



# IPv6/VSE

## Migration Guide

© 1997-2016 Barnard Software, Inc.

## Table of Contents

Preface.....	5
About this Publication.....	5
Trademarks.....	5
Copyrights.....	5
Technical Support.....	6
IBM Customers.....	6
BSI Customers.....	6
BSIUsers Announcement List Server.....	7
Problem Determination.....	7
Introduction.....	8
REXX Conversion Tools.....	8
Typical Conversion Process.....	10
Manuals Available.....	11
TCP/IP Stack Start Up.....	12
Configuration Parameters.....	12
BSTTINET IPv4 Sample Start Up.....	13
BSTT6NET IPv6 Sample Start Up.....	14
SBCS Translation Tables.....	15
Format.....	16
DBCS Translation Tables.....	17
BSTTFTPS FTP Server.....	18
Configuration Parameters.....	18
IPv4 Start Up.....	18
IPv6 Start Up.....	19
FTP Server Security.....	20
Default BSTTSCTY.T Member.....	20
Creating a User Defined SAM File System.....	22
Sample BSTTSFSD.T member.....	22
View From an FTP Client.....	23
AUXDIR Specification.....	24
Sample FTP Client Log.....	24
Configuring an Emulation FTP Server.....	25
Configuring Multiple BSTTFTPS FTP Server Partitions.....	26
Sample BSTTPRXY Configuration.....	26
Sample BSTTFTPS Configuration.....	27
BSTTVNET TN3270E Server.....	28
Configuration Parameters.....	28
VTAM Definitions.....	29
CICS Terminal Types.....	29
VNETAPPL.B.....	29
ATCCON00.B.....	30
BSTTVNET LUNAME allocation.....	31
Multiple Port Numbers.....	33

## IPv6/VSE Migration Guide

* \$\$ SLI Usage.....	33
Menu Systems.....	34
IPv4/IPv6 Start Up.....	35
BSTTFTPC Batch FTP.....	36
Configuration Parameters.....	36
IPv4 Sample JCL.....	37
IPv6 Sample JCL.....	37
Converting FTPBATCH to BSTTFTPC.....	38
Edit the BSTTC2BF.PROC.....	38
Rename BSTTC2BF.PROC.....	38
Enable the BSTTJXIT \$JOBEXIT routine.....	39
Sample JCL.....	40
Output.....	40
Auto-FTP Processing.....	42
Sample JCL.....	42
CSI Auto-FTP Script Emulation.....	43
Sample JCL.....	43
BSTTLPRC LPR Client.....	44
Configuration Parameters.....	44
IPv4 Sample JCL.....	45
IPv6 Sample JCL.....	45
BSTTLPRC Auto-LPR Processing.....	46
Sample JCL.....	46
BSTTMTPC Batch EMAIL Client.....	47
Configuration Parameters.....	47
IPv4 Sample JCL.....	48
IPv6 Sample JCL.....	49
Converting CLIENT/EMAIL to BSTTMTPC.....	50
Edit the BSTTC2BE.PROC.....	50
Rename BSTTC2BF.PROC.....	50
Enable the BSTTJXIT \$JOBEXIT routine.....	51
Sample JCL.....	52
Output.....	53
BSTTREXC Batch Remote Execution Client.....	54
Configuration Parameters.....	54
Linux System Considerations.....	54
IPv4 Sample JCL.....	55
IPv6 Sample JCL.....	55
Replacing Batch Telnet.....	56
Sample Batch TELNET.....	56
Replacement BSTTREXC Job.....	56
Sample Batch Telnet File Delete.....	57
Sample BSTTFTPC File Delete.....	57
BSTTTELN Batch Telnet Client.....	58
Configuration Parameters.....	58

## IPv6/VSE Migration Guide

Sample Batch TELNET.....	59
Replacement BSTTTELN Job.....	59
BSTTLPDS Line Print Daemon Service.....	60
Configuration Parameters.....	60
QUEUE and OUTPUT Commands.....	60
Sample BSTTLPDS Execution JCL.....	61
BSIREXXC Symbolic Variable Support.....	62
Symbolic Variables.....	63
Overview.....	64
Updating ARXJOBTB.....	65
Invoking BSIREXXC.....	67
Special BSTTMTPC Considerations.....	68
BSTTFTPS FTP Server Scripts.....	69
Mounting a Filesystem.....	69
Script Example.....	69
FTP Server SITE commands.....	70
BSTTPREP Converting CSI EXEC TCP to EZASOKET.....	72
CICS TS FTP Client Transactions.....	73
IBM Virtual Tape Server.....	74
Modified JCL.....	74
Transferring .aws VTAPE Files using FTP.....	75
IBM Connector Server.....	77
Modified JCL.....	77
CICS TS Web Services.....	78
CICS TS JCL Changes.....	78
CEDA TCPIP Entries.....	78
ASCII/EBCDIC Conversion.....	79
CICS TS Web Services Test.....	80

## Preface

### ***About this Publication***

This is the IPv6/VSE Migration Guide manual. The manual provides a reference for migration from TCP/IP for VSE to IPv6/VSE.

### **Trademarks**

The following are lists of the trademark and products referenced in this manual. Symbols for trademarks and registered trademarks do not appear in subsequent references.

Barnard Software, Inc.

TCP/IP-TOOLS and IPv6/VSE are registered trademarks of Barnard Software, Inc.

International Business Machines Corporation

IBM, the IBM logo, and [ibm.com](http://ibm.com) are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at “Copyright and trademark information” at [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both. Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

### **Copyrights**

This software and documentation is covered by the following copyright:

Copyright (c) 1998-2012 Barnard Software, Inc. All rights reserved.

## ***Technical Support***

### **IBM Customers**

IBM IPv6/VSE customers should contact IBM for support.

### **BSI Customers**

Technical Support is available from Barnard Software, Inc. by phone, mail or email:

Barnard Software, Inc.  
806 Silk Oak Terrace  
Lake Mary, FL 32746  
Phone: 1-407-323-4773  
Support: [bsiopti@bsiopti.com](mailto:bsiopti@bsiopti.com)  
Sales: [bsisales@bsiopti.com](mailto:bsisales@bsiopti.com)

Support is available from 9:00 a.m. through 5:00 p.m. EST, Monday through Friday.

If a TSR (Technical Support Representative) is not available at the time of your call, please leave a message and a TSR will return your call as soon as possible. Please provide the following information: name, company, phone number, product name, product release level, and a short description of the problem.

### ***BSIUsers Announcement List Server***

When new releases of IPv6/VSE are available BSI will post an announcement on its BSIUsers announcement list.

To subscribe to the BSIUsers announcement list send an email to this email address  
[BSIUsers-subscribe@yahoogroups.com](mailto:BSIUsers-subscribe@yahoogroups.com)

To unsubscribe to the BSIUsers announcement list send an email to this email address  
[BSIUsers-unsubscribe@yahoogroups.com](mailto:BSIUsers-unsubscribe@yahoogroups.com)

### ***Problem Determination***

If you have a problem using a IPv6/VSE application always check the SYSLST output for additional information and messages. Most messages are written to SYSLST and not to the VSE/ESA or z/VSE system console.

When contacting BSI for technical support always have the applications JCL/commands, console and SYSLST output available for problem determination. The SYSLST output is very important.

While a IPv6/VSE application is running, you can issue the AR CANCEL XX,PARTDUMP command to terminate IPv6/VSE application and dump the partition to SYSLST. Using the VSE/POWER Flush (F) command cancels the IPv6/VSE application partition without a dump.

If the IPv6/VSE application partition stops responding to its console interface, use the AR DUMP XX command to obtain a dump of the partition.

## Introduction

The manual is designed to give new IPv6/VSE users some tips and tricks to allow them to migrate their TCP/IP operations to IPv6/VSE quickly and easily.

While the product is named IPv6/VSE, the product supports both IPv4 and IPv6 communications. IPv6/VSE provides a full-function IPv4 stack and applications as well as a full-function IPv6 stack and applications. Both TCP/IP stacks (IPv4 and IPv6) can be run together, individually or even stand alone.

Sometimes we hear questions about why a z/VSE customer would move to IPv6/VSE. The basic reasons are the high availability, stability and performance of IPv6/VSE along with the quality support provided by Barnard Software, Inc.

It is important to point out that this is a migration. Both TCP/IP for VSE and IPv6/VSE can coexist and run on the same z/VSE system at the same time. There is no requirement that you convert from TCP/IP for VSE to IPv6/VSE in a single evening or a single weekend. The migration to IPv6/VSE can be done on a step-by-step basis as time permits.

### **REXX Conversion Tools**

There are a number of REXX conversion tools provided to assist in the migration process. Each of these conversion tools will be discussed in detail in a later chapter.

REXX EXEC	Description
BSIREXXC.PROC	Symbolic Variable Support for BSTTFTPC, BSTTLPRC, BSTTREXC, BSTTMPTC Provided by Dave Clark
BSTTAFTP.PROC	Auto-FTP processor
BSTTALPR.PROC	Auto-LPR processor
REXXAFTP.PROC	CSI Auto-FTP script emulator CSI Auto-LPR script emulator Provided by Rich Smrcina
BSTTPREP.PROC	Converts CSI EXEC TCP to EZASOKET
BSTTC2BF.PROC	Converts CSI FTPBATCH commands to BSI BSTTFTPC commands
BSTTC2BE.PROC	Converts CSI CLIENT/EMAIL commands to BSI BSTTMTPC commands using BSIREXXC





## ***Typical Conversion Process***

There are several steps involved in the migration process. Each of these migration steps will be discussed in detail in a later chapter.

Step	Description
1	Bring up the BSTITNET and/or BSTIT6NET TCP/IP stacks
2	Bring up the BSTITFTPS FTP Server
3	Bring up BSTITVNET TN3270E Server
4	Migrate FTPBATCH to BSTITFTPC
5	Test/modify FTP Server scripts (if necessary)
6	Migrate CSI EXEC TCP applications to EZASOCKET
7	Special case items (CICS FTP transactions, etc.)
8	Install/enable SSL support (if needed)

**Manuals Available**

Manual Name	Description
IP-IPv6_VSE-MigrationGuide.pdf	This manual
IP-IPv6_VSE-IPv4_InstallationGuide.pdf	IPv4 Installation Guide
IP-IPv6_VSE-IPv6_InstallationGuide.pdf	IPv6 Installation Guide
IP-IPv6_VSE-IPv4_UsersGuide.pdf	IPv4 Users Guide
IP-IPv6_VSE-IPv6_UsersGuide.pdf	IPv6 Users Guide
IP-IPv6_VSE-SSL_Install_Users_Programming.pdf	SSL Installation, Users and Programming Guide
IP-IPv6_VSE-ProgrammingGuide.pdf	Programming Guide
IP-IPv6_VSE-Debugging_Guide.pdf	Debugging Guide
IP-IPv6_VSE-MessagesAndCodes.pdf	Message and Codes
IP-IPv6_VSE-SSL_Creating and Using SSL Certificates.pdf	SSL Creating and Using SSL Certificates
IP-IPv6_VSE-DesignAndFlow.pdf	Design and Flow
IP-ITAMSupplementGuide.pdf	ITAM Supplement (BSI Only)

## TCP/IP Stack Start Up

Step 1, Bring up the BSTTINET and/or BSTT6NET TCP/IP stacks.

The complete reference for this step can be found in the IPv4 Installation Guide and the IPv6 Installation Guide manuals.

While the product is named IPv6/VSE, the product supports both IPv4 and IPv6 communications. IPv6/VSE provides a full-function IPv4 stack and applications as well as a full-function IPv6 stack and applications. Both TCP/IP stacks (IPv4 and IPv6) can be run together, individually or even stand alone.

BSTTINET is the IPv4 TCP/IP stack and BSTT6NET is the IPv6 TCP/IP stack. These two TCP/IP stacks can be run together, separately or stand alone.

Starting the TCP/IP stacks is quite simple. The following table shows the information you will need to start the BSTTINET and BSTT6NET TCP/IP stacks.

### Configuration Parameters

	Sample IPv4	Sample IPv6	Your IPv4	Your IPv6
DEVICE	OSAX	OSAX		
CUU	710-712	720-722		
Stack ID	00	66		
IP Address	192.168.1.238	FD00:806:1::7		
Subnet Mask	255.255.255.0	/64		
Gateway	192.168.1.100	FD00:806:1::1		
DNS	192.168.1.101	FD00:806:1::2		
DNS	192.168.1.102	FD00:806:1::3		
Domain	.bsitcpip.com	.bsitcpip.com		
HOST	vse51	zvse51		
BSI library	bsilib.slib	bsilib.slib		
Config library	PRD2.CONFIG	PRD2.CONFIG		

The following two sets of JCL show the start up JCL and commands for each TCP/IP stack. Once running, these two TCP/IP stacks act as a single dual-stack image. Each of the TCP/IP stacks runs in its own partition (for example, S1 and S2). This provides for maximum reliability and performance.

## BSTTINET IPv4 Sample Start Up

This is the BSTTINET IPv4 stack sample start up.

This JCL assumes an ADD 710:712,OSAX has been done at z/VSE system IPL.

```
// OPTION LOG, PARTDUMP, NOSYSDUMP, SADUMP=1
// ASSGN SYS000, SYSLST
// LIBDEF PHASE, SEARCH=(bsilib.slib)
// LIBDEF SOURCE, SEARCH=(PRD2.CONFIG,bsilib.slib)
// SETPFIX LIMIT=(256K,1280K)
// EXEC BSTTINET, SIZE=BSTTINET, OS390
ID 00
INTERVAL 120
*
DEVICE OSAX710 OSAX 710 portname 712
LINK OSAX710 0 192.168.1.238 255.255.255.0 1500
*
ROUTE OSAX710 192.168.1.0 255.255.255.0 0.0.0.0 0
ROUTE OSAX710 0.0.0.0 0.0.0.0 192.168.1.100 1
*
HOST VSE51 192.168.1.238
*
DNS 192.168.1.101 PRI
DNS 192.168.1.102 SEC
DOMAIN .bsitcpip.com
*
ATTACH TCP/IP
/*
```

## BSTT6NET IPv6 Sample Start Up

This is the BSTT6NET IPv6 stack sample start up.

This JCL assumes an ADD 720:722,OSAX has been done at z/VSE system IPL.

```
// OPTION LOG, PARTDUMP, NOSYSDUMP, SADUMP=1
// ASSGN SYS000, SYSLST
// LIBDEF PHASE, SEARCH=(bsilib.slib)
// LIBDEF SOURCE, SEARCH=(PRD2.CONFIG,bsilib.slib)
// SETPFIX LIMIT=(256K,1280K)
// EXEC BSTT6NET, SIZE=BSTT6NET, OS390
ID 66
COUPLE 00
INTERVAL 120
*
DEVICE OSAX720 OSAX 720 portname 722
LINK OSAX720 0 FD00:806:1::7 /64 8192
*
ROUTE OSAX720 FD00:806:1::0 /64 ::0 0
ROUTE OSAX720 ::0 ::0 FD00:806::1:1 1
*
HOST ZVSE51 FD00:806::1:7
*
DNS FD00:806:1::2 PRI
DNS FD00:806:1::3 SEC
DOMAIN .bsitcpip.com
*
ATTACH TCP/IP
/*
```

## **SBCS Translation Tables**

Before starting the BSI applications determine if you are using special Single Byte Character Set translation tables. If you have created special tables for use with TCP/IP for VSE, you will want to add these tables to the BSI BSTTSBCS.T translation table library member.

Single Byte Character Set translation tables are located in the BSTTSBCS.T library member. This member is located in the BSI installation lib.slib. If you modify this table, you should place your modified BSTTSBCS.T member into a configuration lib.slib (E.g., PRD2.CONFIG). Remember, any time you receive an update from BSI you will receive a new BSTTSBCS.T member. Placing the BSTTSBCS.T member in a configuration lib.slib ensure your changes will not be lost. The BSTTSBCS.T member is accessed through the // LIBDEF SOURCE,SEARCH chain.

The default SBCS table is called US\_ENG\_03. If you change this table you should rename the original table and then create a new default table with the US\_ENG\_03 name.

## IPv6/VSE Migration Guide

```
US_ENG_03
X'000102039C09867F978D8E0B0C0D0E0F'
X'101112139D8508871819928F1C1D1E1F'
X'80818283840A171B88898A8B8C050607'
X'909116939495960498999A9B14159E1A'
X'20A0E2E4E0E1E3E5E7F1A22E3C282B7C'
X'26E9EAE8E8E8E8E8E8E8E8E8E8E8E8E8'
X'2D2FC2C4C0C1C3C5C7D1A62C255F3E3F'
X'F8C9CACBC8CDCECFCC603A2340273D22'
X'D8616263646566676869ABBBF0FDFEB1'
X'B06A6B6C6D6E6F707172AABAE6B8C6A4'
X'B57E737475767778797AA1BFD0DDDEAE'
X'5EA3A5B7A9A7B6BCBDBE5B5DAFA8B4D7'
X'7B414243444546474849ADF4F6F2F3F5'
X'7D4A4B4C4D4E4F505152B9FBFCF9FAFF'
X'5CF7535455565758595AB2D4D6D2D3D5'
X'30313233343536373839B3DBDCD9DA9F'
X'00010203372D2E2F1605250B0C0D0E0F'
X'101112133C3D322618193F271C1D1E1F'
X'405A7F7B5B6C507D4D5D5C4E6B604B61'
X'F0F1F2F3F4F5F6F7F8F97A5E4C7E6E6F'
X'7CC1C2C3C4C5C6C7C8C9D1D2D3D4D5D6'
X'D7D8D9E2E3E4E5E6E7E8E9BAE0BBB06D'
X'79818283848586878889919293949596'
X'979899A2A3A4A5A6A7A8A9C04FD0A107'
X'202122232415061728292A2B2C090A1B'
X'30311A333435360838393A3B04143EFF'
X'41AA4AB19FB26AB5BDB49A8A5FCAAFBC'
X'908FEAFABEA0B6B39DDA9B8BB7B8B9AB'
X'6465626663679E687471727378757677'
X'AC69E8E8E8E8E8E8E8E8E8E8E8E8E8E8'
X'4445424643479C485451525358555657'
X'8C49CDCECBCFCCE170DDDEDBDC8D8EDF'
```

### **Format**

- The 1st line is a 16 character table name.
- The next 16 lines is the EBCDIC-to-ASCII portion.
- The final 16 lines is the ASCII-to-EBCDIC portion.
- When adding your own special tables use a unique name.



## DBCS Translation Tables

Before starting the BSI applications determine if you are using Double Byte Character Set translation tables. Double Byte Character Set translation tables for Japan, China and Korea are available for download on the BSI web download page.

Each Double Byte Character Set translation table is located in a library member with a prefix of BSTTDBC. The 1st character of the name specified on the DBCS command is appended to the DBCS member prefix to create the member name. For example, DBCS JAPAN would load the BSTTDBCJ.T member. The member should be placed in a configuration lib.slib (E.g., PRD2.CONFIG).

The DBCS member is accessed through the // LIBDEF SOURCE,SEARCH chain.

## BSTTFTPS FTP Server

Step 2, Bring up the BSTTFTPS FTP Server.

The complete reference for this step can be found in the IPv4 Users Guide and the IPv6 Users Guide manuals.

There can be only one (1) SMNT command in the BSTTFTPS startup. This SMNT command defines the default-filesystem mounted when a FTP client connects to the FTP server. After a connection has been made another filesystem can be mounted by the FTP client if desired.

### Configuration Parameters

Parameter	Description	Sample Value	Your Value
ID	Stack ID	00 (IPv4), 66 (IPv6)	
OPEN	IP address or name	VSE51 (IPv4) ZVSE51 (IPv6)	
SMNT	Default File System	POWER	
LIBDEF	BSI Library	bsilib.slib	

### IPv4 Start Up

```
// LIBDEF *,SEARCH=(PRD2.CONFIG,bsilib.slib)
// OPTION SYSPARM='00'
// EXEC BSTTWAIT,SIZE=BSTTWAIT
/*
// SETPARM LRGBUF=YES
// EXEC BSTTFTPS,SIZE=BSTTFTPS
ID 00
OPEN VSE51
*
SMNT POWER
*
ATTACH SERVER-1
ATTACH SERVER-2
ATTACH SERVER-3
ATTACH SERVER-4
/*
```

## IPv6 Start Up

```
// LIBDEF *,SEARCH=(PRD2.CONFIG,bsilib.slib)
// OPTION SYSPARM='66'
// EXEC BSTTWAIT,SIZE=BSTTWAIT
/*
// SETPARM LRGBUF=YES
// EXEC BSTTFTPS,SIZE=BSTTFTPS
ID 66
OPEN ZVSE51
*
SMNT POWER
*
ATTACH SERVER-1
ATTACH SERVER-2
ATTACH SERVER-3
ATTACH SERVER-4
/*
```

**Note:** The BSTTFTPS FTP server does not listen for both IPv4 and IPv6 connections at the same time. To support both IPv4 and IPv6 connections you must run 2 BSTTFTPS FTP server partitions. The first with the IPv4 stack ID and an IPv4 address specified in the OPEN command. The second with the IPv6 stack ID and an IPv6 address specified in the OPEN command.

You can not run the BSTTFTPS FTP server with an IPv4 stack ID and OPEN an IPv6 address (or with an IPv6 stack ID and OPEN an IPv4 address). If you do, unpredictable results may occur.

## FTP Server Security

After starting the BSTTFTPS FTP server, one of the next items to consider is FTP server security. This is basically a fairly simple and easy process. FTP server security rules are defined in a library member called BSTTSCTY.T.

The default BSTTSCTY.T member allows complete access to the FTP server and z/VSE system by anyone for testing. Naturally, you will want to change this member.

### Default BSTTSCTY.T Member

```
CATALOG BSTTSCTY.T          EOD=/%          REPLACE=YES
*
* FTP-IP ALLOW SUBNET-ADDRESS SUBNET-MASK
* FTP-IP DENY SUBNET-ADDRESS SUBNET-MASK
*
FTP-IP  ALLOW 0.0.0.0 0.0.0.0
FTP-IP6 ALLOW ::0      ::0
*
* FTP-USER USERID PASSWORD  DEFAULT-FILESYSTEM
*
FTP-USER *      *          SMNT-POWER
*
* FTP-ACCESS ALLOW USERID POWER QUEUE CLASSES PREFIX
* FTP-ACCESS DENY  USERID POWER QUEUE CLASSES PREFIX
* FTP-ACCESS ALLOW USERID LIBRARY LIB SLIB MEMBER TYPE
* FTP-ACCESS DENY  USERID LIBRARY LIB SLIB MEMBER TYPE
* FTP-ACCESS ALLOW USERID VSAM CATALOG DATASET
* FTP-ACCESS DENY  USERID VSAM CATALOG DATASET
* FTP-ACCESS ALLOW USERID SAM FILE.NAME
* FTP-ACCESS DENY  USERID SAM FILE.NAME
* FTP-ACCESS ALLOW USERID DLBL DLBL
* FTP-ACCESS DENY  USERID DLBL DLBL
*
FTP-ACCESS ALLOW * * * * *
/%
```

## IPv6/VSE Migration Guide

The 1<sup>st</sup> security rule to change is the last line of the BSTTSCTY.T member. Change the ALLOW verb to DENY. This means that unless a previous rule permits access, the request will be denied by default.

```
FTP-ACCESS DENY * * * * *
```

The 2<sup>nd</sup> security rule to consider changing is the FTP-IP or FTP-IP6 rule. Do you need or want IP address security? If so, you can define the subnet addresses or specific IP addresses using the FTP-IP or FTP-IP6 rules.

```
FTP-IP ALLOW 192.168.1.0 255.255.255.0
FTP-IP ALLOW 10.0.0.0 255.0.0.0
FTP-IP6 ALLOW FD00:806:1::0 /64
```

The 3<sup>rd</sup> security area is the FTP-USER rules. These rules define users to the FTP server. The FTP-USER command specifies the user name, password and optionally an initial file system to mounted when the user logs into the FTP server.

```
FTP-USER JCB JCB SMNT-POWER
FTP-USER TJB TERI SMNT-LIBRARY-PRD2/CONFIG
FTP-USER TONY THIGPEN SMNT-VSAM-VSESP.USER.CATALOG
FTP-USER TST01 TESTING
```

Once users have been defined to the FTP server, access rights need to be defined also. This is done using the FTP-ACCESS command.

```
FTP-ACCESS ALLOW JCB * * * * *
*
FTP-ACCESS ALLOW TJB DLBL * *
FTP-ACCESS ALLOW TJB POWER LST * * *
FTP-ACCESS ALLOW TJB POWER CMD * D *
FTP-ACCESS ALLOW TJB LIBRARY BSILIB INSTALL * A
FTP-ACCESS DENY TJB * * * * *
```

Remember, unless you are permitting full access to a user (E.g., JCB) the last FTP-ACCESS command for a user must be an FTP-ACCESS DENY (E.g., TJB). Also, all FTP-ACCESS commands for a user must be grouped together in the BSTTSCTY.T member.

## Creating a User Defined SAM File System

The BSTTSFSD.T member creates a user defined SAM file system. Defining this file system is optional but can be useful in you access a large number of SAM files from the BSTTFTPS FTP server. Both SAM and VSAM files can be defined in this file system. When a SAM is defined the DLBL of the file is specified. The DLBL must be located in System Standard Labels, Partition Standard Labels or the BSTTFTPS FTP server JCL Labels.

The BLKSZ/RECSZ/RECFM specified for the SAM file must be correct unless the FTP client is using SITE commands to specify the BLKSZ (or BLOCK), RECSZ (or LRECL) and RECFM. If the FTP client is specifying the BLKSZ and RECSZ using SITE commands then specify a BLKSZ 0 RECSZ 0 in the BSTTSFSD.T member for the file. If the RECFM value is specified via a SITE command the value in the SITE command overrides the value specified in the BSTTSFSD.T member.

## Sample BSTTSFSD.T member

```

AC S=PRD2.CONFIG
CATALOG BSTTSFSD.T          EOD=/%          REPLACE=YES
* * * * *
*
*          SAM FILE SYSTEM DEFINITIONS
*
* 1 ENTRY PER LINE
* MUST START IN COLUMN 1
*
* DLBL BLKSZ NNNNN RECSZ NNNNN RECFM XX
* DLBL ESDS
* DLBL KSDS
*
* * * * *
*
FTP1TST BLKSZ 8000 RECSZ    80 RECFM FB
FTP2TST BLKSZ    0 RECSZ    0 RECFM FB
BSTTCAP ESDS
BSOTCAT KSDS
/%
    
```

## View From an FTP Client

```
ftp> quote smnt sam
200 Command OK.
ftp> dir
229 Entering Extended Passive Mode (|||04334|).
150 File status OK, about to open data connection.
drwxr-xr-x  1 0      0      4096 Jan  1 1999 .
drwxr-xr-x  1 0      0      4096 Jan  1 1999 ..
-rwxrw-r-x  1 0      0          0 Nov 19 2012  FTP1.TEST.FILE
-rwxrw-r-x  1 0      0          0 Nov 19 2012  FTP2.TEST.FILE
-rwxrw-r-x  1 0      0          0 Nov 19 2012  IPV6.VSE.PACKET.CAPTURE.FILE
-rwxrw-r-x  1 0      0          0 Nov 19 2012  VSESP.USER.CATALOG
226 Closing data connection.
ftp> bin
200 Command OK.
ftp> get ipv6.vse.packet.capture.file
local: ipv6.vse.packet.capture.file remote: ipv6.vse.packet.capture.file
229 Entering Extended Passive Mode (|||04336|).
150 File status OK, about to open data connection.
250 Requested file action OK, completed.
190396 bytes received in 00:00 (312.46 KB/s)
ftp> quit
221 TCP/IP-TOOLS for VSE terminating connection.
jcb@dv9500t:/tmp>
```

## **AUXDIR Specification**

When the AUXDIR ON command is specified, either as a BSTTFTPS startup command or as a SITE command sent to the FTP server, directory list output for the VSE/POWER queues is shown in the CSI format. The BSI format is class.number.name while the CSI format is class.name.number.segment.

The AUXDIR ON command also enables emulation of filenames specified on STOR/RETR commands sent to the FTP server. For example, STOR name.RDR.class causes the POWER file system to be mounted and the directory set to the RDR queue before transferring data to POWER. Also, a filename on a STOR/RETR command is combined with any directory specification sent to the FTP server using a CWD command. For example, CD /FTP1/TEST followed by PUT FILE results in the SAM file system being mounted and the file FTP1.TEST.FILE transferred.

## **Sample FTP Client Log**

```
ftp> site auxdir on
200 Command OK.
ftp> bin
200 Command OK.
ftp> site block 800
200 Command OK.
ftp> site lrecl 80
200 Command OK.
ftp> cd /ftp2/test
250 Requested file action OK, completed.
ftp> get file ftp2.test.file
local: ftp2.test.file remote: file
229 Entering Extended Passive Mode (|||04432|).
150 File status OK, about to open data connection.
250 Requested file action OK, completed.
159992000 bytes received in 00:40 (3.76 MB/s)
ftp> asc
200 Command OK.
ftp> put vsejcl.job sysout.rdr.p
local: vsejcl.job remote: sysout.rdr.p
229 Entering Extended Passive Mode (|||04434|).
150 File status OK, about to open data connection.
100% |*****|
155      1.22 MB/s      --:-- ETA
250 Requested file action OK, completed.
155 bytes sent in 00:00 (0.38 KB/s)
ftp> quit
```



## ***Configuring an Emulation FTP Server***

The following steps can be used to create and startup an emulation mode FTP server using BSTTFTPS. An emulation mode FTP server is useful when remote host access the FTP server using scripts. For casual or adhoc FTP client access use a standard mode BSTTFTPS FTP server partition.

1. Add AUXDIR ON to the standard BSTTFTPS FTP server JCL

When AUXDIR ON is specified in the BSTTFTPS FTP server JCL, all FTP client sessions begin in emulation mode by default.

2. Review your IPINITnn.L startup member and create a BSTTSFSD.T User Defined SAM file system for the BSTTFTPS FTP Server to use.

For example,

```
DEFINE FILE,TYPE=SAM,DLBL=FTPLB09,PUBLIC='DEV.PWU30710.IPROD'  
will result in a BSTTSFSD.T line of  
FTPLB09 BLKSZ 0 RECSZ 0 RECFM FB
```

The FTP client script might contain lines like this ...

```
site lrecl 80  
site block 80  
cd /DEV/PWU30710  
put IPROD
```

Or,

```
DEFINE FILE,TYPE=ESDS,DLBL=FTPLB07,PUBLIC='PRD.CENTRAL.DEPOSIT.ESDS',  
LRECL=171  
will result in a BSTTSFSD.T line of  
FTPLB07 ESDS RECSZ 171
```

The FTP client script might contain lines like this ...

```
ascii  
cd /PRD/CENTRAL  
put DEPOSIT.ESDS
```

3. Remote host FTP client scripts accessing the emulation FTP server should now work unchanged.

## Configuring Multiple BSTTFTPS FTP Server Partitions

The BSTTPRXY Proxy Server supports up to 8 PRXY statement per partition. This allows BSTTPRXY to proxy connections to up to 8 BSTTFTPS partitions. Since each BSTTFTPS partition allows up to 4 FTP server subtasks, using the BSTTPRXY to proxy FTP connection allow you to run up to 32 (8 x 4) FTP server subtasks per z/VSE image.

### Sample BSTTPRXY Configuration

This example shows how to configure the BSTTPRXY to proxy connections to 3 BSTTFTPS partitions. Users connect to the BSTTPRXY proxy server partition on the standard FTP port 21. BSTTPRXY proxies the connection to one of the available BSTTFTPS partitions.

```
// OPTION SYSPARM='00'
// SETPARM LRGBUF=YES
// SETPARM IPTRACE='NNNNNNNN'
// SETPARM SSL$DBG='NO ' OR 'NO' (DEFAULT = NO)
// SETPARM SSL$ICA='YES' OR 'NO' (DEFAULT = YES)
// LIBDEF *,SEARCH=(PRD2.CONFIG,BSILIB.TTDEV)
// SETPARM SUBTASK=8
// UPSI 000
// EXEC BSTTPRXY,SIZE=BSTTPRXY,PARM='TRAP(OFF) /'
ID 66
*
KEYRING PRD2.CONFIG
KEYFILE ZVSE62
SECTYPE TLV1
*
OPTION FTP
OPTION SERVER
*
PROXY TCP V4 21 TXT * TO V4 1021 TXT * VSE51B
PROXY TCP V4 21 TXT * TO V4 2021 TXT * VSE51B
PROXY TCP V4 21 TXT * TO V4 3021 TXT * VSE51B
/*
```

## Sample BSTTFTPS Configuration

```
* BSTTFTPS FTP Server #1
// SETPARM LRGBUF=YES
// EXEC BSTTFTPS,SIZE=BSTTFTPS
ID 00
OPEN VSE51B 1021
*
SMNT POWER
*
ATTACH SERVER-1
ATTACH SERVER-2
ATTACH SERVER-3
/*

* BSTTFTPS FTP Server #2
// SETPARM LRGBUF=YES
// EXEC BSTTFTPS,SIZE=BSTTFTPS
ID 00
OPEN VSE51B 2021
*
SMNT POWER
*
ATTACH SERVER-1
ATTACH SERVER-2
ATTACH SERVER-3
/*

* BSTTFTPS FTP Server #3
// SETPARM LRGBUF=YES
// EXEC BSTTFTPS,SIZE=BSTTFTPS
ID 00
OPEN VSE51B 3021
*
SMNT POWER
*
ATTACH SERVER-1
ATTACH SERVER-2
ATTACH SERVER-3
/*
```

## BSTTVNET TN3270E Server

The BSTTVNET TN3270E server is a TN3270E server application that supports TN3270, TN3270E terminal sessions, TN3270E printer sessions (both non-SNA and SNA) plus DIRECT, LPR and FTP printer sessions. DIRECT printer session print directly to a network printer and do not require a printer client (or server) application at the remote printer. LPR printers sessions are equivalent to TCP/IP for VSE GPS printers.

The complete reference for this step can be found in the IPv4 Users Guide and the IPv6 Users Guide manuals.

### **Configuration Parameters**

Parameter	Description	Sample Value	Your Value
ID	Stack ID	00 (IPv4), 66 (IPv6)	
OPEN	IP address or name	VSE51 (IPv4) ZVSE51 (IPv6)	
VTAM	.B book	VNETAPPL.B	
LIBDEF	BSI Library	bsilib.slib	

## VTAM Definitions

There are two VTAM .B books to consider. The 1<sup>st</sup> VTAM .B book contains the APPL definitions required by the BSTTVNET TN3270E server. The 2<sup>nd</sup> VTAM .B book is ATCCON00.B. This .B book is updated to activate the APPL definitions used by BSTTVNET during VTAM initialization.

## CICS Terminal Types

If you are using terminal or printer definitions that are predefined in the CICS RDO file, verify the CICS TYPETERM used for these sessions is correct. The CICS TYPETERM must be one of the values specified in the IPv6/VSE Users Guide.

## VNETAPPL.B

This VNETAPPL.B book shows the basic APPL statements used by the BSTTVNET sample start up JCL and commands shown later in the section.

```
CATALOG VNETAPPL.B      REPLACE=YES
VNETAPPL VBUILD TYPE=APPL
BSTTVNET APPL
BSTTUSST APPL AUTH=(PASS,ACQ)
VNETTRM  GROUP MODETAB=IESINCLM,DLOGMOD=SP3272QN
T001     APPL AUTH=(ACQ),EAS=1
T002     APPL AUTH=(ACQ),EAS=1
T003     APPL AUTH=(ACQ),EAS=1
T004     APPL AUTH=(ACQ),EAS=1
T005     APPL AUTH=(ACQ),EAS=1
T006     APPL AUTH=(ACQ),EAS=1
T007     APPL AUTH=(ACQ),EAS=1
T008     APPL AUTH=(ACQ),EAS=1
T009     APPL AUTH=(ACQ),EAS=1
T010     APPL AUTH=(ACQ),EAS=1
T999     APPL AUTH=(ACQ),EAS=1
```

## ATCCON00.B

This ATCON00.B book shows the change needed to activate the VNETAPPL.B book during VTAM start up.

```
CATALOG ATCCON00 .B          EOD=XY          REPLACE=YES
      VTMAPPL ,
      TCPAPP00 ,
      VTMSNA ,
      VTMSNSNA ,
      VTMCTCA ,
      VTMPATH ,
      VTMCA1 ,
      VTMCA2 ,
      VTMCA3 ,
      VTMCDRM ,
      VTMCDRS ,
      VNETAPPL ,
      VTMSW1
```

XY

## ***BSTTVNET LUNAME allocation***

BSTTVNET supports several methods for allocating VTAM LUNAMES. Both TN3270 and TN3270E terminal and printer sessions are supported.

All printer sessions are TN3270E (the TN3270 protocol does not support printers). Therefore, when a TN3270E client is configured for a PRINTER session an LU/Resource name is specified as part of the configuration. The LU/Resource name is the name of the VTAM LU to be used for the session.

TERMINAL sessions can be either TN3270 or TN3270E. The old TN3270 protocol does not support the specification of an LU/Resource name when a connection is configured. This type of connection will result in either a GENERIC LUNAME being selected (#1) or the IPv4/IPv6 address can be used to select a SPECIFIC LUNAME (#4).

When you configure a TN3270E client there is a LU/Resource name that is part of the specification. This LU/Resource name field can be used to specifically request a VTAM LUNAME (#3) or an VTAM LUNAME that is part of a pool of GENERIC LUNAMES (#2).

### **#1. TERMINAL luname GENERIC**

This type of luname selection is first-come, first-served (random) allocation of an luname.

### **#2, TERMINAL luname GENERIC \* \* POOL name**

This type of LUNAME selection is first-come, first-served (random) base on the POOL name provided by the TN3270E client.

### **#3, TERMINAL luname GENERIC \* \* PORT number**

This type of LUNAME selection is first-come, first-served (random) base on the port number used for the connection by the TN3270E client.

### **#4, TERMINAL luname SPECIFIC**

This type of luname selection is specific. This means that the TN3270E client running on the PC will ask for this luname specifically.

### **#5, TERMINAL luname SPECIFIC \* \* IP 10.200.11.111 255.255.255.255**

This type of LUNAME selection is done based on the IPv4/IPv6 address making the connection. In the above example, this luname would be allocated to a session using the IP address of 10.200.11.111

## IPv6/VSE Migration Guide

A specification of ...

```
DEFINE TELNETD, ID=P1, TERMNAME=P1, PORT=23, BASE=45, -  
COUNT=04, POOL=YES, TARGET=JABAPRD2, IPADDR=10.200.11.111
```

Would look like this in BSTTVNET ...

```
TERMINAL P145 SPECIFIC BSTTUSST DEDICATE IP 10.200.11.111 255.255.255.255  
TERMINAL P146 SPECIFIC BSTTUSST DEDICATE IP 10.200.11.111 255.255.255.255  
TERMINAL P147 SPECIFIC BSTTUSST DEDICATE IP 10.200.11.111 255.255.255.255  
TERMINAL P148 SPECIFIC BSTTUSST DEDICATE IP 10.200.11.111 255.255.255.255
```

The VTAM logmode used for BSTTVNET session is specified in the VTAM .B book where the LUNAMES are defined.

E.g.,

```
VNETTRM GROUP MODETAB=IESINCLM, DLOGMOD=SP3272QN  
T001  APPL AUTH=(ACQ), EAS=1  
T002  APPL AUTH=(ACQ), EAS=1
```

We recommend using SP3272QN but any local 3270 non-SNA logmode can be used. Using SP3272QN tells the application (CICS) to query the TN3270(E) client to determine the type of terminal to use (mod2/mod3/mod4/etc.).



## ***Multiple Port Numbers***

Each BSTTVNET TN3270E server partition listens on up to 16 ports. Each port is specified on a OPEN command.

### ***\* \$\$ SLI Usage***

You may find that placing all of your BSTTVNET start up command in a z/VSE library member or ICCF member is nice. This make editing them simple and easy.

You BSTTVNET JCL would then look like this ...

```
// EXEC BSTTVNET,SIZE=BSTTVNET,OS390,DSPACE=3M  
* $$ SLI ...  
/*
```

When using this method, updates are made to the ICCF/library member but no changes to the BSTTVNET JCL are needed.

## **Menu Systems**

We have 2 menu systems.

#1 is the standard menu system.

#2 is the VTAM USSTAB emulation menu system.

With the standard menu system is simple to setup and use. However, you can only customize the title and descriptions. This is done by modifying the TITLE and APPLID commands in the BSTTVNET startup.

With the VTAM USSTAB emulation menu system you get much more control over how the menu appears on the screen. This menu can have variable fields.

```
@@LUNAME = 8 character VTAM LUNAME
@@CURDAT = 8 character date
@@CURTIM = 8 character time
@@IPADDR = 40 character IPv6 address of connection source
           IPv4 addresses are shown as IPv4-mapped-IPv6 addresses
```

The @ characters must be coded using the X'7C' byte value. So, some users may have to code the USSTAB variable fields as X'7C7C',C'IPADDR', etc.

For each BSTTVNET partition you can have a standard menu and multiple VTAM USSTAB menus. For example, operations and systems personal might use the standard menu giving access to a broad list of applications while general users see a more limited VTAM menu. The *IPv6/VSE IPv4 Users Guide* has an example of running multiple USSTAB menus in a BSTTVNET partition.

## **IPv4/IPv6 Start Up**

The is sample BSTTVNET start up JCL and commands for both IPv4 and IPv6 sessions.

The COUPLED ON command tells the BSTTVNET server to accept connection on both the IPv4 (BSTTINET) and IPv6 (BSTT6NET) TCP/IP stacks.

```
// OPTION LOG, PARTDUMP, NOSYS DUMP
// LIBDEF PHASE, SEARCH=(bsilib.slib)
// LIBDEF SOURCE, SEARCH=(PRD2.CONFIG,bsilib.slib)
// OPTION SYSPARM='66'
// EXEC BSTTWAIT, SIZE=BSTTWAIT
/*
// EXEC BSTTVNET, SIZE=BSTTVNET, DSPACE=4M, TASKS=ANY
ID 66
OPEN ZVSE51 23
COUPLED ON
*
APPLID DBDCCICS CICS/TS AND ICCF
APPLID BSTTVNET PRINTER SHARING APPLICATION
*
TITLE *** ** Welcome to Barnard Software, Inc. *** **
*
TERMINAL T001 GENERIC
TERMINAL T002 GENERIC
TERMINAL T003 GENERIC
TERMINAL T004 GENERIC
TERMINAL T005 GENERIC
TERMINAL T006 GENERIC
TERMINAL T007 GENERIC
TERMINAL T008 GENERIC
TERMINAL T009 GENERIC
TERMINAL T010 GENERIC
*
ATTACH TN3270E
/*
```

You can now configure your PC TN3270(E) client to connect to BSTTVNET on port 23.

One nice tip for BSTTVNET is using an \* \$\$ SLI to contain the configuration statements. This allows the BSTTVNET configuration to be updated without changing the JCL in the POWER RDR queue.

## BSTTFTPC Batch FTP

The BSTTFTPC batch FTP client application is an easy to use utility. Most batch FTP jobs are similar to each other. So, once you have one BSTTFTPC job ready to go and tested, most other batch FTP jobs will be quite similar.

A common request from customers is to have the date and/or time included in the filename used by the BSTTFTPC job. This is pretty simple. We use a REXX EXEC for this and there is an example of how to do this in the IPv6/VSE IPv4 Users Guide. See “Invoking BSTTFTPC from REXX”.

The complete reference for this step can be found in the IPv4 Users Guide and the IPv6 Users Guide manuals.

### *Configuration Parameters*

Parameter	Description	Sample Value	Your Value
ID	Stack ID	00 (IPv4), 66 (IPv6)	
OPEN	IP address or name	VSE51 (IPv4) ZVSE51 (IPv6)	
User	FTP Userid	jcb	
Password	FTP Password	password	
Filename	FTP file name	bsttfil.txt	
Directory	FTP Directory	my/dir2	
LIBDEF	BSI Library	bsilib.slib	

You will see in the sample JCL provided that the only real difference between the IPv4 Sample and the IPv6 Sample is the Stack ID used and the name of the Remote Host.

## IPv4 Sample JCL

```
// LIBDEF PHASE,SEARCH=(bsilib.slib)
// LIBDEF SOURCE,SEARCH=(PRD2.CONFIG,bsilib.slib)
// DLBL BSTTFIL,'FTP.TEST.FILE',,VSAM,CAT=VSESPUC
// EXEC BSTTFTPC,SIZE=BSTTFTPC
ID 00
OPEN VSE51
USER jcb
PASS password
*
CWD my/dir2
INPUT VSAM BSTTFIL
TYPE A
STOR bsttfil.txt
*
QUIT
/*
```

## IPv6 Sample JCL

```
// LIBDEF PHASE,SEARCH=(bsilib.slib)
// LIBDEF SOURCE,SEARCH=(PRD2.CONFIG,bsilib.slib)
// DLBL BSTTFIL,'FTP.TEST.FILE',,VSAM,CAT=VSESPUC
// EXEC BSTTFTPC,SIZE=BSTTFTPC
ID 66
OPEN ZVSE51
USER jcb
PASS password
*
CWD my/dir2
INPUT VSAM BSTTFIL
TYPE A
STOR bsttfil.txt
*
QUIT
/*
```

## **Converting FTPBAT~~CH~~ to BSTTFTPC**

The REXX BSTTC2BF.PROC converts CSI FTPBAT~~CH~~ commands to BSTTFTPC commands. Once the commands have been converted, if no errors are detected, BSTTFTPC is invoked. If errors are detected, messages are displayed upon the console and FTPBAT~~CH~~ is re-invoked to complete the data transfer. The BSTTC2BF.PROC can be automatically invoked (instead of FTPBAT~~CH~~) by enabling the provided \$JOBEXIT routine.

## **Edit the BSTTC2BF.PROC**

Look for the line that define the default stack ID. Change these values to the values used on your system can re-catalog the BSTTC2BF.PROC.

```
_id = '01'           /* default CSI stack id */
_bsi_id = '00'      /* BSI stack ID */
```

## **Rename BSTTC2BF.PROC**

Use LIBR to rename the BSTTC2BF.PROC to C2B.PROC. You also may wish to copy the C2B.PROC to a lib.slib present in your LIBDEF PROC,SEARCH chain.

```
// EXEC LIBR,SIZE=256K,PARM='MSHP'
CONNECT S=bsilib.slib:PRD2.CONFIG
COPY BSTTC2BF.PROC:C2B.PROC REPLACE=YES
/*
```

## Enable the BSTTJXIT \$JOBEXIT routine

The IBM z/VSE Guide to System Functions SC33-8312-00 contains instructions for enabling a \$JOBEXIT routine. In the simplest case where you do not have a \$JOBEXIT routine present on your system, you can rename the BSTTJXIT.PHASE to \$JOBEXIT.PHASE and update your BG ASI PROC to load the new \$JOBEXIT routine at IPL time.

### \$JOBEXIT Phases

Name	Usage
BSTTJXIT.PHASE	Convert FTPBATCH and CLIENT/EMAIL
BSTTJXI1.PHASE	Convert FTPBATCH
BSTTJXI2.PHASE	Convert CLIENT/EMAIL

### LIBR Activation JCL

```
// EXEC LIBR,SIZE=256K,PARM='MSHP'
CONNECT S=bsilib.slib:IJSYSRS.SYSLIB
COPY BSTTJXIT.PHASE:$JOBEXIT.PHASE REPLACE=YES
/*
SET SDL
$JOBEXIT,SVA
/*
```

If you already have a \$JOBEXIT routine present on your system, you will need to follow the directions found in the IBM z/VSE Guide to System Functions SC33-8312-00 for enabling multiple \$JOBEXIT routines on a system.

## Sample JCL

```
// EXEC FTPBATCH,SIZE=1M
LOPEN
LUSER JCB
LPASS BSI
OPEN 192.168.1.60
USER jcb
PASS bsi
CD /tmp
PUT %BSAULOG,ESDS opti.audit.log.file
QUIT
/*
```

## Output

```
// EXEC REXX=BSTTC2BF,SIZE=1M
Old stack ID 01
New stack ID 00
Card 1 LOPEN
Card 2 LUSER JCB
Card 3 LPASS BSI
Card 4 OPEN 192.168.1.60
Card 5 USER jcb
Card 6 PASS bsi
Card 7 CD /tmp
Card 8 PUT %BSAULOG,ESDS opti.audit.log.file
Card 9 QUIT

BSTTFTPC commands ...
ID 00
LUSER JCB
LPASS BSI
OPEN 192.168.1.60
USER jcb
PASS bsi
CWD /tmp
INPUT VSAM BSAULOG
STOR opti.audit.log.file
QUIT
Invoking BSTTFTPC ...
```





## ***Auto-FTP Processing***

The BSTTAFTP.PROC is the Auto-FTP REXX EXEC provided for IPv6/VSE. This REXX EXEC is a long running REXX application. The BSTTAFTP REXX EXEC runs in a partition monitoring the POWER LST queue for a specific class (Class X by default). When a entry appears in the POWER LST queue in Class X, BSTTAFTP changes the POWER Class from X to Y and submits a BSTTFTPC job to execute in a different partition. This BSTTFTPC job will transfer the Class Y member to an FTP server. Finally, the Class Y member is changed from DISP=K to DISP=L upon successful transfer completion.

Since BSTTAFTP is a REXX EXEC you can modify this REXX EXEC to use different POWER classes or to handle any other processing requirements.

## **Sample JCL**

```
// LIBDEF PROC,SEARCH=bsilib.slib  
// EXEC REXX=BSTTAFTP  
/*
```

## **CSI Auto-FTP Script Emulation**

The REXX BSTTECSI.PROC, provided by Rich Smrcina, provides CSI Auto-FTP script emulation. This REXX EXEC can be modified to perform CSI Auto-LPR script processing also.

```
/* REXXAFTP:  This program replace the AutoFTP function of CSI's
              TCP/IP.

IMPORTANT:
This program is in the Public Domain and is distributed "AS IS".
There is no guarantee of support for this program either
expressed or implied, either by the author or Grede Foundries, Inc.
The author or Grede Foundries, Inc. will incur no responsibility
for any damages due to the use of this program.

Written by: Rich Smrcina  Grede Foundries, Inc.
Release 1: 12/12/2001  Initial Release.

Rich Smrcina ... rsmrcina@wi.rr.com

*/
```

The BSTTECSI.PROC is a long running REXX EXEC that monitors POWER LST queue class F.

## **Sample JCL**

```
// LIBDEF PROC,SEARCH=bsilib.slib
// EXEC REXX=BSTTECSI
/*
```

## BSTTLPRC LPR Client

The BSTTMTPC batch EMAIL client application is an easy to use utility. Most batch EMAIL jobs are similar to each other. So, once you have one BSTTMTPC job ready to go and tested, most other batch EMAIL jobs will be quite similar.

The complete reference for this step can be found in the IPv4 Users Guide and the IPv6 Users Guide manuals.

### *Configuration Parameters*

Parameter	Description	Sample Value	Your Value
ID	Stack ID	00 (IPv4), 66 (IPv6)	
OPEN	IP address or name	192.168.1.60	
QUEUE	Queue Name	epon	
HOST	Host Name (required)	z/VSE	
USER	User Name (required)	POWER	
LIBDEF	BSI Library	bsilib.slib	

You will see in the sample JCL provided that the only real difference between the IPv4 Sample and the IPv6 Sample is the Stack ID used and the name of the Remote Host.

## IPv4 Sample JCL

```
// LIBDEF PHASE,SEARCH=(bsilib.slib)
// LIBDEF SOURCE,SEARCH=(PRD2.CONFIG,bsilib.slib)
// EXEC BSTTLPRC,SIZE=BSTTLPRC
ID 00
OPEN 192.168.1.60
*
INPUT POWER LST VSAMINIT 014 A SYSA
QUEUE epson
USER POWER
HOST z/VSE
*
SEND
*
QUIT
/*
```

## IPv6 Sample JCL

```
// LIBDEF PHASE,SEARCH=(bsilib.slib)
// LIBDEF SOURCE,SEARCH=(PRD2.CONFIG,bsilib.slib)
// EXEC BSTTLPRC,SIZE=BSTTLPRC
ID 66
OPEN FD00:806:1::3
*
INPUT POWER LST VSAMINIT 014 A SYSA
QUEUE epson
USER POWER
HOST z/VSE
*
SEND
*
QUIT
/*
```

### ***BSTTLPRC Auto-LPR Processing***

The BSTTALPR.PROC is the Auto-LPR REXX EXEC provided for IPv6/VSE. This REXX EXEC is a long running REXX application. The BSTTALPR REXX EXEC runs in a partition monitoring the POWER LST queue for a specific class (Class L by default). When a entry appears in the POWER LST queue in Class L, BSTTALPR changes the POWER Class from L to M and submits a BSTTLPRC job to execute in a different partition. This BSTTLPRC job will transfer the Class M member to an LPD. Finally, the Class M member is changed from DISP=K to DISP=L upon successful transfer completion.

Since BSTTALPR is a REXX EXEC you can modify this REXX EXEC to use different POWER classes or to handle any other processing requirements.

### **Sample JCL**

```
// LIBDEF PROC,SEARCH=bsilib.slib  
// EXEC REXX=BSTTALPR  
/*
```

## BSTTMTPC Batch EMAIL Client

The BSTTMTPC batch EMAIL client application is an easy to use utility. Most batch EMAIL jobs are similar to each other. So, once you have one BSTTMTPC job ready to go and tested, most other batch EMAIL jobs will be quite similar.

The complete reference for this step can be found in the IPv4 Users Guide and the IPv6 Users Guide manuals.

### *Configuration Parameters*

Parameter	Description	Sample Value	Your Value
ID	Stack ID	00 (IPv4), 66 (IPv6)	
OPEN	IP address or name	SMTP-SERVER.CFL.RR.COM	
EHLO	Domain	bsiopti.com	
AUTH LOGIN	Userid	userid	
AUTH LOGIN	Password	password	
LIBDEF	BSI Library	bsilib.slib	

You will see in the sample JCL provided that the only real difference between the IPv4 Sample and the IPv6 Sample is the Stack ID used and the name of the Remote Host.

## IPv4 Sample JCL

```
// LIBDEF PHASE,SEARCH=(bsilib.slib)
// LIBDEF SOURCE,SEARCH=(PRD2.CONFIG,bsilib.slib)
// EXEC BSTTMTPC,SIZE=BSTTMTPC
ID 00
OPEN SMTTP-SERVER.CFL.RR.COM
*
EHLO bsiopti.com
AUTH LOGIN userid password
MAIL From: bsiopti@bsiopti.com
RCPT To: jeff@bsiopti.com
SUBJ Subject: Requested Report
ORGA Organization: Barnard Software, Inc.
*
DATA
*
INPUT POWER LST BSTTCLIB 243 Q COMPILE
ASA OFF
TYPE A
INCLUDE bsttclib.txt
*
QUIT
/*
The report you requested is attached.

Regards,
Your IT Staff
/*
```



## IPv6 Sample JCL

```
// LIBDEF PHASE,SEARCH=(bsilib.slib)
// LIBDEF SOURCE,SEARCH=(PRD2.CONFIG,bsilib.slib)
// EXEC BSTTMTPC,SIZE=BSTTMTPC
ID 66
OPEN IPV6-SMTP-SERVER.CFL.RR.COM
*
EHLO bsiopti.com
AUTH LOGIN userid password
MAIL From: bsiopti@bsiopti.com
RCPT To: jeff@bsiopti.com
SUBJ Subject: Requested Report
ORGA Organization: Barnard Software, Inc.
*
DATA
*
INPUT POWER LST BSTTCLIB 243 Q COMPILE
ASA OFF
TYPE A
INCLUDE bsttclib.txt
*
QUIT
/*
The report you requested is attached.

Regards,
IT Staff
/*
```

## **Converting CLIENT/EMAIL to BSTTMTPC**

The REXX BSTTC2BE.PROC converts CSI CLIENT/EMAIL commands to BSTTMTPC commands. Once the commands have been converted, if no errors are detected, BSTTMTPC is invoked using the BSIREXXC.PROC REXX EXEC. If errors are detected, messages are displayed upon the console and CLIENT/EMAIL is re-invoked to complete the data transfer. The BSTTC2BEPROC can be automatically invoked (instead of CLIENT/EMAIL) by enabling the provided \$JOBEXIT routine.

The BSTTC2BE.PROC using the BSIREXXC.PROC REXX EXEC to invoke BSTTMTPC. In effect, the BSTTC2BE.PROC is frontending the BSIREXXC.PROC.

*Please refer to the 'BSIREXXC Symbolic Variable Support' section and install/enable this feature also.*

## **Edit the BSTTC2BE.PROC**

Look for the line that define the default stack ID. Change these values to the values used on your system can re-catalog the BSTTC2BE.PROC.

```
_id = '01'                /* default CSI stack id */
_bsi_id = '00'           /* BSI stack ID */
```

## **Rename BSTTC2BF.PROC**

Use LIBR to copy the BSTTC2BE.PROC.

```
// EXEC LIBR,SIZE=256K,PARM='MSHP'
CONNECT S=bsilib.slib:PRD2.CONFIG
COPY BSTTC2BF.PROC REPLACE=YES
/*
```

## Enable the BSTTJXIT \$JOBEXIT routine

The IBM z/VSE Guide to System Functions SC33-8312-00 contains instructions for enabling a \$JOBEXIT routine. In the simplest case where you do not have a \$JOBEXIT routine present on your system, you can rename the BSTTJXIT.PHASE to \$JOBEXIT.PHASE and update your BG ASI PROC to load the new \$JOBEXIT routine at IPL time.

### \$JOBEXIT Phases

Name	Usage
BSTTJXIT.PHASE	Convert FTPBATCH and CLIENT/EMAIL
BSTTJXI1.PHASE	Convert FTPBATCH
BSTTJXI2.PHASE	Convert CLIENT/EMAIL

### LIBR Activation JCL

```
// EXEC LIBR,SIZE=256K,PARM='MSHP'
CONNECT S=bsilib.slib:IJSYSRS.SYSLIB
COPY BSTTJXIT.PHASE:$JOBEXIT.PHASE REPLACE=YES
/*
SET SDL
$JOBEXIT,SVA
/*
```

If you already have a \$JOBEXIT routine present on your system, you will need to follow the directions found in the IBM z/VSE Guide to System Functions SC33-8312-00 for enabling multiple \$JOBEXIT routines on a system.

## Sample JCL

```
// OPTION PARTDUMP,NOSYSDUMP
// LIBDEF *,SEARCH=(PRD2.CONFIG,BSILIB.TTDEV)
// SETPARM DEST='JEFF'
// EXEC CLIENT,PARM='CLIENT=EMAIL,ID=00'
SET HOST=SMTP-SERVER.CFL.RR.COM
SET FROM=vse26@bsiopti.com
SET TO=&DEST@bsiopti.com
SET COPY=bsiopti@bsiopti.com
SET SUBJECT=NOS160E0 - NOS COMMERCIAL INVOICE
TEXT
DELIMITED FILE
/+
CD IJSYSUC
ATTACH OPTI.AUDIT.LOG.FILE AS BSAUFIL
SEND
QUIT
/*
```

## Output

```
// SETPARM DEST='JEFF'  
// EXEC REXX=BSTTC2BE,PARM='CLIENT=EMAIL, ID=00'  
// DLBL IFILE,'OPTI.AUDIT.LOG.FILE',,VSAM,CAT=IJSYSUC  
/*  
... @DEST set to 'JEFF'  
Old stack ID 00  
New stack ID 00  
Card 1 SET HOST=SMTP-SERVER.CFL.RR.COM  
Card 2 SET FROM=vse26@bsiopti.com  
Card 3 SET TO=&DEST@bsiopti.com  
Card 4 SET COPY=bsiopti@bsiopti.com  
Card 5 SET SUBJECT=NOS160E0 - NOS COMMERCIAL INVOICE  
Card 6 TEXT  
Card 9 CD IJSYSUC  
Card 10 ATTACH OPTI.AUDIT.LOG.FILE AS BSAUFIL  
Card 11 SEND  
Card 12 QUIT  
  
BSIREXXC commands ...  
ID 00  
OPEN SMTP-SERVER.CFL.RR.COM  
MAIL From: vse26@bsiopti.com  
RCPT To: JEFF@bsiopti.com  
RCPT CC: bsiopti@bsiopti.com  
SUBJ Subject: NOS160E0 - NOS COMMERCIAL INVOICE  
DATA  
INPUT VSAM IFILE  
INCLUDE BSAUFIL  
QUIT  
/?  
DELIMITED FILE  
  
Invoking BSIREXXC ...
```

## BSTTREXC Batch Remote Execution Client

The BSTTREXC batch Remote EXEC client application is an easy to use utility. Most batch Remote EXEC jobs are similar to each other. So, once you have one BSTTREXC job ready to go and tested, most other batch Remote EXEC jobs will be quite similar.

The complete reference for this step can be found in the IPv4 Users Guide and the IPv6 Users Guide manuals.

### Configuration Parameters

Parameter	Description	Sample Value	Your Value
ID	Stack ID	00 (IPv4), 66 (IPv6)	
OPEN	IP address or name	10.1.1.1	
USER	Userid	userid	
PASS	Password	password	
LIBDEF	BSI Library	bsilib.slib	

You will see in the sample JCL provided that the only real difference between the IPv4 Sample and the IPv6 Sample is the Stack ID used and the name of the Remote Host.

### Linux System Considerations

When using BSTTREXC to execute scripts on Linux systems, the EXEC Daemon (server) must be enabled and the /etc/hosts file must contain an entry for the z/VSE system's IP address. The EXEC Daemon does a reverse DNS lookup of the entries in the /etc/hosts file to verify the IP address of the z/VSE Remote EXEC client.

## IPv4 Sample JCL

```
// LIBDEF PHASE,SEARCH=(bsilib.slib)
// LIBDEF SOURCE,SEARCH=(PRD2.CONFIG,bsilib.slib)
// EXEC BSTTREXC,SIZE=BSTTREXC
ID 00
OPEN 10.1.1.1
*
USER userid
PASS password
CODE *
*
EXEC sh -x rxtest.sh
*
QUIT
/*
```

## IPv6 Sample JCL

```
// LIBDEF PHASE,SEARCH=(bsilib.slib)
// LIBDEF SOURCE,SEARCH=(PRD2.CONFIG,bsilib.slib)
// EXEC BSTTREXC,SIZE=BSTTREXC
ID 66
OPEN FD00:806:1::3
*
USER userid
PASS password
CODE *
*
EXEC sh -x rxtest.sh
*
QUIT
/*
```

## ***Replacing Batch Telnet***

The CSI batch Telnet application is often used to open a connection to a remote host and execute one (or a few) commands. The BSI BSTTREXC batch Remote EXEC Client can easily be used to replace a batch Telnet execution.

## **Sample Batch TELNET**

```
// EXEC TELNET, PARM='IP=172.17.0.209'  
userid  
password  
nohup /exe_prd/pigwig/bin/upload_billing_final &  
/*
```

## **Replacement BSTTREXC Job**

```
// EXEC BSTTREXC, SIZE=BSTTREXC  
ID 10  
OPEN 172.17.0.209  
USER userid  
PASS password  
CODE *  
EXEC nohup /exe_prd/pigwig/bin/upload_billing_final &  
QUIT  
/*
```



## IPv6/VSE Migration Guide

Often, the batch Telnet client is used to simply delete a file located on a remote host. In this case a simple BSTTFTPC batch FTP client job can be used too.

### Sample Batch Telnet File Delete

```
// EXEC TELNET, PARM='IP=172.17.0.209'  
userid  
password  
rm /exe_prd/pigwig/bin/PWU30009.in.progress  
/*
```

### Sample BSTTFTPC File Delete

```
// EXEC BSTTFTPC, SIZE=BSTTFTPC  
ID 10  
OPEN 172.17.0.209  
USER userid  
PASS password  
*  
DELE /exe_prd/pigwig/bin/PWU30009.in.progress  
*  
QUIT  
/*
```

## BSTTTELN Batch Telnet Client

IPv6/VSE has a batch telnet utility. This utility can be used to establish a connection with a remote host telnet daemon and send basic commands.

While the BSTTTELN utility is useful we strongly recommend using the BSTTREXC Remote EXEC Client utility instead of BSTTTELN. The BSTTREXC utility will wait until a command has completed and set the z/VSE return code based upon the completion. The BSTTTELN utility does not do this.

At this time the BSTTTELN utility has been tested with the Linux and IBM AIX telnet daemons.

The complete reference for this step can be found in the IPv4 Users Guide and the IPv6 Users Guide manuals.

### *Configuration Parameters*

Parameter	Description	Sample Value	Your Value
ID	Stack ID	00 (IPv4), 66 (IPv6)	
OPEN	IP address or name	10.1.1.1	
USER	Userid	userid	
PASS	Password	password	
LIBDEF	BSI Library	bsilib.slib	

You will see in the sample JCL provided that the only real difference between the IPv4 Sample and the IPv6 Sample is the Stack ID used and the name of the Remote Host.

## Sample Batch TELNET

```
// EXEC TELNET, PARM='IP=172.17.0.209'  
userid  
password  
nohup /exe_prd/pigwig/bin/upload_billing_final &  
/*
```

## Replacement BSTTTELN Job

```
// EXEC BSTTTELC, SIZE=BSTTTELC  
ID 00  
OPEN 172.17.0.209  
*  
USER userid  
PASS password  
*  
CMD nohup /exe_prd/pigwig/bin/upload_billing_final &  
*  
SEND  
QUIT  
/*
```

## BSTTLPDS Line Print Daemon Service

IPv6/VSE has a Line Print Daemon (LPD) Service (BSTTLPDS). This service makes z/VSE appear to be a printer on the network. Multiple print queues can be defined with the output for each queue directed to any supported destination. E.g., POWER LST, SAM, VSAM ESDS, etc.

The BSTTLPDS service follows RFC 1179 and will accept LPR control and data files arriving in any sequence. Text arriving in the LPR data file is expected to be ASCII text with no ASA characters and imbedded LineFeed and FormFeed characters. The ASCII data is translated to EBCDIC and written to the defined OUTPUT destination.

The complete reference for this step can be found in the IPv4 Users Guide and the IPv6 Users Guide manuals.

### Configuration Parameters

Parameter	Description	Sample Value	Your Value
ID	Stack ID	00 (IPv4), 66 (IPv6)	
OPEN	IP address or name	10.1.1.1	
USER	Userid	userid	
PASS	Password	password	
LIBDEF	BSI Library	bsilib.slib	

### QUEUE and OUTPUT Commands

The QUEUE and OUTPUT commands define the LPD queue and where the queue's output will be directed.

Parameter	Description
QUEUE	Queue name
OUTPUT	OUTPUT command

### Sample BSTTLPDS Execution JCL

The following sample JCL shows how to use the BSTTLPDS service. Note the use of mixed and lower case characters. On remote systems character case can be very important.

The use of the literal \$\$NAME\$\$ in the OUTPUT POWER command is required. And, the OUTPUT commands processed by BSTTLPDS are not free form. Only a single space is permitted between any parameter.

```
// DLBL IJSYSUC, 'VSESP.USER.CATALOG',, VSAM
// DLBL BSAULOG, 'TEST.AUDIT.LOG.FILE',, VSAM
/*
// OPTION SYSPARM='00'
// EXEC BSTTWAIT, SIZE=BSTTWAIT
/*
// EXEC BSTTLPDS, SIZE=BSTTLPDS
ID 00
OPEN 127.0.0.1
SBCS RACOON
*
* QUEUE COMMAND SPECIFIES QUEUE NAME
* OUTPUT COMMAND IS THE OUTPUT DESTINATION
* QUEUE AND OUTPUT COMMANDS ARE COMMAND PAIRS
*
QUEUE raw
OUTPUT POWER LST $$NAME$$ 0 P
*
QUEUE RAW
OUTPUT POWER LST $$NAME$$ 0 P
*
QUEUE ZVSE
OUTPUT POWER LST $$NAME$$ 0 L
*
QUEUE ESDS
OUTPUT ESDS BSAULOG
*
ATTACH LPD-1
ATTACH LPD-2
ATTACH LPD-3
ATTACH LPD-4
/*
```

## BSIREXXC Symbolic Variable Support

The BSIREXXC.PROC was written by Dave Clark. BSIREXXC.PROC provides symbolic variable support for invoking the BSTTFTP, BSTTMTPC, BSTTLPRC and BSTTREXC applications.

In addition to symbolic variables, BSIREXXC supports variable passed into the REXX EXEC using the // SETPARAM JCL statement.

```
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */  
/* No warranty is expressed or implied.        Written by Dave L Clark */  
/* Neither the author nor the company is       WinWholesale Group Svcs */  
/* responsible for any loss or damage           3110 Kettering Blvd.   */  
/* resulting from the use of this code.         Dayton, OH  45439      */  
/* Source is provided on an as-is basis.        Phone (937)531-6378    */  
/* Dave Clark ... dlclark@winwholesale.com */  
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
```

**Symbolic Variables**

```

/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
/* Note: The passed argument string must take into consideration */
/* the delimiters for variable names as expected by this function. */
/* The less-than (<) and greater-than (>) symbols are expected as */
/* delimiters for VSE Symbolic Parameter names. The percent (%) */
/* symbol is expected as delimiters for internal variable names -- */
/* as shown below: */
/* */
/* %D$1% 2-digit year value for the current date; */
/* %D$2% 2-digit month value for the current date; */
/* %D$3% 2-digit day value for the current date; */
/* %D$4% 4-digit year value for the current date; */
/* %D$5% occurrence of this weekday (1-5) in current month; */
/* %D$6% weekly cycle number (1-6) for the current month; */
/* %D$B% complete days since base date of 01/01/0001; */
/* %D$C% inclusive days since start of current century; */
/* %D$D% inclusive days since start of current year; */
/* %D$E% current European formatted date (dd/mm/yy); */
/* %D$J% current Julian unformatted date (yyddd); */
/* %D$M% full English name for the current month; */
/* %D$N% current Normalized date format (dd mon ccy); */
/* %D$O% current Ordered formatted date (yy/mm/dd); */
/* %D$Q% current Quarter number of the current year; */
/* %D$S% current Standard unformatted date (ccyymmdd); */
/* %D$U% current USA formatted date (mm/dd/yy); */
/* %D$V% 3-char abbreviated name for the current weekday; */
/* %D$W% full English name for the current weekday; */
/* %D$X% 1-digit weekday number (1-7: Sunday thru Sabbath); */
/* %D$Y% occurrence of this weekday (01-53) in current year; */
/* %D$Z% weekly cycle number (01-54) for the current year; */
/* */
/* %T$C% current Civil (12hr) formatted time (hh:mmxx); */
/* %T$H% hours since midnight of the current day; */
/* %T$L% current Long formatted time (hh:mm:ss.uuuuuu); */
/* %T$M% minutes since midnight of the current day; */
/* %T$N% current Normal (24hr) formatted time (hh:mm:ss); */
/* %T$S% seconds since midnight of the current day; */
/* %T$T% current unformatted Time (24hr) value (hhmmssuu); */
/* */
/* %V$C% current POWER Job Class; */
/* %V$I% current CPU Id; */
/* %V$J% current POWER Job Name; */
/* %V$N% current POWER Job Number; */
/* %V$P% current VSE SYSLOG Partition Id; */
/* %V$R% current VSE MRC value; */
/* %V$U% current User Id (or, if none, the Job Name); */
/* %V$V% current VSE // JOB Name; and, */
/* %V$Z% current zVSE Supervisor version number. */

```

```
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
```

## Overview

```
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
/* REXX: BSIREXXC Proc Created 05/18/2004 By Dave Clark */
/* Last Revised 04/13/2007 */
/* */
/* This procedure reads card input and performs variable name */
/* resolution for the purpose of passing such cards to an */
/* internally-invoked instance of any of the BSI batch clients. */
/* Supplied input must consist of all input cards (not JCL) */
/* required for a particular BSI batch client. Any JCL needed for */
/* the particular BSI batch client would precede execution of this */
/* procedure. The particular BSI batch client desired is */
/* specified as a JCL EXEC PARM argument -- as follows: */
/* */
/* // EXEC REXX=BSIREXXC,SIZE=196K,PARM='...' */
/* */
/* Arguments are: <CLIENT=>client */
/* */
/* Where: CLIENT= is an optional keyword tag; */
/* client is the REXX synonym of the particular BSI batch */
/* client to invoke. The REXX synonym is based */
/* on what you specified in the ARXEOJT table */
/* so that REXX would support it (more below). */
/* */
/* Exception: Due to its primary design, the BSTTMTPC client */
/* normally couldn't be invoked internally. The reason */
/* is that it requires two SYSIPT files, which can't be */
/* accommodated from any procedure execution in VSE. */
/* Hence, this procedure must perform special */
/* processing on its behalf (more info below). */
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
```



## Updating ARXJOBTB

```

/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
/*      The following is an abbreviated example of the type of */
/* modifications you must make to the PRD1.BASE.ARXEOJTB.Z library */
/* member and then re-assemble it into the PRD2.CONFIG library: */
/*                                                                */
/* .                                                                */
/* .                                                                */
/* .                                                                */
/* TOTAL      DC F'12'          /* Total # of entries           */
/* USED       DC F'12'          /* # OF ENTRIES USED           */
/* .                                                                */
/* .                                                                */
/* .                                                                */
/* ENTRY_9    EQU *            /* ARXEOJTB Entry 9           */
/*          DC CL8'BSIFTP '    /* SYNONYM USED IN ADDRESS LINK */
/*          DC CL8'BSTTFTPC'   /* NAME OF PHASE                */
/*          DC AL4(0)          /* Must be Zero                 */
/*          DC CL1'NO'         /* Phase Loaded in program area */
/*          DC CL7' '         /* Reserved                       */
/* ENTRY_10   EQU *            /* ARXEOJTB Entry 10          */
/*          DC CL8'BSILPR '    /* SYNONYM USED IN ADDRESS LINK */
/*          DC CL8'BSTTLPRC'   /* NAME OF PHASE                */
/*          DC AL4(0)          /* Must be Zero                 */
/*          DC CL1'NO'         /* Phase Loaded in program area */
/*          DC CL7' '         /* Reserved                       */
/* ENTRY_11   EQU *            /* ARXEOJTB Entry 11          */
/*          DC CL8'BSIMTP '    /* SYNONYM USED IN ADDRESS LINK */
/*          DC CL8'BSTTMTPC'   /* NAME OF PHASE                */
/*          DC AL4(0)          /* Must be Zero                 */
/*          DC CL1'NO'         /* Phase Loaded in program area */
/*          DC CL7' '         /* Reserved                       */
/* ENTRY_12   EQU *            /* ARXEOJTB Entry 12          */
/*          DC CL8'BSIREX '    /* SYNONYM USED IN ADDRESS LINK */
/*          DC CL8'BSTTREXC'   /* NAME OF PHASE                */
/*          DC AL4(0)          /* Must be Zero                 */
/*          DC CL1'NO'         /* Phase Loaded in program area */
/*          DC CL7' '         /* Reserved                       */
/* .                                                                */
/* .                                                                */
/* .                                                                */
/* .                                                                */
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */

```



## Invoking **BSIREXXC**

```

/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
/*      Note that this procedure is only required in order to */
/* process symbolic variable name resolutions and, as such, those */
/* symbolic variables must be passed to this procedure so that it */
/* has access to their values. For example: */
/* */
/* // JOB      MYFTP      INTERNALLY INVOKE FTP/W VAR SUBSTITUTION */
/* // ... JCL for BSTTFTPC goes here ... */
/* // SETPARAM CO='921' */
/* // SETPARAM ASOF='200405' */
/* // EXEC     REXX=BSIREXXC,SIZE=196K,PARM='BSIFTP',CO,ASOF */
/* ID 00 */
/* OPEN xxx.xxx.xxx.xxx */
/* USER xxxxxxxxx */
/* PASS xxxxxxxxx */
/* SYST */
/* CWD \ACCT\SALESTAX\CO<CO>\DATAFILES\ */
/* INPUT VSAM TAX2000 */
/* TYPE A */
/* STOR P<ASOF>_TAX2000.PHX */
/* QUIT */
/* /* EOD */
/* /& EOJ */
/* */
/* Also note that the less-than/greater-than symbols are used to */
/* delineate the VSE symbolic parameter name for which this */
/* procedure is to perform variable name resolution. In addition, */
/* see the external REPLVARS REXX function for the internal */
/* symbolic variables available. */
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */

```

### **Special BSTTMTPC Considerations**

```

/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
/*      The following is an example of submitting a job to process */
/* an email with variable name resolution:                               */
/*                                                                           */
/* // JOB      MYMTP      INTERNALLY INVOKE MTP/W VAR SUBSTITUTION      */
/* // ... JCL for BSTTMTPC goes here ...                                  */
/* // SETPARAM CO='921'                                                  */
/* // SETPARAM ASOF='200405'                                             */
/* // EXEC     REXX=BSIREXXC,SIZE=196K,PARM='BSIMTP',CO,ASOF             */
/* * $$ SLI MEM=BSTTMTPC.B                                               */
/* RCPT To: email@dapsco.com                                             */
/* SUBJ Subject: TAX2000 file downloaded                                  */
/* * $$ SLI MEM=BSTTMTP2.B                                               */
/* /? end of BSTTMTPC commands                                           */
/* TAX2000 file for company <CO> has been downloaded to:                */
/*                                                                           */
/* \\SERVER\ACCT\SALESTAX\CO<CO>\DATAFILES\                              */
/*                                                                           */
/* with a file name of:                                                 */
/*                                                                           */
/* P<ASOF>_TAX2000.PHX                                                    */
/* /* EOD */                                                              */
/* /& EOJ                                                                  */
/*                                                                           */
/* Note the '/?' used (instead of '/*') to delimit the two SYSIPT */
/* files that the BSTTMTPC client normally requires. This is */
/* detected and prior input written to a member in the PRD2.CONFIG */
/* VSE library (as supplied, but may be changed below) -- which */
/* member the BSTTMTPC client can read instead of the first SYSIPT */
/* file it would normally require. Then, succeeding input is used */
/* as the second SYSIPT file for the client. The library member */
/* is deleted after the client invocation completes.                  */
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */

```

## BSTTFTPS FTP Server Scripts

Many FTP client applications (both command line and GUI) support scripting.

The Windows command line FTP client has scripting support ...

### Syntax

```
ftp [-v] [-d] [-i] [-n] [-g] [-s:FileName] [-a] [-w:WindowSize] [-A] [Host]
```

`-s:FileName` : Specifies a text file that contains ftp commands. These commands run automatically after ftp starts. This parameter allows no spaces. Use this parameter instead of redirection (<).

FTP client scripts will generally run with either the TCP/IP for VSE FTP server or the IPv6/VSE FTP server. The only real differences are; One, using a `cd` command to mount the filesystem with IPv6/VSE and Two, `SITE` commands used with the IPv6/VSE FTP server.

The IPv6/VSE FTP server will accept any `SITE` command that is sent to it. This is useful when an FTP client attempts to send an unusual `SITE` command. However, only valid BSTTFTPS `SITE` commands are actually processed.

### ***Mounting a Filesystem***

Mounting a filesystem and setting a directory is simple using a `'cd'` command. Note the use of dashes (-) in the `SMNT` command. In the `cd` form the of `smnt` command dashes are used instead of spaces.

```
cd smnt-power/rdr
cd smnt-library-prd2/prod
cd smnt-vsam-vsesp.user.catalog
```

### ***Script Example***

```
ftp -s:ftptest.txt
```

With `ftptest.txt` containing

```
192.168.1.1
userid
password
cd smnt-power
cd lst
ascii
get class.number.name
```

```
quit
```

### **FTP Server SITE commands**

The IPv6/VSE FTP server will accept any SITE command that is sent to it. This is useful when non-z/VSE hosts attempt to send unusual SITE commands. However, only valid BSTTFTPS SITE commands are actually processed.

A complete list of BSTTFTPS SITE can also be found in the IPv6/VSE IPv4 and IPv6 Users Guides.

<b>FTP Server SITE Command</b>	<b>Description</b>
SITE PAD ON OFF	Set record padding on or off
SITE PADCHAR nnn	Set the value of the pad character (decimal)
SITE BLKSZ nnn	Set the block size value
SITE RECSZ nnn	Set the record size value
SITE RESET ON OFF	Set VSAM reset-to-empty-on-open flag
SITE LMODE F S	Set library mode to fixed or string
SITE SBCS ON OFF	Set SBCS on or off
SITE DBCS ...	Set DBCS information
SITE SOSI ON OFF	Set DBCS SOSI on or off
SITE CONVERT ON OFF	Set DBCS CONVERT on or off
SITE KEEP ON OFF	Set DBCS KEEP on or off
SITE XLATE ON OFF	Set DBCS XLATE on or off
SITE BLANK ON OFF	Set DBCS BLANK on or off
SITE NONE ON OFF	Set DBCS NONE on or off
SITE CRLF ON OFF	Set CRLF on or off
SITE NL ON OFF	Set New Line on or off
SITE LF ON OFF	Set Line Feed on or off
SITE ASA ON OFF	Set VSE/POWER ASA on or off
SITE LINES nnn	Set VSE/POWER auto skip line count
SITE TRCMD ON OFF	Set VSE/POWER translate commands on or off
SITE HTML ON OFF	Set library member to HTML mode
SITE PSWD value	Set VSAM password
SITE FNO value	Set VSE/POWER form
SITE NODE value	Set VSE/POWER destination node name
SITE USER value	Set VSE/POWER destination node userid

## IPv6/VSE Migration Guide

SITE JSEP n	Set VSE/POWER JSEP value
SITE UPJS ON OFF	Set Use VSE/POWER Job Separators
SITE DISP x	Set VSE/POWER DISP value
SITE VALIDATE ON OFF	Set VSE/POWER character validation on/off
SITE PWRCMD <command>	Send VSE/POWER a command
SITE WTO	Write message to console
SITE AUXDIR ON OFF	Use alternate directory format (VSE/POWER)
SITE UINF <string>	Set VSE/POWER USER='string' value

## **BSTTPREP Converting CSI EXEC TCP to EZASOKET**



## **CICS TS FTP Client Transactions**

## IBM Virtual Tape Server

The TAPESRVR job found in the POWER RDR queue after installation of z/VSE requires slight modification to execution using IPv6/VSE.

### **Modified JCL**

```
// OPTION PARTDUMP,NOSYSDUMP
// OPTION SYSPARM='00'
// SETPARM IPTRACE='NNNNNNNNN'
// SETPARM SVABUF=YES
// SETPARM SENDALL=YES
// SETPARM LRGBUF=YES
// LIBDEF *,SEARCH=(BSILIB.TTDEV,PRD2.CONFIG,PRD1.BASE,PRD2.SCEEBASE)
// EXEC $VTMAIN
```

SVABUF=YES indicates the application using SVA buffers.

IPTRACE='NNNNNNNNN' disabled application API tracing.

SENDALL=YES ensures correct processing of short send() requests from LE/C applications.

LGRBUF=YES indicates the application would like the stack to use large TCP buffers.

The LIBDEF \*,SEARCH specifies the BSI lib.slib first before other libraries.

## ***Transferring .aws VTAPE Files using FTP***

The following example assume the DLBL for the VSAM ESDS VTAPE cluster is in standard labels.

When transferring a VSAM ESDS VTAPE cluster to a remote host VTAPE JAVA server PC, a BINARY transfer of the data is required.

Using the BSTTFTPS FTP Server ...

```
ftp> bin
ftp> quote site input vsam dlbl
ftp> get pc.file.name
ftp> quit
```

Using the BSTTFTPC Batch FTP Client ...

```
// DLBL VTAPE0,'...!',VSAM,CAT=...
// EXEC BSTTFTPC,SIZE=BSTTFTPC
ID 00
OPEN ...
USER ...
PASS ...
CWD ...
*
TYPE I
*
INPUT VSAM VTAPE0
STOR vtape.file.name
*
QUIT
/*
```

## IPv6/VSE Migration Guide

When transferring a remote host JAVA server PC .aws file to a z/VSE VSAM ESDS cluster, a BINARY transfer of the data is required plus variable length must be specified.

Using the BSTTFTPS FTP Server ...

```
ftp> bin
ftp> quote site output esds dlbl recsz 32758 * * V
ftp> put vtape.file.name
ftp> quit
```

Using the BSTTFTPC Batch FTP Client ...

```
// DLBL VTAPE0,'...!',VSAM,CAT=...
// EXEC BSTTFTPC,SIZE=BSTTFTPC
ID 00
OPEN ...
USER ...
PASS ...
CWD ...
*
TYPE I
*
OUTPUT ESDS VTAPE0 RECSZ 32758 * * V
RETR file.name
*
QUIT
/*
```

## IBM Connector Server

The IBM connector server JCL requires slight modification. The STARTVCS JCL found in the POWER RDR after installation will look similar to this ...

### **Modified JCL**

```
// LIBDEF *,SEARCH=(bsilib.slib,PRD2.CONFIG,PRD1.BASE,PRD2.SCEEBASE)
// OPTION SYSPARM='00'
// EXEC BSTTWAIT,SIZE=BSTTWAIT
// SETPARM IPTRACE='NNNNNNNN'
// SETPARM SENDALL='YES'
// EXEC IESVCSRV,PARM='DD:PRD2.CONFIG(IESVCSRV.Z)'
/*
```

The LIBDEF must specify the BSI lib.slib first.

The BSTTWAIT execution waits until the TCP/IP stack is available.

The IPTRACE='NNNNNNNN' disables tracing.

The SENDALL=YES ensures correct processing of short send() requests from LE/C applications.

## CICS TS Web Services

These are the steps to activate CICS TS Web Services on a base install z/VSE system. The process is pretty simple.

### CICS TS JCL Changes

Then I added the following JCL to the CICS job ...

```
1) // OPTION SYSPARM='00' Stack ID
2) // SETPARAM LOCALGT=YES
3) Added bsilib.slib first in the LIBDEF SEARCH chain
4) Added the BSTTWAIT JCL

// EXEC BSTTWAIT,SIZE=BSTTWAIT
/*
```

Note: BSTTWAIT sets the VSE JCL return code. RC=0 indicates the BSTTINET TCP/IP stack is up and available. RC=8 indicates the TCP/IP stack was not started within 10 minutes. If you want to be able to wait more than 10 minutes using BSTTWAIT, use VSE conditional JCL to re-execute the step if the return code is equal to 8.

5) Add TCPIP=YES to the CICS SYSIPT overrides on both // EXEC DFHSIP steps

Now your CICS TS JCL is ready to go. However, a couple of more things to do before you bring up CICS TS CWS for the first time.

### CEDA TCPIP Entries

Now logon to your current CICSICCF system to copy and activate the TCPIP CEDA entries.

From a CEDA display ...

```
COPY GROUP (DFH$SOT) TCPIPSERVICE (HTTPNSSL) TO (BSI$SOT)
ALTER GROUP (BSI$SOT) TCPIPSERVICE (HTTPNSSL)
```

Change the port number if desired (I used 8080)

And finally

```
ADD GROUP (BSI$SOT) LIST (VSELIST) AFTER (TCPIP)
```

**ASCII/EBCDIC Conversion**

At this point we need to create an ASCII/EBCDIC conversion table. This JCL was taken directly from the IBM manual. You will need to modify it as necessary for you system.

```

* $$ JOB JNM=DFHCVN,CLASS=8,LDEST=(,BARNARD),PDEST=(,BARNARD)
* $$ LST JSEP=0,CLASS=0
* $$ PUN JSEP=0,CLASS=0
// JOB DFHCVN
// LIBDEF *,CATALOG=PRD2.CONFIG
// LIBDEF SOURCE,SEARCH=(PRD1.BASE,PRD1.MACLIB)
// OPTION CATAL,LIST
// EXEC ASMA90,SIZE=(ASMA90,64K),PARM='EXIT(LIBEXIT(EDECKXIT)),SIZE(MAXC
      -200K,ABOVE)'
      DFHCVN TYPE=INITIAL
      DFHCVN TYPE=ENTRY,RTYPE=PC,RNAME=DFHVBHH,USREXIT=NO,          X
          SRVERCP=037,CLINTCP=437
      DFHCVN TYPE=SELECT,OPTION=DEFAULT
      DFHCVN TYPE=FIELD,OFFSET=0,DATATYP=CHARACTER,DATALEN=32767,  X
          LAST=YES
      DFHCVN TYPE=ENTRY,RTYPE=PC,RNAME=DFHVBUD,USREXIT=NO,        X
          SRVERCP=037,CLINTCP=437
      DFHCVN TYPE=SELECT,OPTION=DEFAULT
      DFHCVN TYPE=FIELD,OFFSET=0,DATATYP=CHARACTER,DATALEN=32767,  X
          LAST=YES
      DFHCVN TYPE=FINAL
      END
/*
// IF $MRC GT 4 THEN
// GOTO NOLINK
// EXEC LNKEDT
/. NOLINK
/*
/&
* $$ EOJ

```

Now you are ready to shutdown your original CICSICCF partition and bring up your new CICSICCF JCL that will include CWS support.

## ***CICS TS Web Services Test***

To test the CWS interface using this command from your web browser ...

`http://192.168.1.228:8080/CICS/CWBA/DFH$WB1A`

Of course, you will have to change the IP address to reflect your system's IP address.

When I entered the above URL in my Firefox browser I received this response ...

*DFH\$WB1A on system DBDCCICS successfully invoked through CICS Web Support on CICS Transaction Server for VSE/ESA.*