S.A.

S dotazy ohledně licencí týkajícími se informací v dvoubajtové znakové sadě (DBCS) se obraťte na IBM Intellectual Property Department ve své zemi nebo zašlete písemně dotaz na adresu:

IBM  
World Trade Asia Corporation  
Licensing  
2-31 Roppongi 3-chome, Minato-ku  
Tokyo 106, Japan

**Následující odstavec neplatí pro Velkou Británii a další země, ve kterých taková ustanovení nejsou v souladu s místními právními předpisy:** IBM POSKYTUJE TUTO PUBLIKACI "JAK JE" (AS-IS), BEZ JAKÝCHKOLIV ZÁRUK, VYJÁDŘENÝCH NEBO ODVOZENÝCH, VČETNĚ, MIMO JINÉ, ODVOZENÉ ZÁRUKY NEPORUŠENÍ PRÁV TŘETÍCH STRAN, ZÁRUKY PRODEJNOSTI NEBO VHODNOSTI PRO URČITÝ ÚČEL. Právní řády některých zemí nepřipouštějí vyloučení vyjádřených nebo odvozených záruk v určitých transakcích a proto se na vás výše uvedené omezení nemusí vztahovat.

Tato publikace může obsahovat technické nepřesnosti nebo typografické chyby. Informace zde uvedené jsou pravidelně aktualizovány a v nových vydáních této publikace již budou tyto změny zahrnuty. IBM má právo kdykoliv bez upozornění zdokonalovat anebo měnit produkt(y) anebo program(y) popsané v této publikaci.

Jakékoli odkazy v této publikaci na webové stránky jiných společností než IBM jsou poskytovány pouze pro pohodlí uživatele a nemohou být žádným způsobem vykládány jako doporučení těchto webových stránek ze strany IBM. Materiály obsažené na takovýchto webových stránkách nejsou součástí materiálů k tomuto produktu IBM a tyto webové stránky mohou být používány pouze na vlastní nebezpečí.

IBM může použít nebo distribuovat jakékoli informace, které jí sdělíte, libovolným způsobem, který společnost považuje za odpovídající, bez vzniku jakýchkoli závazků vůči vám.

Držitelé licence na tento program, kteří si přejí mít přístup i k informacím o programu za účelem (i) výměny informací mezi nezávisle vytvořenými programy a jinými programy (včetně tohoto) a (ii) vzájemného použití sdílených informací, mohou kontaktovat:

IBM Corporation  
Software Interoperability Coordinator, Department 49XA  
3605 Highway 52 N  
Rochester, MN 55901  
U.S.A.

Informace tohoto typu mohou být dostupné za určitých podmínek. V některých případech připadá v úvahu zaplacení poplatku.

IBM poskytuje licencovaný program popsáný v tomto dokumentu a veškeré dostupné licencované materiály na základě podmínek uvedených ve smlouvě IBM Customer Agreement, v Mezinárodní licenční smlouvě IBM na programy nebo v jiné ekvivalentní smlouvě.

Tyto informace obsahují příklady dat a sestav používaných při každodenních činnostech firem. Aby byly příklady co nejdokonalejší, obsahují jména osob, firem, obchodních značek a produktů. Všechna tato jména jsou smyšlená a jejich podobnost se jmény a adresami používanými ve skutečných firemních organizacích je zcela náhodná.

#### AUTORSKÁ PRÁVA:

Tyto informace obsahují ukázkové aplikační programy ve zdrojovém jazyce ilustrující programovací techniky na různých operačních platformách. Tyto ukázkové programy můžete bez závazků vůči IBM jakýmkoliv způsobem kopírovat, měnit a distribuovat za účelem vývoje, používání, odbytu či distribuce aplikačních programů odpovídajících rozhraní API pro operační platformu, pro kterou byly ukázkové programy napsány. Tyto ukázky nebyly náležitě otestovány za všech podmínek. Proto IBM nemůže zaručit ani naznačit spolehlivost, provozuschopnost ani funkčnost těchto programů. Tyto ukázkové programy můžete bez závazků vůči IBM jakýmkoliv způsobem kopírovat, měnit a distribuovat za účelem vývoje, používání, marketingu či distribuce aplikačních programů odpovídajících rozhraním API od IBM.

Každá kopie a libovolná část těchto ukázkových programů nebo jakékoli odvozené dílo musí obsahovat tuto copyrightovou výhradu:

(C) (jméno vaší firmy) (rok). Části tohoto kódu jsou odvozeny z ukázkových programů IBM. (C) Copyright IBM Corp. \_zadejte rok nebo roky\_. Všechna práva vyhrazena.

Jestliže si prohlížíte tyto informace ve formě softcopy, nemusí se zobrazit fotografie a barevné ilustrace.

---

## Ochranné známky

Následující výrazy jsou ochrannými známkami IBM v USA anebo jiných zemích:

AIX  
Application System/400  
AS/400  
e (logo)  
IBM  
iSeries  
Operating System/400  
OS/400  
pSeries  
400

UNIX je registrovaná ochranná známka skupiny The Open Group ve Spojených státech a jiných zemích.

Další jména společností, produktů nebo služeb mohou být ochrannými známkami jiných společností.

---

## Ustanovení a podmínky pro stahování a tisk publikací

Oprávnění k používání publikací, které jste se rozhodli stáhnout, závisí na níže uvedených ustanoveních a podmínkách a na vašem potvrzení, že je akceptujete.

**Osobní použití:** Pokud zachováte všechny výhrady týkající se vlastnických práv, můžete tyto publikace kopírovat pro své osobní nekomerční použití. Tyto publikace ani jakékoliv jejich části nesmíte bez výslovného souhlasu IBM distribuovat, prezentovat ani z nich vytvářet odvozená díla.

**Komerční použití:** Pokud zachováte všechny výhrady týkající se vlastnických práv, můžete tyto publikace kopírovat, distribuovat a prezentovat výhradně uvnitř svého podniku. Bez výslovného souhlasu IBM nesmíte z těchto publikací vytvářet odvozená díla ani je (nebo jejich části) nesmíte kopírovat, distribuovat či prezentovat mimo rámec svého podniku.

Kromě oprávnění, která jsou zde výslovně udělena, se na publikace a veškeré informace, data, software a další duševní vlastnictví obsažené v těchto publikacích nevztahují žádná další vyjádřená ani odvozená oprávnění, povolení či práva.

IBM si vyhrazuje právo odvolat oprávnění zde udělená, kdykoli usoudí, že používání publikací poškozuje její zájmy nebo že výše uvedené pokyny nejsou řádně dodržovány.

Tyto informace můžete stahovat, exportovat či reexportovat pouze při dodržení všech příslušných zákonů a nařízení včetně veškerých vývozních zákonů a nařízení USA. IBM NEPOSKYTUJE ŽÁDNOU ZÁRUKU, POKUD JDE O OBSAH TĚCHTO PUBLIKACÍ. PUBLIKACE JSOU POSKYTOVÁNY NA BÁZI "JAK JSOU" (AS-IS), BEZ JAKÝCHKOLIV ZÁRUK, VYJÁDŘENÝCH NEBO ODVOZENÝCH, VČETNĚ, MIMO JINÉ, ODVOZENÉ ZÁRUKY PRODEJNOSTI NEBO VHODNOSTI PRO URČITÝ ÚČEL.

Autorská práva na veškeré materiály náleží společnosti IBM Corporation.

Stažením nebo vytištěním publikace z tohoto serveru vyjadřujete svůj souhlas s těmito ustanoveními a podmínkami.

---

## **Prohlášení o vyloučení záruky na příklady programovacího kódu**

Tento dokument obsahuje příklady programového kódu.

IBM<sup>(R)</sup> vám uděluje nevýhradní copyright licenci na užívání všech uvedených příkladů programového kódu, z nichž můžete generovat obdobné funkce přizpůsobené vašim specifickým potřebám.

Veškeré ukázky programového kódu poskytnuté společností IBM jsou zde uvedeny pouze pro ilustrativní účely. Tyto příklady nebyly důkladně testovány za všech podmínek. Proto IBM nemůže zaručit ani naznačit spolehlivost, provozuschopnost ani funkčnost těchto programů.

Všechny zde obsažené programy jsou poskytovány "JAK JSOU" (AS-IS), bez záruky jakéhokoliv druhu. Odvozené záruky neporušení práv třetích stran, prodejnosti a vhodnosti pro určitý účel se tímto výslovně vylučují.









Vytištěno v Dánsku společností IBM Danmark A/S.