



**Program Directory for  
TCP/IP Feature for z/VM  
and Features**

Level 3A0

Program Number 5654-A17

for Use with  
z/VM Version 3 Release 1.0

Document Date: May 2003

GI10-4710-00

**Note!**

Before using this information and the product it supports, be sure to read the general information under "Notices" on page 132.

This program directory, dated May 2003, applies to the TCP/IP Feature for z/VM (TCP/IP Feature), Program Number 5654-A17.

A form for reader's comments appears at the back of this publication. When you send information to IBM®, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© **Copyright International Business Machines Corporation 1990, 2001. All rights reserved.**

Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

---

# Contents

<b>1.0 Introduction</b>	1
1.1 Program Description	2
<b>2.0 Program Materials</b>	4
2.1 Obtaining Authorization for the TCP/IP or TCP/IP NFS Server Feature	5
2.2 Basic Machine-Readable Material	6
2.3 Optional Machine-Readable Material	6
2.3.1 Optional Machine-Readable Material for the TCP/IP NFS Server Feature for z/VM	6
2.3.2 Optional Machine-Readable Material for the TCP/IP Source Feature for z/VM	6
2.4 Program Publications	7
2.4.1 Basic Program Publications	7
2.4.2 Softcopy Publications	8
2.5 Microfiche Support	8
2.6 Publications Useful During Installation and Service	8
<b>3.0 Program Support</b>	9
3.1 Preventive Service Planning	9
3.2 Statement of Support Procedures	9
3.3 Service Information	10
3.3.1 Problem Documentation	10
3.3.2 Communicating Your Comments to IBM	11
<b>4.0 Program and Service Level Information</b>	12
4.1 Program Level Information - TCP/IP Feature for z/VM	12
4.2 Program Level Information - TCP/IP NFS Server Feature for z/VM	12
4.3 Service Level Information	13
4.4 Cumulative Service (RSU) Tape Information	13
<b>5.0 Installation Requirements and Considerations</b>	14
5.1 Hardware Requirements	14
5.2 Program Considerations	14
5.2.1 Operating System Requirements	14
5.2.2 Other Program Product Requirements	14
5.2.2.1 Other Program Product Requirements - TCP/IP Feature for z/VM	14
5.2.2.2 Other Program Product Requirements - TCP/IP NFS Server Feature for z/VM	15
5.2.2.3 Other Program Product Requirements - SSL Server	15
5.2.3 Program Installation/Service Considerations	15
5.2.4 Migration Considerations	17
5.2.4.1 Packaging	17
5.2.4.1.1 General Information About TCP/IP Level 3A0	17
5.2.4.1.2 Changes Introduced in TCP/IP Level 3A0	17
5.2.4.1.3 Changes Introduced in TCP/IP Function Level 310	18

5.2.4.2	Server Configuration	18
5.2.4.2.1	Changes Introduced in TCP/IP Function Level 310	18
5.2.4.3	TCPIP Server	19
5.2.4.3.1	Changes Introduced in TCP/IP Function Level 320	19
5.2.4.4	Routed/MPROUTE Server	19
5.2.4.4.1	Changes Introduced in TCP/IP Level 3A0	19
5.2.4.4.2	Changes Introduced in TCP/IP Function Level 320	19
5.2.4.5	DNS Server	19
5.2.4.5.1	Changes Introduced in TCP/IP Level 3A0	19
5.2.4.6	SMTP Server	20
5.2.4.6.1	Changes Introduced in TCP/IP Level 3A0	20
5.2.4.6.2	Changes Introduced in TCP/IP Function Level 320	20
5.2.4.6.3	Changes Introduced in TCP/IP Function Level 310	20
5.2.4.7	FTP Server	21
5.2.4.7.1	Changes Introduced in TCP/IP Level 3A0	21
5.2.4.7.2	Changes Introduced in TCP/IP Function Level 320	21
5.2.4.8	FTP Client	22
5.2.4.8.1	Changes Introduced in TCP/IP Level 3A0	22
5.2.4.8.2	Changes Introduced in TCP/IP Function Level 320	22
5.2.4.9	Printing	22
5.2.4.9.1	Changes Introduced in TCP/IP Function Level 310	22
5.2.4.10	Kerberos Server	22
5.2.4.10.1	Changes Introduced in TCP/IP Level 3A0	22
5.3	DASD Storage and User ID Requirements	23
5.3.1	DASD Requirements for the TCP/IP Feature for z/VM	26
5.3.2	DASD Requirements for the TCP/IP Source Feature for z/VM	32
<b>6.0</b>	<b>Installation Instructions</b>	<b>33</b>
6.1	TCP/IP Feature Installation and Activation Process Overview	34
6.2	Installing the TCP/IP Feature for z/VM — Where to Begin	34
6.2.1	Review the TCP/IP Feature for z/VM Default Installation Environment	35
6.2.1.1	TCP/IP Feature Directory PROFILES and User IDs	35
6.2.2	(Optional) Moving the TCP/IP Feature to SFS Directories	37
6.2.3	Enable the TCP/IP Feature	38
6.2.4	Place the TCP/IP Feature for z/VM Into Production	40
6.2.4.1	3TCPIPA0 CATALOG File Modification Notes	40
6.2.4.2	Copy TCP/IP Feature Run-time Files Into Production	41
6.2.4.3	Copy TCP/IP Feature Configuration Files Into Production	43
6.2.4.4	Configure the TCP/IP Feature for z/VM for Your Installation	44
6.2.4.5	Initialize TCP/IP Services	44
6.2.4.6	(Optional) Copy TCP/IP Client Code to the z/VM Product Code Disk	45
6.2.4.7	TCP/IP Feature for z/VM Product and Sample Configuration Files	45
6.3	Enabling the TCP/IP NFS Server Feature for z/VM	51
6.4	Installing the TCP/IP Source Feature for z/VM	53
6.4.1	Plan Your Installation for the TCP/IP Source Feature for z/VM	53
6.4.2	Allocate Resources for Installing the TCP/IP Source Feature for z/VM	57

6.4.2.1	Installing the TCP/IP Source Feature for z/VM to Minidisks	57
6.4.2.2	Installing the TCP/IP Source Feature for z/VM to SFS Directories	57
6.4.3	Install the TCP/IP Source Feature for z/VM	59
6.4.4	Place the TCP/IP Source Feature for z/VM Into Production	62
6.4.4.1	Unpack the TCP/IP Source Feature for z/VM Source Files	62
<b>7.0</b>	<b>Service Instructions</b>	<b>63</b>
7.1	VMSES/E Service Process Overview	63
7.2	Servicing the TCP/IP Feature for z/VM	65
7.2.1	Important Service Notes	65
7.2.1.1	PPF Override Considerations	65
7.2.1.2	Language Environment Run-time Library Considerations	65
7.2.1.3	Installing RSU and COR Service — Where to Begin	65
7.2.2	Preventive (RSU) Service for the TCP/IP Feature for z/VM	65
7.2.2.1	Prepare to Receive Service	66
7.2.2.2	Receive the Service	68
7.2.2.3	Apply the Service	69
7.2.2.4	Update the Build Status Table	70
7.2.2.5	Build Serviced Objects	72
7.2.2.6	Test the New Service	73
7.2.2.7	Place the Service into Production	73
7.2.3	Corrective (COR) service for the TCP/IP Feature for z/VM	74
7.2.3.1	Prepare to Receive Service	74
7.2.3.2	Receive the Service	76
7.2.3.3	Apply the Service	77
7.2.3.4	Update the Build Status Table	78
7.2.3.5	Build Serviced Objects	80
7.2.3.6	Test the New Service	80
7.2.4	Place the New TCP/IP Feature for z/VM Service Into Production	81
7.2.4.1	3TCPIPA0 CATALOG File Modification Notes	81
7.2.4.2	Copy Serviced TCP/IP Feature Run-time Files Into Production	82
7.2.4.3	Copy New TCP/IP Feature Configuration Files Into Production	84
7.2.4.4	Build and Copy Serviced TCP/IP Feature System Files Into Production	86
7.2.4.5	Configure the TCP/IP Feature for z/VM	88
7.2.4.6	(Optional) Copy Serviced TCP/IP Client Code to the z/VM Product Code Disk	89
7.2.4.7	Re-Initialize TCP/IP Services	89
7.2.4.8	(Optional) Verify the RSU Service Level	89
7.3	Servicing the TCP/IP NFS Server Feature for z/VM	91
7.4	Servicing the TCP/IP Source Feature for z/VM	92
<b>Appendix A.</b>	<b>The TCP2PROD Command</b>	<b>93</b>
A.1	TCP2PROD	93
A.1.5	The TCP2PROD \$MSGLOG File	95
A.2	TCP/IP for VM CATALOG Files	96
<b>Appendix B.</b>	<b>Modifying TCP/IP Feature CATALOG Files</b>	<b>100</b>

<b>Appendix C. PTF Commit Procedure for TCP/IP Feature System Files</b> . . . . .	104
<b>Appendix D. Making Local Modifications to TCP/IP Feature Modules</b> . . . . .	108
D.1.1 Prepare for Local Service . . . . .	109
D.1.2 Receive the Local Service . . . . .	110
D.1.2.1 Create the Replacement TEXT File . . . . .	110
D.1.2.2 Add an Update Record to the AUX File . . . . .	111
D.1.2.3 Create the Update File for the Part . . . . .	111
D.1.2.4 Create an Updated Replacement TEXT File . . . . .	112
D.1.2.5 Copy the TEXT File for Use by VMSES/E . . . . .	113
D.1.3 (Optional) Modify the Build List for Added TEXT Files . . . . .	113
D.1.3.1 Add an Update Record to the Build List AUX File . . . . .	114
D.1.3.2 Create the Update File for the Build List . . . . .	114
D.1.3.3 Create the Updated Replacement Build List . . . . .	115
D.1.4 Rebuild the Modified Objects . . . . .	116
<b>Appendix E. Modifying TCP/IP NFS Server Feature VMNFS Code</b> . . . . .	118
E.1 Modifying the NFSHCHIP ASSEMBLE and TCPBLC91 EXEC Files . . . . .	118
E.2 Modifying VMNFS C Source Files . . . . .	119
<b>Appendix F. TCP/IP Feature Build Lists</b> . . . . .	120
F.1 TCP/IP Feature for z/VM Build Lists . . . . .	120
<b>Appendix G. Moving the TCP/IP Feature to SFS Directories</b> . . . . .	122
G.1.1 Setup the SFS File Pool and Required Directories . . . . .	123
G.1.2 Copy Minidisk Files to SFS Directories . . . . .	124
G.1.3 Create a Product Parameter File (PPF) Override . . . . .	125
<b>Appendix H. Copying TCP/IP Feature Client Code to the Y-Disk</b> . . . . .	127
<b>Appendix I. Disabling the TCP/IP and TCP/IP NFS Server Features</b> . . . . .	130
<b>Notices</b> . . . . .	132
Trademarks and Service Marks . . . . .	134
<b>Reader's Comments</b> . . . . .	135

---

## Figures

1.	Basic Material: Program Tape for the TCP/IP Source Feature for z/VM	6
2.	Program Tape: File Content for the TCP/IP Source Feature for z/VM	7
3.	Basic Material: Unlicensed Publications	7
4.	Publications Useful During Installation / Service on z/VM Version 3 Release 1.0	8
5.	PSP Upgrade and Subset ID	9
6.	Component IDs	10
7.	Problem Documentation Addresses	11
8.	Cumulative Service (RSU) Tape Information	13
9.	Alternative Minidisk Storage Requirements	24
10.	DASD Storage Requirements for Target Minidisks - TCP/IP Feature	26
11.	DASD Storage Requirements for Target Minidisks - TCP/IP Source Feature for z/VM	32
12.	TCP/IP Feature for z/VM System Directory Profiles	35
13.	Default User IDs - TCP/IP Feature for z/VM	36
14.	Files to Copy into Production - TCP/IP Feature for z/VM	46
15.	Files to Customize - TCP/IP Feature for z/VM	48
16.	System Files to Build/Copy into Production - TCP/IP Feature for z/VM	50
17.	TCP/IP Feature for z/VM CATALOG Files	96
18.	TCP/IP Feature System Files	104
19.	VMSES/E Build Lists - TCP/IP Feature for z/VM	120





---

## 1.0 Introduction

This program directory is intended for the system programmer responsible for program installation and maintenance. It contains information concerning the material and procedures associated with the installation and service of the following:

- TCP/IP Feature for z/VM (TCP/IP Feature)
- TCP/IP Network File System (NFS) Server Feature for z/VM
- TCP/IP Source Feature for z/VM

You should read all of this program directory before you install this program, then keep this document for future reference.

The program directory contains the following sections:

- 2.0, “Program Materials” on page 4 identifies the basic and optional program materials and documentation for the TCP/IP Feature for z/VM, and its corresponding features.
- 3.0, “Program Support” on page 9 describes the IBM support available for TCP/IP Feature for z/VM and its features.
- 4.0, “Program and Service Level Information” on page 12 lists the APARs (program level) and PTFs (service level) incorporated into the TCP/IP Feature for z/VM and its features.
- 5.0, “Installation Requirements and Considerations” on page 14 identifies the resources and considerations for installing and using the TCP/IP Feature for z/VM and its features.
- 6.0, “Installation Instructions” on page 33 provides detailed installation instructions for the TCP/IP Feature for z/VM and its features. These instructions are comprised of the following:
  - 6.2, “Installing the TCP/IP Feature for z/VM — Where to Begin” on page 34
  - 6.3, “Enabling the TCP/IP NFS Server Feature for z/VM” on page 51
  - 6.4, “Installing the TCP/IP Source Feature for z/VM” on page 53
- 7.0, “Service Instructions” on page 63 provides detailed servicing instructions for the TCP/IP Feature for z/VM and its features. These instructions are comprised of the following:
  - 7.2, “Servicing the TCP/IP Feature for z/VM” on page 65
  - 7.3, “Servicing the TCP/IP NFS Server Feature for z/VM” on page 91
  - Note:** The TCP/IP NFS Server Feature is serviced as part of the TCP/IP Feature for z/VM.
  - 7.4, “Servicing the TCP/IP Source Feature for z/VM” on page 92
- Appendix A, “The TCP2PROD Command” on page 93 provides information about the TCP2PROD command, supplied for placing TCP/IP Feature files into production.
- Appendix B, “Modifying TCP/IP Feature CATALOG Files” on page 100 provides information about how to change TCP/IP Feature CATALOG files using a VMSES/E local modification.

- Appendix C, “PTF Commit Procedure for TCP/IP Feature System Files” on page 104 provides information about how to commit existing PTF service that is associated with TCP/IP Feature (binary) system files.
- Appendix D, “Making Local Modifications to TCP/IP Feature Modules” on page 108 provides information to help you implement local modifications to TCP/IP Feature modules.
- Appendix E, “Modifying TCP/IP NFS Server Feature VMNFS Code” on page 118 provides information specific to TCP/IP NFS Server Feature VMNFS module local modifications.
- Appendix F, “TCP/IP Feature Build Lists” on page 120 provides information about the VMSES/E build lists used to maintain the TCP/IP Feature.
- Appendix G, “Moving the TCP/IP Feature to SFS Directories” on page 122 provides instructions for changing the TCP/IP Feature service environment to use Shared File System directories instead of default minidisks.
- Appendix H, “Copying TCP/IP Feature Client Code to the Y-Disk” on page 127 provides considerations and instructions for copying client files to the system Product Code minidisk.
- Appendix I, “Disabling the TCP/IP and TCP/IP NFS Server Features” on page 130 provides information about deactivating the TCP/IP and TCP/IP NFS Server services.

Before you install the TCP/IP Feature for z/VM, read 3.1, “Preventive Service Planning” on page 9. This section tells you how to find any updates to the information and procedures in this program directory.

---

## 1.1 Program Description

TCP/IP (Transmission Control Protocol/Internet Protocol) enables z/VM customers to participate in a multivendor, open networking environment using the TCP/IP protocol suite for communications and interoperability. The applications provided in TCP/IP include the ability to transfer files, send mail, log on a remote host, allow access from any other TCP/IP node in the network, and perform other network client and server functions.

The Transmission Control Protocol/Internet Protocol Feature for z/VM, Level 3A0, (TCP/IP Feature for z/VM) contains the functions provided by TCP/IP Feature for z/VM Function Level 320, and provides the following enhancements:

- Queued Direct I/O support within the TCP/IP (*Stack*) server
- Addition of the Multi-Path Routing server (MPROUTE), which utilizes Open Shortest Path First (OSPF) and/or Routing Information (RIP) protocols
- IP Multicast support
- NFS client support that allows CMS Byte File System (BFS) users and applications to access data on remote systems (the TCP/IP NFS Server Feature for z/VM is **not** required in order to use this new NFS client support)
- FTP server modifications to better accommodate web browser and graphical FTP clients

- Addition of a Secure Socket Layer (SSL) server that can provide SSL connection support for any VM TCP/IP server
- Inclusion of the former TCP/IP Data Encryption Standard (DES) Feature for VM/ESA within the TCP/IP Feature for z/VM
- Miscellaneous documentation improvements
- Incorporation of miscellaneous service APARs for previous IBM TCP/IP for VM releases

---

## 2.0 Program Materials

An IBM program is identified by a program number. The program number for the TCP/IP Feature for z/VM is 5654-A17.

The z/VM Version 3 Release 1.0 program announcement material describes the features supported by the TCP/IP Feature for z/VM. Ask your IBM marketing representative for this information if you have not already received a copy.

The following sections identify:

- basic and optional program materials available with this program
- publications useful during installation.

## 2.1 Obtaining Authorization for the TCP/IP or TCP/IP NFS Server Feature

### Authorization and Ordering Information

Before you **USE** or **ENABLE** either of the following program features:

- TCP/IP Feature for z/VM
- TCP/IP NFS Server Feature for z/VM

**you MUST be authorized** for their use. Authorization is obtained by placing an order with IBM for the program feature you want to use. The TCP/IP features that you wish to use **must** be ordered by selecting the appropriate license group when you order z/VM Version 3 Release 1.0 using the IBM software configurator (**CFSW**). The selections available for this purpose are identified in the sample screen image that follows:

```
WSCATC07                7. Product License Selection

SW product: z/VM V3 3390 DDR                3 01.00 5654-A17

Type an action code next to the following (required) product licenses.
Then press Enter. Action codes: /=Select.
Marked (*) licenses already exist, installed licenses are highlighted.
                                           Page 1 of 1

The SW product requires only one of
- TCP/IP NFS Feature for VM 1                5654-A17
- TCP/IP Base Feature for VM                5654-A17
  z/VM Version 3 Base                        5654-A17
- TCP/IP Base Feature for VM 2                5654-A17
  z/VM Version 3 Base                        5654-A17
- z/VM Version 3 Base 3                5654-A17
```

The first selection ( **1** ) provides authorization for z/VM Version 3 Release 1.0 and all TCP/IP Feature functions, *including* the TCP/IP NFS Server Feature.

The second selection ( **2** ) provides authorization for z/VM Version 3 Release 1.0 and TCP/IP functions *except* the TCP/IP NFS Server Feature.

The third selection ( **3** ) provides authorization for z/VM Version 3 Release 1.0 **only**; the use of TCP/IP functions is **not** authorized.

If you need ordering assistance to obtain the desired features, please contact your IBM representative.

---

## 2.2 Basic Machine-Readable Material

The TCP/IP Feature for z/VM is distributed as part of the z/VM Version 3 Release 1.0 System DDR. If necessary, see the *z/VM Version 3 Release 1.0 Program Directory* for more information about the distribution media for z/VM Version 3 Release 1.0.

---

## 2.3 Optional Machine-Readable Material

There are two features associated with the TCP/IP Feature for z/VM:

- TCP/IP NFS Server Feature for z/VM
- TCP/IP Source Feature for z/VM

### 2.3.1 Optional Machine-Readable Material for the TCP/IP NFS Server Feature for z/VM

Executable files for the TCP/IP NFS Server Feature for z/VM are distributed as part of the TCP/IP Feature for z/VM. However, **you must separately order** the TCP/IP NFS Server Feature for z/VM before you enable and use the code associated with this feature. For ordering information, see 2.1, “Obtaining Authorization for the TCP/IP or TCP/IP NFS Server Feature” on page 5.

TCP/IP NFS Server Feature source files are distributed with the TCP/IP Source Feature for z/VM; details about optional machine-readable material for this feature are provided in 2.3.2, “Optional Machine-Readable Material for the TCP/IP Source Feature for z/VM.”

### 2.3.2 Optional Machine-Readable Material for the TCP/IP Source Feature for z/VM

The distribution medium for this program is 3480 tape cartridge, 4-mm tape cartridge, or CD-ROM. Each medium contains all of the programs and data necessary for program installation. See 6.0, “Installation Instructions” on page 33 for more information about how to install the program. Figure 1 describes each medium. Figure 2 describes the tape file content provided through each medium.

*Figure 1. Basic Material: Program Tape for the TCP/IP Source Feature for z/VM*

---

Feature Number			Physical		
Function-Media	BAU	Medium	Volume	Tape Content	External Tape Label
6470-3404	3622	3480 cart.	1	TCP/IP Source Feature for z/VM	TCP/IP VM Source
6470-3431	3623	4-mm cart.	1	TCP/IP Source Feature for z/VM	TCP/IP VM Source
6470-3410	3635	CD-ROM	1	TCP/IP Source Feature for z/VM	TCP/IP VM Source

Figure 2. Program Tape: File Content for the TCP/IP Source Feature for z/VM

<b>Tape File</b>	<b>Content</b>
1	Tape Header
2	Tape Header
3	Product Header
4	Product Memo
5	Service Apply Lists
6	PTFPARTs
7	TCP/IP Source Service
8	TCP/IP Source Service
9	TCP/IP Source Source Files

## 2.4 Program Publications

The following sections identify the basic and optional publications associated with the TCP/IP Feature for z/VM. There are no additional publications for the TCP/IP Source feature.

### 2.4.1 Basic Program Publications

The publications associated with the TCP/IP Feature are listed in Figure 3.

Figure 3. Basic Material: Unlicensed Publications

<b>Publication Title</b>	<b>Form Number</b>
z/VM: TCP/IP Level 3A0 Planning and Customization	SC24-5981
z/VM: TCP/IP Level 3A0 User's Guide	SC24-5982
z/VM: TCP/IP Level 3A0 Programmer's Reference	SC24-5983
z/VM: TCP/IP Level 3A0 Messages and Codes	GC24-5984
z/VM: TCP/IP Level 3A0 Diagnosis Guide	GC24-5985

One copy of the following program publication is provided at no charge to licencees of the TCP/IP Feature:

- z/VM: TCP/IP Level 3A0 Planning and Customization

## 2.4.2 Softcopy Publications

TCP/IP Feature for z/VM softcopy publications are provided as part of the z/VM Online Library that is installed as part of the z/VM Version 3 Release 1.0 System DDR. These softcopy publications are also included in the VM Collection kit CD-ROM. In addition, softcopy publications are available at the TCP/IP Feature home page on the World Wide Web; the URL for this home page is:

<http://www.vm.ibm.com/related/tcpip/>

---

## 2.5 Microfiche Support

There is no microfiche for the TCP/IP Feature for z/VM.

---

## 2.6 Publications Useful During Installation and Service

The publications listed in Figure 4 may be useful during the installation and service of the TCP/IP Feature for z/VM and its corresponding features. To order copies, contact your IBM representative.

*Figure 4. Publications Useful During Installation / Service on z/VM Version 3 Release 1.0*

<b>Publication Title</b>	<b>Form Number</b>
z/VM: VMSES/E Introduction and Reference	GC24-5947
z/VM: Service Guide	GC24-5946
z/VM: Planning and Administration	SC24-5948
z/VM: CMS Command Reference	SC24-5969
z/VM: CMS File Pool Planning, Administration, and Operation	SC24-5949
z/VM: System Messages and Codes	GC24-5974
z/VM: REXX/VM Reference	SC24-5963
C for VM/ESA: Library Reference	SC23-3908
z/VM: CMS Callable Services Reference	SC24-5959
z/VM: TCP/IP Level 3A0 Planning and Customization	SC24-5981
z/VM: TCP/IP Level 3A0 User's Guide	SC24-5982



---

## 3.0 Program Support

This section describes the IBM support available for the TCP/IP Feature for z/VM.

---

### 3.1 Preventive Service Planning

Before you install the TCP/IP Feature for z/VM, or any of its corresponding features, check with your IBM Support Center or use IBMLink™ to see if additional Preventive Service Planning (PSP) information is available that you should know. To obtain this information, specify the following UPGRADE and SUBSET values:

Figure 5. PSP Upgrade and Subset ID

---

Retain				
COMPID	Release	Upgrade	Subset	Component Name
5735FAL00	3A0	TCPIP3A0(1*)	VM3A0 (1*)	TCP/IP Feature for z/VM
5735FAL00	3A1	TCPIP3A0	VM3A0	TCP/IP Source Feature for z/VM
5735FAL00	3A0	TCPIP3A0	yynnRSU	RSU Service Recommendations
5735FAL00	3A0	TCPIP3A0	RSU-BY-PTF	Sorted by PTF
5735FAL00	3A0	TCPIP3A0	RSU-BY APAR	Sorted by APAR
5735FAL00	3A0	TCPIP3A0	RSU-BY-LEVEL	Sorted by RSU level

---

**Notes:**

1. These UPGRADE and SUBSET values are also used for the TCP/IP NFS Server Feature for z/VM.

---

### 3.2 Statement of Support Procedures

Report any difficulties you have using this program to your IBM Support Center. If an APAR (or, Authorized Program Analysis Report) is required, the Support Center will provide the address to which any needed documentation can be sent.

Figure 6 on page 10 identifies the Component ID (COMPID), Retain Release and Field Engineering Service Number (FESN) for the TCP/IP Feature for z/VM, and its corresponding features.

Figure 6. Component IDs

---

<b>Retain</b>			
<b>COMPID</b>	<b>Release</b>	<b>Component Name</b>	<b>FESN</b>
5735FAL00	3A0	TCP/IP Feature for z/VM (1*)	0461035
5735FAL00	3A1	TCP/IP Source Feature for z/VM	0461035

---

**Notes:**

1. The FESN for this component is also used for the TCP/IP NFS Server Feature for z/VM.

---

### 3.3 Service Information

The IBM Software Support Center provides you with telephone assistance in problem diagnosis and resolution. You can call the IBM Software Support Center at any time; you will receive a return call within eight business hours (Monday—Friday, 8:00 a.m.—5:00 p.m., local customer time). The number to call is:

**1-800-237-5511**

Outside of the United States or Puerto Rico, contact your local IBM representative or your authorized supplier.

Some installation and service-related information, such as the Preventive Service Planning (PSP) “bucket,” is also available on-line, via the TCP/IP Feature home page on the World Wide Web. The URL for this home page is:

<http://www.vm.ibm.com/related/tcpip/>

#### 3.3.1 Problem Documentation

When working with TCP/IP Feature support personnel on problems associated with an active Problem Management Record (PMR), diagnostic information may be occasionally requested. In such cases, the support staff will work with you to determine how to best provide the requested documentation. For reference, addresses that can be used to submit various documentation formats are listed in this section.

**Note:** The addresses that follow cannot be used as a problem reporting facility. All requests for problem assistance must be processed through the IBM Software Support Center, as previously described. Documentation submitted to any of these addresses will be reviewed only if it is associated with an active PMR.

Figure 7. Problem Documentation Addresses

---

<b>Format</b>	<b>Address</b>
Internet	vmtcpdoc@vnet.ibm.com
IBMLink	GDLVM7(TCPLVL2)
Carrier Service	IBM Corporation Attention: <i>IBM contact name</i> Dept. G79G 1701 North St. Endicott, NY 13760

---

### 3.3.2 Communicating Your Comments to IBM

If you have comments about or suggestions for the TCP/IP Feature for z/VM program product, or this Program Directory, you can provide them to IBM through one of the following:

- If you prefer to send comments by mail, use the address provided with the Reader's Comments form (RCF) at the back of this document.
- If you prefer to send comments electronically, use this Internet e-mail address:  
**vmtcpip@vnet.ibm.com**

When sending documentation-related comments, please include:

- the title of this publication
- the section title, section number, or topic to which your comment applies.

---

## 4.0 Program and Service Level Information

This section identifies the program level and any relevant service levels of the TCP/IP Feature for z/VM, and its corresponding features. The program level refers to the APAR fixes incorporated into the program. The service level refers to the PTFs shipped with this product. Information about the cumulative service tape is also provided.

---

### 4.1 Program Level Information - TCP/IP Feature for z/VM

The following APAR fixes against previous releases of the IBM TCP/IP for VM program product have been incorporated into this level of TCP/IP:

	PQ16190	PQ21811	PQ23644	PQ23801	PQ24344	PQ24872	PQ25155	PQ25284
	PQ26202	PQ26273	PQ26583	PQ26898	PQ26927	PQ26989	PQ27109	PQ27110
	PQ27112	PQ27156	PQ27338	PQ27851	PQ27876	PQ28043	PQ28058	PQ28138
	PQ28148	PQ28157	PQ28161	PQ28226	PQ28354	PQ28650	PQ28762	PQ28862
	PQ28922	PQ29376	PQ29464	PQ29594	PQ29597	PQ29669	PQ29740	PQ29745
	PQ29794	PQ29816	PQ29855	PQ29861	PQ29997	PQ30017	PQ30177	PQ30718
	PQ31079	PQ31128	PQ31152	PQ31407	PQ31576	PQ31964	PQ31993	PQ32201
	PQ32306	PQ32450	PQ32456	PQ32490	PQ32689	PQ32799	PQ32830	PQ32877
	PQ33001	PQ33069	PQ33236	PQ33869	PQ33928	PQ34052	PQ34228	PQ34272
	PQ34318	PQ34985	PQ35205	PQ35374	PQ35437	PQ35950	PQ36087	PQ36240
	PQ36325	PQ37002	PQ37131	PQ37393	PQ37520	PQ37902	PQ38029	PQ38300
	PQ38431	PQ38658	PQ38736	PQ38806	PQ39289	PQ39328	PQ39525	PQ39570
	PQ39574	PQ40435	PQ40494	PQ40826	PQ41301	PQ41584	PQ41591	PQ41683
	PQ41745	PQ41889	PQ42294	PQ42665	PQ44371			

---

### 4.2 Program Level Information - TCP/IP NFS Server Feature for z/VM

The following APAR fixes against previous releases of the IBM TCP/IP for VM NFS Feature program product have been incorporated into this level of TCP/IP:

	PQ27106	PQ27108	PQ27111	PQ27667	PQ32259	PQ33050	PQ35339	PQ35655
	PQ36105	PQ39650	PQ40839	PQ41048				

---

### 4.3 Service Level Information

Check the TCPIP3A0 PSP bucket for any additional PTFs that should be installed or any additional install information you should be aware of before you install and configure the TCP/IP Feature for z/VM, or any of its corresponding features.

---

### 4.4 Cumulative Service (RSU) Tape Information

Cumulative service for the TCP/IP Feature for z/VM is available through a periodic preventive service tape, the Recommended Service Upgrade (RSU). The current level of the TCP/IP Feature RSU can be obtained using the information provided in Figure 8:

*Figure 8. Cumulative Service (RSU) Tape Information*

---

<b>Retain</b>			
<b>COMPID</b>	<b>Release</b>	<b>APAR Number</b>	<b>PTF</b>
5735FAL00	RSU	<b>PQ44915</b>	<b>UQ99300</b>

**Note:** Cumulative service is not provided for the TCP/IP Source feature.

---

## 5.0 Installation Requirements and Considerations

The following sections identify system requirements for installing and activating the TCP/IP Feature for z/VM, and its corresponding features.

---

### 5.1 Hardware Requirements

There are no special hardware requirements to install the TCP/IP Feature for z/VM, or its features. Additional hardware requirements for exploiting specific functions of the TCP/IP Feature for z/VM and its features are documented in the announcement material and in *z/VM: TCP/IP Level 3A0 Planning and Customization (SC24-5981)*.

---

### 5.2 Program Considerations

The following sections list the programming considerations for installing the TCP/IP Feature for z/VM and its features, and activating their functions.

#### 5.2.1 Operating System Requirements

The TCP/IP Feature for z/VM requires the following operating system:

- z/VM Version 3 Release 1.0
- CMS Level 16

#### 5.2.2 Other Program Product Requirements

##### 5.2.2.1 Other Program Product Requirements - TCP/IP Feature for z/VM

IBM C for VM/ESA Version 3 Release 1 (5654-033) has been used to build the C components that provide the following TCP/IP Feature for z/VM services. To use these services, the Language Environment® run-time library must be available:

- Domain Name Server virtual machine (NAMESRV)
- Multi-Path Routing server (MPROUTE)
- Network Data Base servers (NDBPMGR and NDBSRVnn)
- Portmapper server (PORTMAP)
- Remote Execution daemon (REXECD and RXAGENTn)
- RouteD server (ROUTED)
- SNMP Query Engine and Agent (SNMPD and SNMPQE)
- Sockets Applications Programming Interface
- Network File System server (VMNFS)
- Kerberos Authentication and Administrator Servers (VMKERB and ADMSERV)

Various client functions require Language Environment run-time library support as well. Representative of these are:

- CMSRESOL and CMSRESXA
- DIG
- NSLOOKUP
- RPCGEN and RPCINFO
- NDBSRVS

The Language Environment run-time library necessary to use these C components is supplied with z/VM Version 3 Release 1.0.

**Note:** By default, the code for the above services is installed when you install the TCP/IP Feature for z/VM, regardless of whether or not you intend to use all, or only a subset, of these services.

Additional software requirements for exploiting specific functions of the TCP/IP Feature for z/VM are documented in the announcement material and in *z/VM: TCP/IP Level 3A0 Planning and Customization* (SC24-5981).

### 5.2.2.2 Other Program Product Requirements - TCP/IP NFS Server Feature for z/VM

The requirements to install and service the TCP/IP NFS Server Feature for z/VM are the same as those for the TCP/IP Feature for z/VM.

**Note:** The TCP/IP NFS Server Feature for z/VM is **not** required in order to use the NFS client support introduced with TCP/IP Level 3A0.

### 5.2.2.3 Other Program Product Requirements - SSL Server

To use the Secure Socket Layer (SSL) server, a suitably configured Linux kernel and file system must be installed on your z/VM system. Detailed information about Linux requirements and preparation for use by the SSL server are available at the TCP/IP Feature home page on the World Wide Web. The URL for this home page is:

<http://www.vm.ibm.com/related/tcpip/>

## 5.2.3 Program Installation/Service Considerations

This section describes items that should be considered before you install or service the TCP/IP Feature for z/VM and its features.

- You are **authorized to enable** the TCP/IP Feature only **if you have previously ordered this feature** and you are running z/VM Version 3 Release 1.0.
- VMSES/E is required to install and service this product.
- If multiple users install and maintain licensed products on your system, you may encounter problems getting the necessary access to the MAINT 51D minidisk. If you find there is contention for write access to the 51D minidisk, you can eliminate it by converting the Software Inventory from minidisk to

Shared File System (SFS). See the *z/VM: VMSES/E Introduction and Reference*, Chapter 18 “Changing the Software Inventory to an SFS Directory,” for information about how to make this change.

- The **3TCPIPA0** user ID is used to install and service the TCP/IP Feature for z/VM and its corresponding features. This is the IBM suggested user ID name. You are free to change this to any user ID name you wish; however, a PPF override to implement such a change must be created for the TCP/IP Feature for z/VM and each additional TCP/IP feature that you install.
- The TCP/IP Feature for z/VM, and its corresponding features, each have several defined components. The component you need to use for installation and service depends on whether you maintain the default minidisk installation environment or move the TCP/IP Feature service disks to Shared File System directories. For each feature, you need to choose one of the listed components that follow.

**Note! Minidisk Requirement**

Certain minidisks **must** be defined and used with TCP/IP server machines, even when the TCP/IP Feature service minidisks are moved to Shared File System directories. This requirement is explained further in item 5 of 5.3, “DASD Storage and User ID Requirements” on page 23.

The components for the TCP/IP Feature for z/VM are:

- **TCPIP** - install and service the TCP/IP Feature using minidisks
- **TCPIPSFS** - install and service the TCP/IP Feature using the Shared File System

The components for the TCP/IP Source Feature for z/VM are:

- **SOURCE** - if the TCP/IP Feature is installed and serviced using minidisks
- **SOURCESFS** - if the TCP/IP Feature is installed and serviced using the Shared File System
- During the installation and service processes, you may need to perform additional steps to account for errors that cannot be corrected through the service process. If such steps are required, additional notes and text which explain those steps will be included in updated levels of this document.

Several additional considerations for the TCP/IP Source and TCP/IP NFS Server features follow:

- You are **authorized to enable** the TCP/IP NFS Server Feature only **if you have previously ordered** the TCP/IP *and* TCP/IP NFS Server features, and you are running z/VM Version 3 Release 1.0.
- Enable the TCP/IP NFS Server feature using the TCP/IP Feature installation and service user ID, **3TCPIPA0**, or your chosen equivalent.
- Install the TCP/IP Source feature using the TCP/IP Feature installation and service user ID, **3TCPIPA0**, or your chosen equivalent.
- The TCP/IP Source feature uses the same installation and service minidisks and SFS directories as the TCP/IP Feature for z/VM.
- **Before** the TCP/IP Source Feature is installed, you **must** increase the storage allocated for the **SOURCE** minidisk (**3TCPIPA0 2B3**, by default) or its equivalent SFS directory to meet the requirements listed in Figure 11 on page 32.



- Source files are shipped in packed format. You will need to unpack these files before you can use them.

Additional considerations for using SSL server support follow:

- To use the Secure Socket Layer (SSL) server, a suitably configured Linux kernel and file system must be installed on your z/VM system. Detailed information about Linux requirements and preparation for use by the SSL server are available at the TCP/IP Feature home page on the World Wide Web. The URL for this home page is:

<http://www.vm.ibm.com/related/tcpip/>

## 5.2.4 Migration Considerations

In addition to the introduction of new TCP/IP functions, there have been changes to some existing functions. In a few cases, functions have been deleted or replaced by alternative functions. This section describes these changes or deletions.

TCP/IP and related features are supported on associated releases of CP and CMS only.

### 5.2.4.1 Packaging

#### 5.2.4.1.1 General Information About TCP/IP Level 3A0

- TCP/IP Level 3A0 is included as a priced feature of the z/VM product, requiring that you have a license from IBM for its use. The TCP/IP feature **must be enabled** using the correct feature code before it can be used.
- The Network File System (NFS) server is a separately licensed and priced feature of z/VM. Like the base TCP/IP feature, it is included with the z/VM product, but **it must be enabled** before it can be used.

**Note:** Information about enabling the TCP/IP and TCP/IP NFS Server features is included within this document.

- TCP/IP Level 3A0 is **not** separately orderable from the z/VM product, although it is serviced separately from z/VM.
- This level of TCP/IP relies on the presence of certain functions in CP and CMS. Abends and incorrect results are possible if you attempt to use it with an older level of CP or CMS.
- TCP/IP softcopy publications are included with the other z/VM softcopy publications.

#### 5.2.4.1.2 Changes Introduced in TCP/IP Level 3A0

- DES Kerberos functions are now incorporated in the base TCP/IP Feature (non-DES Kerberos functions are no longer available). Therefore, customers who have a Kerberos database that was created in a non-DES environment **must** delete and then rebuild that database using the supplied DES Kerberos functions. Refer to the chapter titled “Configuring the Kerberos Server” in *z/VM: TCP/IP Level 3A0 Planning and Customization* for more information about building the Kerberos database.

- The various servers and code that provide support for the Network Computing System (NCS) are no longer provided.
- The installation user ID-owned “Sample” (2C2) minidisk and its corresponding sample files are no longer provided. The sample files provided on this disk in prior levels of TCP/IP can be obtained via the IBM TCP/IP for VM Internet home page, for which the URL is:

<http://www.vm.ibm.com/related/tcpip/>

#### **5.2.4.1.3 Changes Introduced in TCP/IP Function Level 310**

- TCP/IP-based user e-mail functions are included in CMS. TCP/IP-specific versions of NOTE and SENDFILE are **not** provided with the TCP/IP Feature.
- A subset of RSCS Version 3 Release 2 function is available with the z/VM product to provide enhanced TCP/IP client and server print capabilities. RSCS functions not related to TCP/IP printing enhancements require a separate RSCS license for use. Refer to the *RSCS Version 3 Release 2 Program Directory* for additional information.
- Non-DES Kerberos functions are included in the base TCP/IP Feature. DES Kerberos functions are available as a separate feature that must be separately installed.

### **5.2.4.2 Server Configuration**

#### **5.2.4.2.1 Changes Introduced in TCP/IP Function Level 310**

- Separate server initialization execs (TCPIPXIT, FTPDEXIT, etc.) are no longer used. All server parameters and features are controlled by entries contained in a DTCPARMS file. There is support for you to supply an exit which is specific to a particular server, or an exit that is used by all servers, for customization needs that cannot be met using a customized SYSTEM DTCPARMS file.
- Changing server names (when defining a second set of TCP/IP servers, for example) no longer requires changes to IBM-provided execs. All information related to the user ID of a particular server is kept in the TCPIP DATA file, or is part of the server configuration and is maintained in the DTCPARMS file.
- External Security Manager (ESM) interfaces have been standardized across all servers. The RPIUCMS command is used to initialize the RACROUTE environment, and RPIVAL is used to validate user IDs and passwords supplied by clients. These commands can be changed using a customized SYSTEM DTCPARMS file.
- The ESM environment is automatically established for a server when :ESM\_Enable.YES is specified for that server in a customized SYSTEM DTCPARMS file.

Refer to the chapter titled “TCP/IP Server Configuration” in *z/VM: TCP/IP Level 3A0 Planning and Customization* for more information.

### 5.2.4.3 TCPIP Server

#### 5.2.4.3.1 Changes Introduced in TCP/IP Function Level 320

- The **TIMESTAMP** statement default has been changed from **TIMESTAMP 0** to **TIMESTAMP PREFIX**, which specifies that a time stamp is to preface every trace and console message. This change helps in diagnosing problems and isolating error conditions. Therefore, it is recommended that any existing TCP/IP server configuration files be changed to specify **TIMESTAMP PREFIX** to aid in problem determination.
- The Telnet session connection exit interface has been changed to pass the LU name supplied by a client (if any) as an additional parameter. Existing exits may need to be changed to accommodate this behavior.
- The Telnet printer management exit is called for any printer session, regardless of whether the client LU name and IP address are defined by a **TN3270E** statement in the TCP/IP configuration file. Existing exits may need to be changed to accommodate this behavior.

### 5.2.4.4 RouteD/MPROUTE Server

#### 5.2.4.4.1 Changes Introduced in TCP/IP Level 3A0

- The RouteD and MPROUTE servers cannot be concurrently used with the same TCP/IP stack server.

#### 5.2.4.4.2 Changes Introduced in TCP/IP Function Level 320

- The **#CP EXTERNAL** command can no longer be used to terminate RouteD server. The command to stop RouteD is **#CP SMSG \* SHUTDOWN**.

### 5.2.4.5 DNS Server

#### 5.2.4.5.1 Changes Introduced in TCP/IP Level 3A0

- The use of a cache file (as used with a **CACHINGONLY** name server configuration) has been expanded to the **PRIMARY** and **SECONDARY** configurations. A new **ROOTCACHE** statement allows a cache file to be specified in a manner similar to that which can be specified on the **CACHINGONLY** statement. A corresponding sample root cache file, **NSMAIN SCACHE**, supplies several configuration recommendations, a list of root name servers, and the Internet site from which the most current list can be obtained.
- A **FORWARDERS** statement has been added to improve name resolution capability for z/VM hosts which are connected to other hosts that provide network firewall services. This statement can be used to identify other name server hosts which can perform name resolution outside the firewall system.
- The algorithms used in the caching subsystem have been improved to facilitate faster access to cached information and to more quickly determine when cached information does not exist.

## 5.2.4.6 SMTP Server

### 5.2.4.6.1 Changes Introduced in TCP/IP Level 3A0

- Timing values for the **RETRYAGE** and **WARNINGAGE** statements can now be specified in hours or minutes (in addition to a number of *days*), through the use of additional **H** or **M** statement operands. Existing defaults (specified in days, for which a **D** statement operand can now be specified) remain unchanged.

### 5.2.4.6.2 Changes Introduced in TCP/IP Function Level 320

- Support for the SMTP **EHLO** command has been added, as has support for the Message Size Declaration and 8-bit MIME transport service extensions. With this support, SMTP now accepts and handles the **SIZE** and **BODY** parameters on **MAIL FROM:** commands.

### 5.2.4.6.3 Changes Introduced in TCP/IP Function Level 310

- The **DEBUG** configuration file statement is no longer supported. This statement has been replaced by the **TRACE** statement which, when specified with the **DEBUG** parameter, provides identical function as the **DEBUG** statement, except that output goes to only the console; there is no debug file support. Refer to the chapter titled “Configuring the SMTP Server” in *z/VM: TCP/IP Level 3A0 Planning and Customization* for more information about the **TRACE** statement and additional parameters that can be specified.
- The default for the **WARNINGAGE** configuration file statement has been reduced from 3 days to 1 day.
- Due to the introduction of new and changed trace options, the **TRACE SMSG** command now requires additional options to be specified. In TCP/IP for VM releases prior to TCP/IP Function Level 310, the **TRACE SMSG** command was used to trace host name resolution via name servers. This **SMSG** command has been replaced with the **TRACE RESOLVER SMSG** command. Similar results can be achieved by specifying the **TRACE RESOLVER** statement in the SMTP configuration file. Refer to the chapter titled “Configuring the SMTP Server” in *z/VM: TCP/IP Level 3A0 Planning and Customization* for more information about configuration file changes, and to the *z/VM: TCP/IP Level 3A0 User's Guide* for SMTP **SMSG** command interface changes.
- The SMTP **DATA** file is no longer needed. The **ATSIGN** statement previously supported using this file has been incorporated within the **TCPIP DATA** file.
- The sample SMTP configuration file (**SMTP SCONFIG**) now specifies “**NETDATA**” as the default for both the **LOCALFORMAT** and **RSCSFORMAT** statements, to reflect the operational default value associated with these statements.
- Host names for mail items can now be re-resolved (when required) by using the **REPROCESS SMSG** command. Refer to the chapter titled “Configuring the SMTP Server” in *z/VM: TCP/IP Level 3A0 Planning and Customization* for more information. The formerly documented **SMTP-EXP EXEC** is no longer needed for this purpose and **should not be used**.
- SMTP work files (**NOTE** and **ADDRBLOK** files) have new formats and names; the file types for these files have been changed to **NOTEFILE** and **ADDRFILE**, respectively. When migrating from IBM TCP/IP for VM Version 2 Release 4, any **NOTE** or **ADDRBLOK** files present on the SMTP server

A-disk are converted to the new format and renamed; such converted files cannot be processed by previous versions of TCP/IP for VM.

## 5.2.4.7 FTP Server

### 5.2.4.7.1 Changes Introduced in TCP/IP Level 3A0

- The FTP server now makes use of a server-specific configuration file (**SRVRFTP CONFIG** by default) for its startup information, in a manner more consistent with other TCP/IP servers. Therefore, **:parms.** entries within the DTCPARMS file that are associated with the FTP server need be modified to account for this change. Any SRVRFTP operands currently specified using a **:parms.** entry need to be removed, with appropriate modifications made to either the FTP server configuration file or the DTCPARMS file (for example, to account for use of the ANONYMOU or RACF command operands).  
In addition, the FTP server no longer makes use of the client-oriented FTP DATA file.
- The FTP server has been enhanced to better accommodate web browser and graphical FTP clients. Changes for this support include the ability to provide Unix-format list information in response to client DIR subcommand requests, through the use of the **LISTFORMAT** configuration statement. In addition, the **AUTOTRANS** statement can be used to configure the server to perform automatic EBCDIC-ASCII data translation for transferred files, based on file types (or, file extensions), as determined by **VMFILETYPE** statements which are maintained in the TCPIP DATA file.
- Support for **AUTOTRANS** and **LISTFORMAT** operands to client-supplied SITE subcommands has been implemented as well, to provide client control over automatic EBCDIC-ASCII data translation and list information formats.
- **AUTOTRANS** and **LISTFORMAT** characteristics can now be established when specific FTP users login, when the CHKIPADR exit exec is customized for use.
- Support for additional SMSG interface commands has been added to allow for dynamic server configuration changes, console and tracing control, and shutdown/restart capability.

### 5.2.4.7.2 Changes Introduced in TCP/IP Function Level 320

- The FTP server has been enhanced to exploit new CP and CMS user authorization facilities provided with VM/ESA Version 2 Release 4.0. These enhancements allow an FTP user to access minidisks they own without the need for minidisk passwords to be defined or supplied during an FTP session. Thus, the link results achieved when FTP users access minidisk resources may noticeably differ from those seen using prior levels of TCP/IP for VM.

For example, if a user establishes a read-only link to a minidisk (through the use of an explicit "**ACCT read\_password**" command), a subsequent PUT command that initiates a write request to that minidisk may now succeed *if* the login user ID owns the minidisk in question. By comparison, with a prior TCP/IP level, such a request would cause an FTP user to be prompted to supply a Read/Write (or Multiple Read) password through use of the **ACCT** subcommand in order to first gain write access to the minidisk.

The use of these new user authorization facilities also allows users with LOGONBY authority for a given *base* virtual machine to exercise that same authority during an FTP session. This is accomplished through use of the CP Diagnose X'88' command by the FTP server, which allows

access to *base* user ID resources without requiring the password for that *base* user ID to be supplied. This kind of access is achieved through use of a new “/BY/*byuser\_id*” parameter of the FTP **USER** subcommand. Additionally, the CP directory entry for the FTP server must include an OPTION DIAG88 statement, to allow use of the Diagnose X'88' command.

- FTP virtual reader support has been added, which allows an FTP client to use the virtual reader of a VM user ID as the current directory. For such users to issue DELETE and DIR or LS commands against a reader directory, the FTP server virtual machine must have class D privileges. To allow user to PUT files to a reader directory, the **RDR** parameter must be included with the FTP server startup command, SRVRFTP.

## 5.2.4.8 FTP Client

### 5.2.4.8.1 Changes Introduced in TCP/IP Level 3A0

- The FTP client includes support for the **SIZE** subcommand, which allows a client to obtain the size of a file before it is retrieved.

### 5.2.4.8.2 Changes Introduced in TCP/IP Function Level 320

- The FTP client has been enhanced to make use of login information present in a NETRC DATA file. By default, when a connection is made to a foreign host, user ID and password information in the NETRC DATA file that is specific to that host are used as part of an automatic FTP login process. Automatic FTP login can be suppressed using the new **NONETRC** option, or through use of the new **NETRC** subcommand.

## 5.2.4.9 Printing

### 5.2.4.9.1 Changes Introduced in TCP/IP Function Level 310

- The LPR command can now route print files to RSCS for printing, freeing a user's virtual machine for other work. This may introduce the need for users to learn a limited set of RSCS commands in order to determine the status of their print files. Refer to the *z/VM: TCP/IP Level 3A0 User's Guide* for more information.
- RSCS can be used instead of LPSERVE to provide enhanced printer daemon (LPD) capabilities. The operation and administration characteristics of RSCS are very different from LPSERVE. Refer to the chapter titled “Configuring the RSCS Print Server” in *z/VM: TCP/IP Level 3A0 Planning and Customization* for more information.

## 5.2.4.10 Kerberos Server

### 5.2.4.10.1 Changes Introduced in TCP/IP Level 3A0

- The non-DES Kerberos functions that were provided with previous levels of IBM TCP/IP for VM are no longer available. Instead, only a DES Kerberos system is supported, with the DES Kerberos functions incorporated as part of the TCP/IP Feature for z/VM.

Any Kerberos database that was created in a non-DES environment will not work with the DES Kerberos functions supplied as part of TCP/IP Feature for z/VM. The existing non-DES Kerberos

database **must** be deleted and then rebuilt using the supplied DES Kerberos functions. Refer to the chapter titled “Configuring the Kerberos Server” in *z/VM: TCP/IP Level 3A0 Planning and Customization* for more information about building the Kerberos database.

---

## 5.3 DASD Storage and User ID Requirements

Figure 10 on page 26 lists the user IDs and minidisks used to install and service the TCP/IP Feature for z/VM, and its corresponding features. Figure 11 on page 32 identifies additional minidisk requirements that are unique for the TCP/IP Source Feature.

### Important Installation Notes:

1. The user IDs necessary for all TCP/IP features have been defined as part of the z/VM Version 3 Release 1.0 System DDR that is already installed. Likewise, all required minidisks have been defined as well. These resources have been listed in Figure 10 and Figure 11 so you are aware of the resources that have been allocated on your behalf.

For information about specific user ID directory entry requirements, consult the **3TCPIPA0 PLANINFO** file. This file is located on the 3TCPIPA0 191 minidisk.

2. **3TCPIPA0** is the default installation and service user ID, but this can be changed. If you choose to change the name of this installation user ID, you need to create a Product Parameter (PPF) override.

**Note:** If you change the TCP/IP Feature for z/VM installation user ID to a common or existing user ID (such as MAINT), minidisks may already be defined with addresses which are identical to the TCP/IP Feature minidisk defaults. If any minidisks are in use that have such matching addresses, you will need to create a PPF override to change the default TCP/IP Feature minidisk addresses so they are unique within your environment.

3. The 3TCPIPA0, TCPMAINT, and TCPIP user IDs and minidisks must be maintained. The remaining user IDs listed in Figure 10 are for servers and clients associated with optional TCP/IP Feature services. If you choose to not use a particular optional service, you don't need to maintain the user IDs and production minidisks associated with that service.
4. If you choose to use user IDs for TCP/IP server virtual machines that differ from the IBM-supplied user IDs shown in Figure 10, ensure you review the section titled “Implications of Assigning Different Server Virtual Machine Names” in Chapter 1 of *z/VM: TCP/IP Level 3A0 Planning and Customization*.
5. Note the following, with regard to the user ID and minidisk information provided in Figure 10 and Figure 11:

### Specific Minidisk Requirements

Certain minidisks **must** be defined for the TCP/IP server machines you choose to use, as well as for maintaining the TCP/IP Feature in your environment. Such minidisks **cannot** be replaced with an SFS directory.

Minidisks to which this requirement applies are listed in Figure 10 and Figure 11 with **boldface** virtual addresses. In addition, no default SFS directory names or SFS 4K block values are provided for these minidisks.

6. **All** 3TCPIPA0 test build minidisks **must** be maintained. If the 3TCPIPA0 minidisks for optional services are not maintained, problems will be encountered during installation and service.

To determine which services (if any) you do not want to use, review the descriptions of services provided in *z/VM: TCP/IP Level 3A0 Planning and Customization (SC24-5981)*. The server-specific configuration chapters of this publication may also help you with your determination.

Also, if you omit resources that correspond to an optional service, you should not attempt to place the files associated with that service into production, during step 6.2.4.2, “Copy TCP/IP Feature Run-time Files Into Production” on page 41. The 3TCPIPA0 CATALOG file should be modified so that files not used in your environment are bypassed during TCP2PROD command processing. See Appendix A, “The TCP2PROD Command” on page 93 for information about the TCP2PROD command and TCP/IP Feature catalog files.

7. Additional storage may need to be allocated for a given user ID or server minidisk, depending on your environment. Some examples of minidisks that may need to be increased, and possible reasons for so doing, are listed in Figure 9. Note that certain minidisks (not cited here) may also need to be increased to accommodate the logging of tracing or other activities.

Figure 9. Alternative Minidisk Storage Requirements

User ID / Minidisk	Rationale for Storage Revision
SMTP 191	Allow for SMTP processing of a high volume of e-mail
VMNFS 191	Provide support for a large number of NFS clients
DHCPD 191	Provide support a large number of DHCP clients
LPSERVE 191	Allow for LPD processing of sizeable print jobs
SSLSERV 201	Facilitate installation of a Linux kernel and file system
3TCPIPA0 2D2	Facilitate maintenance of multiple service levels of (binary) system files used by the SSL server

8. The storage requirements for various TCP/IP minidisks may be revised over time to account for TCP/IP Feature content changes. In some cases, this may require the size of existing minidisks to be increased.

When applicable, storage requirement changes for specific minidisks will be identified in Figure 10 and Figure 11, in updated levels of this document. Such changes will be noted through the use of revision characters (usually a vertical bar — “I” in the left margin of a page).



For other minidisks, storage requirement changes will need to be assessed locally, for your specific environment. For example, the capacity of the **DELTA** minidisk (**3TCPIPA0 2D2**, by default), periodically may need to be increased, based on the specific preventive and corrective service applied to your system.

9. You can create multiple virtual machines, named NDBSRV01, NDBSRV02, etc., if you choose to run the Network Database. Each NDBSRV $n$  virtual machine should be defined similar to NDBSRV01, which has been defined as part of the z/VM Version 3 Release 1.0 System DDR that is already installed.
10. You can create multiple virtual machines, named RXAGENT1, RXAGENT2, etc., if you choose to run the remote execution daemon (REXECD). Each RXAGENT $n$  virtual machine should be defined similar to RXAGENT1, which has been defined as part of the z/VM Version 3 Release 1.0 System DDR that is already installed. However, note that the RXAGENT $n$  virtual machines do not own any minidisks.
11. To use the Secure Socket Layer (SSL) server, a suitably configured Linux kernel and file system must be installed on your z/VM system. Detailed information about Linux requirements and preparation for use by the SSL server are available at the TCP/IP Feature home page on the World Wide Web. The URL for this home page is:  
**<http://www.vm.ibm.com/related/tcpip/>**
12. The TCP/IP Source feature uses the same installation and service minidisks and SFS directories as the TCP/IP Feature for z/VM.
13. **Before** the TCP/IP Source Feature is installed, you **must** increase the storage allocated for the **SOURCE** minidisk (**3TCPIPA0 2B3**, by default) or its equivalent SFS directory to meet the requirements listed in Figure 11 on page 32.
14. Source files are shipped in packed format. If you intend to unpack the source files after installation, ensure you define a sufficiently large SOURCE minidisk or directory, as specified by the disk size for unpacked files in Figure 11 on page 32.
15. For information about copying client code to the Product Code minidisk, see Appendix H, “Copying TCP/IP Feature Client Code to the Y-Disk” on page 127.

### 5.3.1 DASD Requirements for the TCP/IP Feature for z/VM

Figure 10 (Page 1 of 6). DASD Storage Requirements for Target Minidisks - TCP/IP Feature

Minidisk owner (User ID)	Default Address	Storage in Cylinders		FB-512 Blocks	SFS 4K Blocks	Usage
		DASD	CYLS			Default SFS Directory Name
3TCPIPA0	191	3390 3380 9345	50 60 60	72000	9000	3TCPIPA0 user ID 191 minidisk  <b>VMSYS:3TCPIPA0</b>
3TCPIPA0	2B2	3390 3380 9345	78 94 94	112000	14000	Contains all the base code shipped with the TCP/IP Feature for z/VM  <b>VMSYS:3TCPIPA0.TCPIP.OBJECT</b>
3TCPIPA0	2B3	3390 3380 9345	2 2 2	2400	300	Source files disk; only VMSES/E build list source files are included with the TCP/IP Feature (3*)  <b>VMSYS:3TCPIPA0.TCPIP.SOURCE</b>
3TCPIPA0	2C4	3390 3380 9345	5 5 5	6000	750	Contains local modifications  <b>VMSYS:3TCPIPA0.TCPIP.LOCAL</b>
3TCPIPA0	2D2	3390 3380 9345	117 140 140	167200	20900	Contains serviced files (4*)  <b>VMSYS:3TCPIPA0.TCPIP.DELTA</b>
3TCPIPA0	2A6	3390 3380 9345	5 5 5	6000	750	Contains AUX files and software inventory tables that represent the test service level of the TCP/IP Feature  <b>VMSYS:3TCPIPA0.TCPIP.APPLYALT</b>

**Notes:**

1. Cylinder values defined in this table are based on a 4K block size. FB-512 block and SFS values are derived from the 3380 cylinder values in this table. FBA minidisk sizes are shown in 512-byte blocks; these minidisks should be CMS formatted at 1K size.
2. For installation to SFS directories, a total of **136950** 4K blocks are required.
3. Additional space **must** be defined or allocated for the Source files minidisk (2B3) or directory if the TCP/IP Source Feature is installed. See Figure 11 on page 32 for more information about additional requirements for this resource.
4. Additional storage may need to be allocated for some minidisks, depending on your environment. For more information, see the accompanying notes on page 23.
5. See Appendix H, "Copying TCP/IP Feature Client Code to the Y-Disk" on page 127 for information about copying client code to the Product Code minidisk

Figure 10 (Page 2 of 6). DASD Storage Requirements for Target Minidisks - TCP/IP Feature

Minidisk owner (User ID)	Default Address	Storage in Cylinders		FB-512 Blocks	SFS 4K Blocks	Usage
		DASD	CYLS			Default SFS Directory Name
3TCPIPA0	2A2	3390 3380 9345	5 5 5	6000	750	Contains AUX files and software inventory tables that represent the service level of the TCP/IP Feature that is currently in production  <b>VMSYS:3TCPIPA0.TCPIP.APPLYPROD</b>
3TCPIPA0	<b>491</b>	3390 3380 9345	38 45 45	53600	_____	Test build disk for server code; files from this disk are copied to a production disk (TCPMAINT 591) which also requires this amount of free space
3TCPIPA0	<b>492</b>	3390 3380 9345	67 80 80	96000	_____	Test build disk for client code; files from this disk are copied to a production disk (TCPMAINT 592) which also requires this amount of free space
TCPMAINT	<b>191</b>	3390 3380 9345	7 8 8	9600	_____	TCPMAINT user ID 191 minidisk
TCPMAINT	<b>198</b>	3390 3380 9345	9 10 10	12000	_____	Contains configuration files for clients and servers.
TCPMAINT	<b>591</b>	3390 3380 9345	38 45 45	53600	_____	Production build disk for server code
TCPMAINT	<b>592</b>	3390 3380 9345	67 80 80	96000	_____	Production build disk for client code

**Notes:**

1. Cylinder values defined in this table are based on a 4K block size. FB-512 block and SFS values are derived from the 3380 cylinder values in this table. FBA minidisk sizes are shown in 512-byte blocks; these minidisks should be CMS formatted at 1K size.
2. For installation to SFS directories, a total of **136950** 4K blocks are required.
3. Additional space **must** be defined or allocated for the Source files minidisk (2B3) or directory if the TCP/IP Source Feature is installed. See Figure 11 on page 32 for more information about additional requirements for this resource.
4. Additional storage may need to be allocated for some minidisks, depending on your environment. For more information, see the accompanying notes on page 23.
5. See Appendix H, "Copying TCP/IP Feature Client Code to the Y-Disk" on page 127 for information about copying client code to the Product Code minidisk

Figure 10 (Page 3 of 6). DASD Storage Requirements for Target Minidisks - TCP/IP Feature

Minidisk owner (User ID)	Default Address	Storage in Cylinders		FB-512 Blocks	SFS 4K Blocks	Usage
		DASD	CYLS			Default SFS Directory Name
ADMSEV	191	3390 3380 9345	5 5 5	6000	_____	ADMSEV user ID 191 minidisk
BOOTPD	191	3390 3380 9345	2 2 2	2400	_____	BOOTPD user ID 191 minidisk
DHCPD	191	3390 3380 9345	2 2 2	2400	_____	DHCPD user ID 191 minidisk (4*)
FTPSERVE	191	3390 3380 9345	9 10 10	12000	_____	FTPSERVE user ID 191 minidisk
LPSEV	191	3390 3380 9345	2 2 2	2400	_____	LPSEV user ID 191 minidisk (4*)
MROUTE	191	3390 3380 9345	2 2 2	2400	_____	MROUTE user ID 191 minidisk
NAMESRV	191	3390 3380 9345	2 2 2	2400	_____	NAMESRV user ID 191 minidisk
NDBPMGR	191	3390 3380 9345	1 1 1	1200	_____	NDBPMGR user ID 191 minidisk

**Notes:**

1. Cylinder values defined in this table are based on a 4K block size. FB-512 block and SFS values are derived from the 3380 cylinder values in this table. FBA minidisk sizes are shown in 512-byte blocks; these minidisks should be CMS formatted at 1K size.
2. For installation to SFS directories, a total of **136950** 4K blocks are required.
3. Additional space **must** be defined or allocated for the Source files minidisk (2B3) or directory if the TCP/IP Source Feature is installed. See Figure 11 on page 32 for more information about additional requirements for this resource.
4. Additional storage may need to be allocated for some minidisks, depending on your environment. For more information, see the accompanying notes on page 23.
5. See Appendix H, "Copying TCP/IP Feature Client Code to the Y-Disk" on page 127 for information about copying client code to the Product Code minidisk

Figure 10 (Page 4 of 6). DASD Storage Requirements for Target Minidisks - TCP/IP Feature

Minidisk owner (User ID)	Default Address	Storage in Cylinders		FB-512 Blocks	SFS 4K Blocks	Usage
		DASD	CYLS			Default SFS Directory Name
NDBSRV01	191	3390 3380 9345	1 1 1	1200	_____	NDBSRV01 user ID 191 minidisk
PORTMAP	191	3390 3380 9345	2 2 2	2400	_____	PORTMAP user ID 191 minidisk
REXECD	191	3390 3380 9345	2 2 2	2400	_____	REXECD user ID 191 minidisk
ROUTED	191	3390 3380 9345	2 2 2	2400	_____	ROUTED user ID 191 minidisk
RXAGENT1	_____	3390 3380 9345	— — —	_____	_____	REXEC agent (a 191 minidisk is <b>not</b> required; REXEC agents utilize the REXECD 191 minidisk)
SMTP	191	3390 3380 9345	25 30 30	36000	_____	SMTP user ID 191 minidisk (4*)
SNALNKA	191	3390 3380 9345	3 3 3	3600	_____	SNALNKA user ID 191 minidisk
SNMPD	191	3390 3380 9345	2 2 2	2400	_____	SNMPD user ID 191 minidisk

**Notes:**

1. Cylinder values defined in this table are based on a 4K block size. FB-512 block and SFS values are derived from the 3380 cylinder values in this table. FBA minidisk sizes are shown in 512-byte blocks; these minidisks should be CMS formatted at 1K size.
2. For installation to SFS directories, a total of **136950** 4K blocks are required.
3. Additional space **must** be defined or allocated for the Source files minidisk (2B3) or directory if the TCP/IP Source Feature is installed. See Figure 11 on page 32 for more information about additional requirements for this resource.
4. Additional storage may need to be allocated for some minidisks, depending on your environment. For more information, see the accompanying notes on page 23.
5. See Appendix H, "Copying TCP/IP Feature Client Code to the Y-Disk" on page 127 for information about copying client code to the Product Code minidisk

Figure 10 (Page 5 of 6). DASD Storage Requirements for Target Minidisks - TCP/IP Feature

Minidisk owner (User ID)	Default Address	Storage in Cylinders		FB-512 Blocks	SFS 4K Blocks	Usage
		DASD	CYLS			Default SFS Directory Name
SNMPQE	191	3390 3380 9345	2 2 2	2400	_____	SNMPQE user ID 191 minidisk
SSLSERV	191	3390 3380 9345	1 1 1	1200	_____	SSLSERV user ID 191 minidisk
SSLSERV	201	3390 3380 9345	1 1 1	1200	_____	SSLSERV user ID 201 minidisk (4*)
SSLSERV	202	3390 3380 9345	30 36 36	42400	_____	SSLSERV user ID 202 minidisk
SSLSERV	203	3390 3380 9345	1 2 2	1400	_____	SSLSERV user ID 203 minidisk
SSLSERV	204	3390 3380 9345	15 17 17	20400	_____	SSLSERV user ID 204 minidisk
SSLSERV	205	3390 3380 9345	140 168 168	200800	_____	SSLSERV user ID 205 minidisk
TCPIP	191	3390 3380 9345	5 5 5	6000	_____	TCPIP user ID 191 minidisk

**Notes:**

1. Cylinder values defined in this table are based on a 4K block size. FB-512 block and SFS values are derived from the 3380 cylinder values in this table. FBA minidisk sizes are shown in 512-byte blocks; these minidisks should be CMS formatted at 1K size.
2. For installation to SFS directories, a total of **136950** 4K blocks are required.
3. Additional space **must** be defined or allocated for the Source files minidisk (2B3) or directory if the TCP/IP Source Feature is installed. See Figure 11 on page 32 for more information about additional requirements for this resource.
4. Additional storage may need to be allocated for some minidisks, depending on your environment. For more information, see the accompanying notes on page 23.
5. See Appendix H, "Copying TCP/IP Feature Client Code to the Y-Disk" on page 127 for information about copying client code to the Product Code minidisk

Figure 10 (Page 6 of 6). DASD Storage Requirements for Target Minidisks - TCP/IP Feature

Minidisk owner (User ID)	Default Address	Storage in Cylinders		FB-512 Blocks	SFS 4K Blocks	Usage
		DASD	CYLS			Default SFS Directory Name
TFTPD	191	3390 3380 9345	2 2 2	2400	_____	TFTPD user ID 191 minidisk
UFTD	191	3390 3380 9345	2 2 2	2400	_____	UFTD user ID 191 minidisk
VMKERB	191	3390 3380 9345	6 7 7	8400	_____	VMKERB user ID 191 minidisk
VMNFS	191	3390 3380 9345	9 10 10	12000	_____	VMNFS user ID 191 minidisk (4*)
X25IPI	191	3390 3380 9345	2 2 2	2400	_____	X25IPI user ID 191 minidisk

**Notes:**

1. Cylinder values defined in this table are based on a 4K block size. FB-512 block and SFS values are derived from the 3380 cylinder values in this table. FBA minidisk sizes are shown in 512-byte blocks; these minidisks should be CMS formatted at 1K size.
2. For installation to SFS directories, a total of **136950** 4K blocks are required.
3. Additional space **must** be defined or allocated for the Source files minidisk (2B3) or directory if the TCP/IP Source Feature is installed. See Figure 11 on page 32 for more information about additional requirements for this resource.
4. Additional storage may need to be allocated for some minidisks, depending on your environment. For more information, see the accompanying notes on page 23.
5. See Appendix H, "Copying TCP/IP Feature Client Code to the Y-Disk" on page 127 for information about copying client code to the Product Code minidisk

### 5.3.2 DASD Requirements for the TCP/IP Source Feature for z/VM

Figure 11. DASD Storage Requirements for Target Minidisks - TCP/IP Source Feature for z/VM

Minidisk owner (User ID)	Default Address	Storage in Cylinders		FB-512 Blocks	SFS 4K Blocks	Usage
		DASD	CYLS			Default SFS Directory Name
3TCPIPA0	2B3	3390	39	52800	6950	Source files disk - <b>PACKED</b> format  <b>VMSYS:3TCPIPA0.TCPIP.SOURCE</b>
		3380	47			
		9345	47			
3TCPIPA0	2B3	3390	80	114080	14260	Source files disk - <b>UNPACKED</b> format  <b>VMSYS:3TCPIPA0.TCPIP.SOURCE</b>
		3380	96			
		9345	96			

**Note:** Cylinder values defined in this table are based on a 4K block size. FB-512 block and SFS values are derived from the 3380 cylinder values in this table. FBA minidisk sizes are shown in 512-byte blocks; these minidisks should be CMS formatted at 1K size.

**6950** 4K blocks are needed for SFS install of the **packed** source files. **14260** 4K blocks are needed for SFS install if source files will be **unpacked**.



---

## 6.0 Installation Instructions

This section describes the installation methods and the step-by-step procedures to complete the installation and activation of the TCP/IP Feature for z/VM, as well as those necessary to install or activate the TCP/IP Source and TCP/IP NFS Server features.

### Authorization and Ordering Information

- If you **have not ordered** the **TCP/IP Feature for z/VM**, installation of the TCP/IP Feature, and any corresponding TCP/IP features, is complete; the additional installation steps provided in this document should not be used.
- If you **are planning to order** the TCP/IP Feature for z/VM, you should contact your IBM marketing representative and order the TCP/IP Feature as described in 2.1, “Obtaining Authorization for the TCP/IP or TCP/IP NFS Server Feature” on page 5. After you have ordered the product, you will need to complete the installation process, as described in the sections that follow.
- If you **have already ordered** the TCP/IP Feature for z/VM, most of the necessary VMSES/E installation steps have been performed. However, you need to continue with the instructions provided in 6.2, “Installing the TCP/IP Feature for z/VM — Where to Begin” on page 34 to complete the installation of the TCP/IP Feature.

The procedures that follow are in two-column format. The steps to be performed are in bold large numbers. Commands for these steps are on the left hand side of the page numbered using bold numerals. Additional information for a command may exist to the right of the command. For more information about the two-column format see “Understanding Dialogs with the System” in the *z/VM: Installation Guide*.

**Each step of the installation instructions must be followed. Do not skip any step unless directed otherwise.**

Throughout these instructions, the use of IBM-supplied default minidisk addresses and user IDs is assumed. If you use different user IDs, minidisk addresses, or SFS directories to install the TCP/IP Feature for z/VM, adapt these instructions as needed for your environment.

### Note!

The sample console output presented throughout these instructions was produced on a z/VM Version 3 Release 1.0 system and assumes that you are installing to minidisks using the default PPF and component names.

---

## 6.1 TCP/IP Feature Installation and Activation Process Overview

The following is a brief description of the steps required to install the TCP/IP Feature for z/VM and its features using VMSES/E.

- Plan Your Installation

While resources have been allocated for TCP/IP Feature as part of the z/VM Version 3 Release 1.0 System DDR that is already installed, this default environment should be reviewed and, if necessary, modified for your installation.

- Review the changes cited in 5.2.4, “Migration Considerations” on page 17, so that you are aware of changes that may affect your customization of TCP/IP Level 3A0.

- Activate (enable) the TCP/IP Feature.

- Place the TCP/IP Feature or other TCP/IP Feature files into production

The TCP/IP Feature files are copied from the test BUILD disks to the production BUILD disks.

- Configure the TCP/IP Feature

Information is provided to assist in customizing configuration files needed to run the TCP/IP Feature.

For a complete description of all VMSES/E installation options refer to:

- *z/VM: VMSES/E Introduction and Reference (GC24-5947)*

---

## 6.2 Installing the TCP/IP Feature for z/VM — Where to Begin

### Quick Index for Installation Instructions

Select the installation instructions you should use, based on the TCP/IP feature you will be working with:

- TCP/IP Feature Installation — Begin with 6.2.1, “Review the TCP/IP Feature for z/VM Default Installation Environment” on page 35
- TCP/IP NFS Server Feature Enablement — Begin with 6.3, “Enabling the TCP/IP NFS Server Feature for z/VM” on page 51
- TCP/IP Source Feature Installation — Begin with 6.4, “Installing the TCP/IP Source Feature for z/VM” on page 53

## 6.2.1 Review the TCP/IP Feature for z/VM Default Installation Environment

Because the TCP/IP Feature has been installed as part of the z/VM Version 3 Release 1.0 system DDR, several installation steps have already been performed on your behalf. Among these are the:

- addition of TCP/IP-specific user ID entries and PROFILES to the z/VM Version 3 Release 1.0 system directory
- creation of a simplified PROFILE EXEC for the 3TCPIPA0 user ID
- allocation of TCP/IP-required minidisks
- loading of TCP/IP Feature product files to minidisks, using VMSES/E commands

### 6.2.1.1 TCP/IP Feature Directory PROFILES and User IDs

For reference, the two PROFILE directory entries that have been added to the z/VM Version 3 Release 1.0 system directory, PROFILE TCPCMSU and PROFILE TCPGCSU, are shown in Figure 12. Each directory entry supplied for a TCP/IP Feature for z/VM service virtual machine includes one of these profiles.

Figure 12. TCP/IP Feature for z/VM System Directory Profiles

<pre>PROFILE TCPCMSU   IPL CMS   MACHINE XA   SPOOL 00C 2540 READER *   SPOOL 00D 2540 PUNCH A   SPOOL 00E 1403 A   CONSOLE 009 3215 T   LINK MAINT 190 190 RR   LINK MAINT 19D 19D RR   LINK MAINT 19E 19E RR * LINK MAINT 0402 0402 RR * LINK MAINT 0401 0401 RR * LINK MAINT 0405 0405 RR</pre>	<pre>PROFILE TCPGCSU   IPL GCSXA PARM AUTOLOG   MACHINE XA   NAMESAVE GCS   SPOOL 00C 2540 READER *   SPOOL 00D 2540 PUNCH A   SPOOL 00E 1403 A   CONSOLE 009 3215 T   LINK MAINT 190 190 RR   LINK MAINT 19D 19D RR   LINK MAINT 19E 19E RR * LINK MAINT 0402 0402 RR * LINK MAINT 0401 0401 RR * LINK MAINT 0405 0405 RR</pre>
--	--

**Note:** The NAMESAVE GCS statement can be removed from PROFILE TCPGCSU if the GCS saved segment is not restricted.

The user IDs that have been defined for the TCP/IP Feature for z/VM are listed in Figure 13 on page 36.

**Note:** Following installation of the z/VM Version 3 Release 1.0 System DDR, the login password for a given TCP/IP user ID is identical to the user ID itself. If you have not already done so, change these passwords to valid passwords, in accordance with your security guidelines.

### Additional User ID Notes:

1. For information about specific user ID directory entry requirements, consult the **3TCPIPA0 PLANINFO** file. This file is located on the 3TCPIPA0 191 minidisk.
2. The directory entry for the TCPIP virtual machine includes the statement: `SHARE RELATIVE 3000`  
For most installations, the relative CPU share allocation of 3000 should be suitable. However, you are free to change this value to conform to local guidelines established for defining server and guest virtual machine share settings.
3. If you create additional `RXAGENTn` machines, duplicate the `RXAGENT1` directory entry for each server you add.
4. If you create additional `NDBSRVnn` machines, duplicate the `NDBSRV01` directory entry for each server you add, and add a link for each 191 disk to the 3TCPIPA0 directory entry.

### Formatting Reminder

Any additional minidisks you create must be formatted before you continue installation of the TCP/IP Feature for z/VM.

Figure 13 (Page 1 of 2). Default User IDs - TCP/IP Feature for z/VM

TCP/IP User ID	Associated TCP/IP Function
3TCPIPA0	Maintains the TCP/IP system.
TCPMAINT	TCP/IP system ownership.
ADMSERV	Runs the Kerberos database remote administration server.
BOOTPD	Responds to client requests for boot information using data defined in a BOOTP machine file.
DHCPD	Responds to client requests for boot information using data defined in a DHCPD machine file.
FTPSERVE	Implements the File Transfer Protocol (FTP) daemon, which controls access to files on the local host.
LPSERVE	Implements the Line Printer Daemon (LPD), which handles client requests to print a file.
MPROUTE	Implements the Multiple Protocol Route (MPROUTE) server, which uses OSPF and/or RIP protocols to manage network routing information.
NAMESRV	Implements the Domain Name System (DNS) server.
NDBPMGR	Provides Network Database (NDB) Port Manager support, and is used in conjunction with the NDBSRV01 server.
NDBSRV01 (1*)	Provides Network Database (NDB) System support, in conjunction with the NDB Port Manager (NDBPMGR) server.

**Notes:**

1. Additional changes may need to be made for some user IDs, depending on your environment. For more information, see the accompanying notes on page 36.

Figure 13 (Page 2 of 2). Default User IDs - TCP/IP Feature for z/VM

TCP/IP User ID	Associated TCP/IP Function
PORTMAP	Runs the Portmapper function for RPC systems that support the Network File System protocol.
REXECD	Provides remote execution services for TCP/IP hosts that support the REXEC client.
ROUTED	Implements the Routed server, which manages IP route table entries.
RXAGENT1 (1*)	Agent virtual machine used by REXECD to process anonymous rexec client requests.
SMTP	Implements the Simple Mail Transfer Protocol (SMTP) server, which provides TCP/IP electronic mail support.
SNALNKA	Provides SNA LU 0 connections between multiple hosts.
SNMPD	Virtual machine for the SNMP Agent.
SNMPQE	Virtual machine for the SNMP Query Engine.
SSLSERV	Provides Secure Sockets Layer (SSL) protocol support for TCP/IP servers.
TCPIP (1*)	Primary virtual machine that provides TCP/IP and Telnet services.
TFTPD	Transfers files between the Byte File System (BFS) and TFTP clients.
UFTD	Implements the Unsolicited File Transfer (UFT) server.
VMKERB	Runs the Kerberos authentication server.
VMNFS	Implements the Network File System (NFS) server.
X25IPI	Provides an interface which allows the TCPIP virtual machine to communicate with hosts that use the X.25 protocol.

**Notes:**

1. Additional changes may need to be made for some user IDs, depending on your environment. For more information, see the accompanying notes on page 36.

## 6.2.2 (Optional) Moving the TCP/IP Feature to SFS Directories

By default, the TCP/IP Feature for z/VM is installed to minidisks, as part of the z/VM Version 3 Release 1.0 System DDR. However, you can move the TCP/IP Feature service minidisks to Shared File System (SFS) directories. See Appendix G, “Moving the TCP/IP Feature to SFS Directories” on page 122 for instructions on how this can be done.

## 6.2.3 Enable the TCP/IP Feature

The TCP/IP Feature for z/VM must be enabled prior to use. To enable the TCP/IP Feature, use the steps that follow.

- 1** Log on the installation user ID, **3TCPIPA0**.

The PROFILE EXEC provided for this user ID (as part of the z/VM Version 3 Release 1.0 System DDR) contains ACCESS commands for the required VMSES/E minidisks — the VMSES/E code minidisk (MAINT 5E5, by default) and the VMSES/E Software Inventory minidisk (MAINT 51D, by default).

- 2** Issue the CMS QUERY DISK command to verify the VMSES/E code and Software Inventory minidisks are correctly linked and accessed.

### query disk

Verify the MAINT 5E5 minidisk is accessed as file mode **B**, and is linked **R/O**.

Verify the MAINT 51D minidisk is accessed as file mode **D**, and is linked **R/W**.

**Note:** If another user has the MAINT 51D minidisk linked in write (R/W) mode, you'll obtain only read (R/O) access to this minidisk. If this occurs, you'll need to have that user re-link the 51D disk in read-only (RR) mode; then, you need issue the appropriate LINK and ACCESS commands for the 51D minidisk. Do not continue with these procedures until a R/W link is established to the 51D minidisk.

- 3** If necessary, establish the appropriate access to the VMSES/E minidisks.

- a** Establish read access to the VMSES/E code minidisk.

```
link MAINT 5e5 5e5 rr  
access 5e5 b
```

- b** Establish write access to the Software Inventory minidisk.

```
link maint 51d 51d mr  
access 51d d
```

#### 4 Set the TCP/IP Feature for z/VM to the enabled state.

**vmfins enable ppf 3tcpipa0 {tcpip|tcpipsfs}**

Use **tcpip** if the TCP/IP Feature default minidisk environment has been maintained; use **tcpipsfs** if the service minidisks were moved to Shared File System directories.

This command will:

- set the TCP/IP Feature as ENABLED in VMSES/E and CP
- create a 3TCIPA0 PRODSYS file on the A-disk, which contains a CP system configuration PRODUCT statement with a state of: ENABLED

#### 5 In order to **retain the ENABLED state** for the TCP/IP Feature, **update the CP system configuration file** with the information in the 3TCIPA0 PRODSYS file.

The 3TCIPA0 PRODSYS file was created by VMFINS in the previous step. You need to update the CP system configuration file so that the TCP/IP Feature will be in the ENABLED state the next time you IPL your system. If you are not authorized to update the CP system configuration file, contact your CP system administrator.

The CP system configuration file resides on the CP “parm” disk (the MAINT CF1 or CF2 minidisk); by default, this file is named: **SYSTEM CONFIG**. The necessary updates need to be made to the “PRODUCT ENABLE/DISABLE INFORMATION” section of this file. You will need to replace the current DISABLED entry for PRODID 3TCIPA0 (TCP/IP Feature for z/VM) with the entry from the 3TCIPA0 PRODSYS file (created by VMFINS in the previous step). If necessary, refer to the *z/VM: Planning and Administration* manual for more information about the CP system configuration file.

## 6.2.4 Place the TCP/IP Feature for z/VM Into Production

### Note!

If you have not determined which TCP/IP Feature services will be used within your environment, you should do so at this time, before you place TCP/IP Feature files into production using the following procedure.

To determine which services (if any) you do not want to use, review the descriptions of services provided in *z/VM: TCP/IP Level 3A0 Planning and Customization* (SC24-5981). The server-specific configuration chapters of this publication may also help you with your determination.

### 6.2.4.1 3TCPIPA0 CATALOG File Modification Notes

Before you continue with the steps in this section, it is recommended that **all** 3TCPIPA0 CATALOG file modifications that may be required for your environment be made at the same time. The 3TCPIPA0 CATALOG file is used by the TCP2PROD command to copy files to TCP/IP Feature minidisks. See Appendix A, “The TCP2PROD Command” on page 93 for information about the TCP2PROD command and TCP/IP Feature catalog files.

The various definition sections of the 3TCPIPA0 CATALOG file, and the files associated with each section, are briefly described here:

Section	Description
<b>TCPRUN</b>	Non-customizable feature run-time files. For reference purposes, files that are processed using the TCPRUN section of this file are listed in Figure 14 on page 46 of 6.2.4.7, “TCP/IP Feature for z/VM Product and Sample Configuration Files” on page 45.
<b>TCPCONFIG</b>	Customizable sample files. For reference purposes, files that are processed using the TCPCONFIG section are listed in Figure 15 on page 48.
<b>TCPSYSFILE</b>	Non-customizable feature (binary) system files. For reference purposes, files that are processed using the TCPSYSFILE section are listed in Figure 16 on page 50.

#### Notes:

1. To ensure you are notified of any service-related changes to the 3TCPIPA0 CATALOG file, make changes to this file using a VMSES/E local modification. See Appendix B, “Modifying TCP/IP Feature CATALOG Files” on page 100 for more information about how to change the 3TCPIPA0 CATALOG file in this manner.
2. When changes are made, ensure the only files that will be processed are those associated with servers you have defined for your environment.
3. The TCP2PROD command is not used to install files that are associated with TCPSYSFILE section of the 3TCPIPA0 CATALOG file; these files are already present in an appropriate state (due to inclusion of the TCP/IP Feature for z/VM as part of the z/VM Version 3 Release 1.0 System DDR).



## 6.2.4.2 Copy TCP/IP Feature Run-time Files Into Production

- 1 Log on the installation user ID, **3TCPIPA0**.

The PROFILE EXEC provided for this user ID (as part of the z/VM Version 3 Release 1.0 System DDR) contains ACCESS commands for the required VMSES/E minidisks — the VMSES/E code minidisk (MAINT 5E5, by default) and the VMSES/E Software Inventory minidisk (MAINT 51D, by default).

- 2 Issue the CMS QUERY DISK command to verify the VMSES/E code and Software Inventory minidisks are correctly linked and accessed.

**query disk**

Verify the MAINT 5E5 minidisk is accessed as file mode **B**, and is linked **R/O**.

Verify the MAINT 51D minidisk is accessed as file mode **D**, and is linked **R/W**.

**Note:** If another user has the MAINT 51D minidisk linked in write (R/W) mode, you'll obtain only read (R/O) access to this minidisk. If this occurs, you'll need to have that user re-link the 51D disk in read-only (RR) mode; then, you need issue the appropriate LINK and ACCESS commands for the 51D minidisk. Do not continue with these procedures until a R/W link is established to the 51D minidisk.

- 3 If necessary, establish the appropriate access to the VMSES/E minidisks.

- a Establish read access to the VMSES/E code minidisk.

```
link MAINT 5e5 5e5 rr
access 5e5 b
```

- b Establish write access to the Software Inventory minidisk.

```
link maint 51d 51d mr
access 51d d
```

- 4 Access the 3TCPIPA0 491 minidisk.

**access 491 i**

The 491 minidisk is where the TCP2PROD EXEC and 3TCPIPA0 CATALOG files reside.

- 5 Modify the 3TCPIPA0 CATALOG file so that the appropriate files will be placed into production for your environment, as described in 6.2.4.1, “3TCPIPA0 CATALOG File Modification Notes” on page 40.

- 6 (Optional) If necessary, establish a write link to any TCP/IP Feature production or server minidisks which are not yet linked in this mode.

LINK statements for the various TCP/IP Feature minidisks are present in the 3TCPIPA0 directory entry that was defined as part of the z/VM Version 3 Release 1.0 System DDR that is already installed.

If you have changed the default installation user ID or use different minidisk addresses in your environment, you may need to manually link the necessary TCP/IP production and server minidisks. See Figure 14 on page 46 and Figure 15 on page 48 for the link addresses you should use. If you have a PPF override that has changed any of these addresses, use your values.

```
link tcpipid vdev1 vdev2 mr
```

**Note:** If another user already has the *vdev1* minidisk linked in write (R/W) mode, you'll obtain only read (R/O) access to this minidisk. If this occurs, you'll need to have that user re-link the *vdev1* disk in read-only (RR) mode, after which you need to re-issue the above LINK command. Do not continue with these procedures until a R/W link is established to the *vdev1* minidisk.

- 7 Copy TCP/IP Feature files into production using the TCP2PROD command. The files copied via the following command are those identified in the TCPRUN section of the 3TCPIPA0 CATALOG file.

#### Verifying Your Environment

When you perform this step, you should first issue the TCP2PROD command with the **TEST** option, to verify that all resources can be accessed and that the appropriate files will be processed. See Appendix A, "The TCP2PROD Command" on page 93 for information about the TCP2PROD command and TCP/IP Feature catalog files.

With the **TEST** option in effect, **no files are copied into production.**

Resolve any reported problems, then invoke TCP2PROD without the TEST option, as illustrated below.

```
tcp2prod 3tcpipa0 {tcpip | tcpipsfs} 3tcpipa0 tcprun (setup
```

Use **tcpip** if the TCP/IP Feature default minidisk environment has been maintained; use **tcpipsfs** if the service minidisks were moved to Shared File System directories.

- 8 Review the TCP2PROD message log (TCP2PROD \$MSGLOG). If necessary, correct any problems before going on.

vmfview tcp2prod

### 6.2.4.3 Copy TCP/IP Feature Configuration Files Into Production

Use the TCP2PROD command, as described below, to place the IBM-supplied sample configured files into production.

**Note:** For step 2 below, it's assumed that the TCPCONFIG section of the 3TCPIPA0 CATALOG has been suitably modified, as discussed in 6.2.4.1, "3TCPIPA0 CATALOG File Modification Notes" on page 40. If this is not the case, you should make any necessary changes to this section of the 3TCPIPA0 CATALOG file before you continue with the steps that follow.

- 1 If necessary, establish the appropriate environment, as described by steps 1 through 4, in 6.2.4.2, "Copy TCP/IP Feature Run-time Files Into Production" on page 41.
- 2 Copy TCP/IP Feature configuration files into production using the TCP2PROD command. For reference purposes, files that can be processed using the TCPCONFIG section are listed in Figure 15 on page 48.

#### Verifying Your Environment

When you perform this step, you should first issue the TCP2PROD command with the **TEST** option, to verify that all resources can be accessed and that the appropriate files will be processed. See Appendix A, "The TCP2PROD Command" on page 93 for information about the TCP2PROD command and TCP/IP Feature catalog files.

With the **TEST** option in effect, **no files are copied into production.**

Resolve any reported problems, then invoke TCP2PROD without the TEST option, as illustrated below.

```
tcp2prod 3tcpipa0 {tcpip | tcpipsfs} 3tcpipa0 tcpconfig (setup forcetcp)
```

Use **tcpip** if the TCP/IP Feature default minidisk environment has been maintained; use **tcpipsfs** if the service minidisks were moved to Shared File System directories.

The **FORCECOPY** option causes **all** existing configuration files to be replaced, so that any sample configuration files included as part of the initial RSU can be customized.

- 3** Review the TCP2PROD message log (TCP2PROD \$MSGLOG). If necessary, correct any problems before going on.

```
vmfview tcp2prod
```

#### 6.2.4.4 Configure the TCP/IP Feature for z/VM for Your Installation

TCP/IP Feature for z/VM has been installed with *sample* copies of the configuration files used by the various TCP/IP servers and clients in your environment. Before any TCP/IP services can be used, these configuration files need to be customized for your environment.

See *z/VM: TCP/IP Level 3A0 Planning and Customization* (SC24-5981) for detailed information about the content and use of these files, and how to configure the various TCP/IP Feature servers you plan to use.

#### 6.2.4.5 Initialize TCP/IP Services

After you have completed configuring the TCP/IP Feature for your environment, you need to initialize the appropriate TCP/IP servers.

For information on starting TCP/IP servers, see “Starting and Stopping TCP/IP Services” in Chapter 3 of *z/VM: TCP/IP Level 3A0 Planning and Customization*.

#### 6.2.4.6 (Optional) Copy TCP/IP Client Code to the z/VM Product Code Disk

After the TCP/IP Feature has been configured for your environment, you may want to consider copying the TCP/IP client code to the z/VM Product Code minidisk. See Appendix H, "Copying TCP/IP Feature Client Code to the Y-Disk" on page 127 for additional information and instructions concerning this process.

#### 6.2.4.7 TCP/IP Feature for z/VM Product and Sample Configuration Files

Figure 14 on page 46 lists the name and location of TCP/IP Feature product run-time files provided by IBM, and the names and location of these files after they've been placed into production.

**Note:** The link addresses listed in this table correspond to LINK statement defaults (defined in the system (CP) directory entry for the 3TCIPA0 user ID) for the various *target* minidisks shown.

Figure 14 (Page 1 of 2). Files to Copy into Production - TCP/IP Feature for z/VM

3TCPIPA0 Source Disk	3TCPIPA0 Link Address	Source File Name/Type	Production File Name/Type	Target Minidisk
491	591	— All Files —	(no change)	———
492	592	— All Files —	(no change)	———
491	262	TCPROFIL EXEC	PROFILE EXEC	TCPIP 191
491	263	TCPROFIL EXEC	PROFILE EXEC	ADMSERV 191
491	264	TCPROFIL EXEC	PROFILE EXEC	BOOTPD 191
491	265	TCPROFIL EXEC	PROFILE EXEC	DHCPD 191
491	266	TCPROFIL EXEC	PROFILE EXEC	FTPSERVE 191
491	267	TCPROFIL EXEC	PROFILE EXEC	LPSERVE 191
491	268	TCPROFIL EXEC	PROFILE EXEC	MROUTE 191
491	269	TCPROFIL EXEC	PROFILE EXEC	NAMESRV 191
491	26A	TCPROFIL EXEC	PROFILE EXEC	NDBPMGR 191
491	26B	TCPROFIL EXEC (2*)	PROFILE EXEC	NDBSRV01 191
491	26C	TCPROFIL EXEC	PROFILE EXEC	PORTMAP 191
491	26D	TCPROFIL EXEC	PROFILE EXEC	REXECD 191
491	26E	TCPROFIL EXEC	PROFILE EXEC	ROUTED 191
491	26F	TCPROFIL EXEC	PROFILE EXEC	SMTP 191
491	270	TCPROFIL GCS (1*)	PROFILE GCS	SNALNKA 191
491	271	TCPROFIL EXEC	PROFILE EXEC	SNMPD 191
491	272	TCPROFIL EXEC	PROFILE EXEC	SNMPQE 191
491	273	TCPROFIL EXEC	PROFILE EXEC	SSLSERV 191

**Notes:**

1. The TCPROFIL GCS profiles are not interchangeable with the TCPROFIL EXEC used for other TCP/IP servers.
2. TCPROFIL EXEC should be copied to the 191 disk of any additional NDBSRV $nn$  servers that are installed.

Figure 14 (Page 2 of 2). Files to Copy into Production - TCP/IP Feature for z/VM

3TCPIPA0 Source Disk	3TCPIPA0 Link Address	Source File Name/Type	Production File Name/Type	Target Minidisk
491	279	TCPROFIL EXEC	PROFILE EXEC	TFTPD 191
491	27A	TCPROFIL EXEC	PROFILE EXEC	UFTD 191
491	27B	TCPROFIL EXEC	PROFILE EXEC	VMKERB 191
491	27C	TCPROFIL EXEC	PROFILE EXEC	VMNFS 191
491	27D	TCPROFIL GCS (1*)	PROFILE GCS	X25IPI 191

**Notes:**

1. The TCPROFIL GCS profiles are not interchangeable with the TCPROFIL EXEC used for other TCP/IP servers.
2. TCPROFIL EXEC should be copied to the 191 disk of any additional NDBSRVnn servers that are installed.

Figure 15 on page 48 lists the name and location of the TCP/IP Feature sample configuration files provided by IBM, and the default names and location for these files after they've been customized. The server or user groups for which a given file is required (or may be used) are identified as well.

**Note:** Unless otherwise noted, the **Sample** and **Configured** disks cited in this table are minidisks owned by the TCPMAINT user ID.

Figure 15 (Page 1 of 2). Files to Customize - TCP/IP Feature for z/VM

Sample Disk	Configured Disk	Sample File Name/Type	Configured File Name/Type	Usage
591	198	PROFILE STCIPIP	PROFILE TCPIP	TCPIP
592	198	HOSTS SLOCAL	HOSTS LOCAL	TCPIP
591	198	SCEXIT SEXEC	SCEXIT EXEC	TCPIP
591	198	SCEXIT SAMPASM	SCEXIT ASSEMBLE	TCPIP
591	198	PMEXIT SEXEC	PMEXIT EXEC	TCPIP
591	198	PMEXIT SAMPASM	PMEXIT ASSEMBLE	TCPIP
591	191 (*)	ADM@AADD SAMPAUTH	ADM@ACL ADD	ADMSERV
591	191 (*)	ADM@AGET SAMPAUTH	ADM@ACL GET	ADMSERV
591	191 (*)	ADM@AMOD SAMPAUTH	ADM@ACL MOD	ADMSERV
591	198	BOOTPTAB SAMPLE	ETC BOOTPTAB	BOOTPD
591	198	CHKIPADR SEXEC	CHKIPADR EXEC	FTPserve
591	198	FTPEXIT SEXEC	FTPEXIT EXEC	FTPserve
591	198	FTPEXIT SAMPASM	FTPEXIT ASSEMBLE	FTPserve
591	198	SRVRFTP SCONFIG	SRVRFTP CONFIG	FTPserve
591	198	LPD SCONFIG	LPD CONFIG	LPserve
591	198	MROUTE SCONFIG	MROUTE CONFIG	MROUTE
591	198	NSMAIN SCACHE	NSMAIN CACHE	NAMESRV
591	198	NSMAIN SDATA	NSMAIN DATA	NAMESRV
591	198	VALIDUSR SEXEC	VALIDUSR EXEC	NAMESRV
591	198	ROUTED SCONFIG	ROUTED CONFIG	ROUTED
591	198	RSCSTCP SCONFIG	RSCSTCP CONFIG	RSCS
591	198	RSCSLPD SCONFIG	RSCSLPD CONFIG	RSCS (LPD)
591	198	RSCSLPR SCONFIG	RSCSLPR CONFIG	RSCS (LPD)
591	198	RSCSLPRP SCONFIG	RSCSLPRP CONFIG	RSCS (LPD)
591	198	RSCSUFT SCONFIG	RSCSUFT CONFIG	RSCS (UFT)
591	198	SMTP SCONFIG	SMTP CONFIG	SMTP
591	198	SMTPCMDX SEXEC	SMTPCMDX EXEC	SMTP
591	198	SMTPCMDX SAMPASM	SMTPCMDX ASSEMBLE	SMTP
591	198	SMPVERX SEXEC	SMPVERX EXEC	SMTP
591	198	SMPVERX SAMPASM	SMPVERX ASSEMBLE	SMTP

**Notes:**

1. The ADMSERV 191 minidisk.
2. TCPIP DATA is used by **all** TCP/IP servers and client programs.
3. KRB CONF is used by client applications that utilize Kerberos authentication servers.



Figure 15 (Page 2 of 2). Files to Customize - TCP/IP Feature for z/VM

Sample Disk	Configured Disk	Sample File Name/Type	Configured File Name/Type	Usage
591	198	SMTPFWDX SEXEC	SMTPFWD EXEC	SMTP
591	198	SMTPFWDX SAMPASM	SMTPFWDX ASSEMBLE	SMTP
591	198	SMPMEMO SAMPLE	SECURITY MEMO	SMTP
591	198	SMPSECT SAMPTABL	SMTP SECTABLE	SMTP
591	198	SNALNKA SAMPGCS	SNALNKA GCS	SNALNKA
591	198	MIB_DESC SDATA	MIB_DESC DATA	SNMPQE
591	198	DTCSSLK PARMSAMP	DTCSSLK PARMS	SSLSERV
591	198	UFTD SCONFIG	UFTD CONFIG	UFTD
591	198	UFTCMDX SEXEC	UFTCMDX EXEC	UFTD
591	198	UFTNSLKX SEXEC	UFTNSLKX EXEC	UFTD
591	198	VMNFS SCONFIG	VMNFS CONFIG	VMNFS
591	198	VMNFSCMS SEXEC	VMNFSCMS EXEC	VMNFS
591	198	VMNFSSMG SEXEC	VMNFSSMG EXEC	VMNFS
591	198	VMNFSMON SEXEC	VMNFSMON EXEC	VMNFS
591	198	X25IPI SCONFIG	X25IPI CONFIG	X25IPI
591	198	X25IPI SAMPGCS	X25IPI GCS	X25IPI
592	592	TCPIP SDATA	TCPIP DATA (2*)	All Users
592	592	ETC SAMPSEV	ETC SERVICES	All Users
592	592	FTP SDATA	FTP DATA	FTP Users
592	592	KRB SCONFIG	KRB CONF (3*)	Clients
<b>Notes:</b>				
1. The ADMSERV 191 minidisk.				
2. TCPIP DATA is used by <b>all</b> TCP/IP servers and client programs.				
3. KRB CONF is used by client applications that utilize Kerberos authentication servers.				

Figure 16 lists the name and location of the TCP/IP Feature (binary) system files provided by IBM, and the default names and location of these files after they've been placed into production. The servers that use a given file are identified as well.

**Note:** The link addresses listed in this table correspond to LINK statement defaults (defined in the system (CP) directory entry for the 3TCPIPA0 user ID) for the various *target* minidisks shown.

*Figure 16. System Files to Build/Copy into Production - TCP/IP Feature for z/VM*

<b>3TCPIPA0 Source Disk</b>	<b>3TCPIPA0 Link Address</b>	<b>Source File Name/Type</b>	<b>Production File Name/Type</b>	<b>Target Minidisk</b>
2D2	276	DTCSSLF FLSBIN	DTCSSLF DATABIN	SSLSERV 203
2D2	275	DTCSSLS FLSBIN	DTCSSLS DATABIN	SSLSERV 202
2D2	274	DTCLFSR FLSBIN (2*)	DTCLFSR DATABIN	SSLSERV 201

**Notes:**

1. Files listed in this table are used by only the Secure Socket Layer (SSL) Server.
2. This file is not used, due to SSL-related packaging changes.

**TCP/IP Feature for z/VM is now installed and built on your system.**

---

## 6.3 Enabling the TCP/IP NFS Server Feature for z/VM

### Authorization and Ordering Information

- If you **have not ordered** the **TCP/IP NFS Server Feature for z/VM**, installation of the TCP/IP NFS Server Feature is complete; the additional installation steps provided in this section should not be used.
- If you **are planning to order** the TCP/IP NFS Server Feature for z/VM, you should contact your IBM marketing representative and order the TCP/IP NFS Server Feature as described in 2.1, “Obtaining Authorization for the TCP/IP or TCP/IP NFS Server Feature” on page 5. After you have ordered the product, you then need to complete the steps provided in this section.
- If you **have already ordered** the TCP/IP NFS Server Feature for z/VM, most of the installation steps have been completed for you. You need only to complete the steps provided in this section.

The TCP/IP NFS Server Feature for z/VM must be explicitly enabled for use. To enable the TCP/IP NFS Server Feature, use the steps that follow.

**1** Log on the installation user ID, **3TCPIPA0**.

The PROFILE EXEC provided for this user ID (as part of the z/VM Version 3 Release 1.0 System DDR) contains ACCESS commands for the required VMSES/E minidisks — the VMSES/E code minidisk (MAINT 5E5, by default) and the VMSES/E Software Inventory minidisk (MAINT 51D, by default).

**2** Issue the CMS QUERY DISK command to verify the VMSES/E code and Software Inventory minidisks are correctly linked and accessed.

**query disk**

Verify the MAINT 5E5 minidisk is accessed as file mode **B**, and is linked **R/O**.

Verify the MAINT 51D minidisk is accessed as file mode **D**, and is linked **R/W**.

**Note:** If another user has the MAINT 51D minidisk linked in write (R/W) mode, you'll obtain only read (R/O) access to this minidisk. If this occurs, you'll need to have that user re-link the 51D disk in read-only (RR) mode; then, you need issue the appropriate LINK and ACCESS commands for the 51D minidisk. Do not continue with these procedures until a R/W link is established to the 51D minidisk.

**3** If necessary, establish the appropriate access to the VMSES/E minidisks.

**a** Establish read access to the VMSES/E code minidisk.

```
link MAINT 5e5 5e5 rr
access 5e5 b
```

- b** Establish write access to the Software Inventory minidisk.

```
link maint 51d 51d mr
access 51d d
```

- 4** Set the TCP/IP NFS Server Feature for z/VM to the enabled state.

```
vmfins enable ppf 3tcnfসা0 nocomp
```

This command will:

- set the TCP/IP NFS Server Feature as ENABLED in VMSES/E and CP
- create a 3TCNFSA0 PRODSYS file on the A-disk, which contains a CP system configuration PRODUCT statement with a state of: ENABLED

- 5** In order to **retain the ENABLED state** for the TCP/IP NFS Server Feature, **update the CP system configuration file** with the information in the 3TCNFSA0 PRODSYS file.

The 3TCNFSA0 PRODSYS file was created by VMFINS in the previous step. You need to update the CP system configuration file so that the TCP/IP NFS Server Feature will be in the ENABLED state the next time you IPL your system. If you are not authorized to update the CP system configuration file, contact your CP system administrator.

The CP system configuration file resides on the CP “parm” disk (the MAINT CF1 or CF2 minidisk); by default, this file is named: **SYSTEM CONFIG**. The necessary updates need to be made to the “PRODUCT ENABLE/DISABLE INFORMATION” section of this file. You will need to replace the current DISABLED entry for PRODID 3TCNFSA0 (TCP/IP NFS Server Feature for z/VM) with the entry from the 3TCNFSA0 PRODSYS file (created by VMFINS in the previous step). If necessary, refer to the *z/VM: Planning and Administration* manual for more information about the CP system configuration file.

**The TCP/IP NFS Server Feature is now enabled on your system.**

---

## 6.4 Installing the TCP/IP Source Feature for z/VM

### 6.4.1 Plan Your Installation for the TCP/IP Source Feature for z/VM

The VMFINS command will be used to plan the installation. This is a two-step process that will:

- load the first tape file, which contains installation files
- generate a “PLANINFO” file that lists:
  - all user ID and MDISK (minidisk) requirements
  - required products

To obtain planning information for your environment:

**1** Log on the installation user ID, **3TCPIPA0**.

The PROFILE EXEC provided for this user ID (as part of the z/VM Version 3 Release 1.0 System DDR) contains ACCESS commands for the required VMSES/E minidisks — the VMSES/E code minidisk (MAINT 5E5, by default) and the VMSES/E Software Inventory minidisk (MAINT 51D, by default).

**2** Issue the CMS QUERY DISK command to verify the VMSES/E code and Software Inventory minidisks are correctly linked and accessed.

**query disk**

Verify the MAINT 5E5 minidisk is accessed as file mode **B**, and is linked **R/O**.

Verify the MAINT 51D minidisk is accessed as file mode **D**, and is linked **R/W**.

**Note:** If another user has the MAINT 51D minidisk linked in write (R/W) mode, you'll obtain only read (R/O) access to this minidisk. If this occurs, you'll need to have that user re-link the 51D disk in read-only (RR) mode; then, you need issue the appropriate LINK and ACCESS commands for the 51D minidisk. Do not continue with these procedures until a R/W link is established to the 51D minidisk.

**3** If necessary, establish the appropriate access to the VMSES/E minidisks.

**a** Establish read access to the VMSES/E code minidisk.

```
link MAINT 5e5 5e5 rr
access 5e5 b
```

**b** Establish write access to the Software Inventory minidisk.

**link maint 51d 51d mr**  
**access 51d d**

**4** Mount the TCP/IP Source Feature for z/VM installation tape and attach it to this user ID at virtual address 181. The VMFINS EXEC requires the tape drive to be at virtual address 181.

**5** Load the TCP/IP Source Feature for z/VM product control files to the 51D minidisk.

#### **vmfins install info (nomemo)**

The NOMEMO option will load the memo from the tape but will not issue a prompt to send it to the system printer. Specify the MEMO option if you want to be prompted for printing the memo.

This command will perform the following:

- load the Memo-to-Users
- load various product control files, including the Product Parameter File (PPF) and the PRODPART files
- create the VMFINS PRODLIST file on your A-disk. This file contains a list of products on the installation tape.

```
VMFINS2767I Reading VMFINS DEFAULTS B for additional options
VMFINS2760I VMFINS processing started
VMFINS1909I VMFINS PRODLIST created on your A-disk
VMFINS2760I VMFINS processing completed successfully
Ready;
```

**6** Obtain resource planning information for the TCP/IP Source Feature for z/VM.

**Note:** The product will **not** be loaded by the VMFINS command at this time.

**vmfins install ppf 3tcpipa1 {source | sourcesfs} (plan nomemo**

Use **source** if you plan to install the TCP/IP Source Feature for z/VM to minidisks; use **sourcesfs** to install to Shared File System directories.

The PLAN option indicates that VMFINS will perform requisite checking, plan system resources, and provide an opportunity to override the defaults in the product parameter file.

**You can override any of the following:**

- the name of the product parameter file
- default product user IDs
- minidisk/directory definitions

**Notes:**

- a. If you change the PPF name, a default user ID, or other parameters via a PPF override, you'll need to use your changed values instead of those indicated (when appropriate), throughout the rest of these installation instructions. For example, you'll need to specify your PPF override file name instead of 3TCPIPA1 for certain VMSES/E commands.
- b. If you're not familiar with creating PPF overrides using VMFINS, you should review the "Using the Make Override Panel" section in Chapter 3 of the *z/VM: VMSES/E Introduction and Reference* before you continue.
- c. For more information about changing the VMSYS file pool name, refer to Chapter 3 of the *z/VM: VMSES/E Introduction and Reference*.

```

VMFINS2767I Reading VMFINS DEFAULTS B for additional options
VMFINS2760I VMFINS processing started
VMFINS2601R Do you want to create an override for :PPF 3TCIPA1 SOURCE :PRODID
3TCIPA1%SOURCE?
Enter 0 (No), 1 (Yes) or 2 (Exit)
0
VMFINS2603I Processing product :PPF 3TCIPA1 SOURCE :PRODID 3TCIPA1%SOURCE
VMFREQ2805I Product :PPF 3TCIPA1 SOURCE :PRODID 3TCIPA1%SOURCE has passed
requisite checking
VMFINT2603I Planning for the installation of product :PPF 3TCIPA1 SOURCE :PRODID
3TCIPA1%SOURCE
VMFRMT2760I VMFRMT processing started
VMFRMT2760I VMFRMT processing completed successfully
VMFINS2760I VMFINS processing completed successfully

```

**Note!**

**Complete this step only if you received message VMFREQ2806W.**

If you receive the following VMFREQ2806W message (from specifying either SOURCE or SOURCESFS), you must do some additional processing.

```

VMFREQ2806W The following requisites for product 3TCIPA1 component SOURCE are
not satisfied
VMFREQ2806W Type          Product  Component      PTF
-----
VMFREQ2806W Prerequisite  3TCIPA0

```

This message indicates that you have not yet installed the TCP/IP Feature for z/VM. You must install the TCP/IP Feature for z/VM before you install the TCP/IP Source Feature for z/VM. Then, return here to complete the installation of the TCP/IP Source Feature for z/VM.

- 7** Review the install message log (\$VMFINS \$MSGLOG). If necessary, correct any problems before going on. For information about handling specific error messages, see *z/VM: System Messages and Codes* or use on-line HELP.

**vmfview install**



## 6.4.2 Allocate Resources for Installing the TCP/IP Source Feature for z/VM

Use the planning information in the 3TCPIPA1 PLANINFO file, created in the **PLAN** step, to:

- Create the 3TCPIPA0 SOURCE minidisk or SFS directory

**OR**

- Enlarge the existing 3TCPIPA0 SOURCE minidisk or SFS directory, if it isn't the appropriate size.

### 6.4.2.1 Installing the TCP/IP Source Feature for z/VM to Minidisks

**1** Obtain the disk information from the 3TCPIPA1 PLANINFO file.

**2** Update the system directory:

- Modify the 3TCPIPA0 directory entry to contain a SOURCE minidisk of the appropriate size. Use one of the two SOURCE disk sizes specified in Figure 11 on page 32 when you define this disk, depending on whether you intend to unpack source files after installation.
- Place the new directory on-line using VM/Directory Maintenance (DIRMAINT) or an equivalent CP directory maintenance method.

#### Formatting Reminder

The SOURCE minidisk must be formatted before installing the TCP/IP Source Feature for z/VM.

### 6.4.2.2 Installing the TCP/IP Source Feature for z/VM to SFS Directories

#### Note!

Perform the steps provided in this section **only** if you previously moved the TCP/IP Feature service minidisks to Shared File System directories.

**1** Obtain the disk/directory information from the 3TCPIPA1 PLANINFO file.

**2** Determine the number of 4K blocks required for the SOURCE SFS directory. Use one of the two 4K sizes specified for this directory in Figure 11 on page 32, depending on whether you intend to unpack the source files after installation.

This information will be used when the 3TCPIPA0 user ID is enrolled in the VMSYS file pool.

- 3 Enroll user 3TCPIPA0 in the VMSYS file pool, if not already done. Use the ENROLL USER command:

```
ENROLL USER 3TCPIPA0 VMSYS: (BLOCKS blocks)
```

where *blocks* is the number of 4K blocks you calculated in the previous step.

**Note:** This must be done from a user ID that is an administrator for the VMSYS file pool.

- 4 Determine if there are enough blocks available in the file pool to install the TCP/IP Source Feature for z/VM. This information can be obtained via the QUERY FILEPOOL STATUS command. Near the end of the output from this command is a list of minidisks in the file pool and the number of blocks free. If the number of blocks free is smaller than the total number needed to install the TCP/IP Source Feature, you will need to add space to the VMSYS file pool before you continue with this installation process. See *z/VM: CMS File Pool Planning, Administration, and Operation* for information about adding space to a file pool.

- 5 If 3TCPIPA0 was already enrolled in the VMSYS file pool, you might instead need to increase 3TCPIPA0's storage to hold the source files. To do this, use the MODIFY USER command:

```
MODIFY USER +blocks FOR 3TCPIPA0 VMSYS:
```

where *blocks* is the number of 4K blocks that you calculated in step 2 on page 57.

**Note:** This must be done from a user ID that is an administrator for the VMSYS file pool.

- 6 Use the CREATE DIRECTORY command to create the required subdirectory, which is listed in the 3TCPIPA1 PLANINFO file. If necessary, refer to the *z/VM: CMS Command Reference* for more information about the CREATE DIRECTORY command.

**Note:** Only create the SOURCE directory if it does not already exist.

```
set filepool vmsys:  
create directory vmsys:3tcpipa0.tcpip.source
```

### 6.4.3 Install the TCP/IP Source Feature for z/VM

- 1** Log on the installation user ID, **3TCPIPA0**.

The PROFILE EXEC provided for this user ID (as part of the z/VM Version 3 Release 1.0 System DDR) contains ACCESS commands for the required VMSES/E minidisks — the VMSES/E code minidisk (MAINT 5E5, by default) and the VMSES/E Software Inventory minidisk (MAINT 51D, by default).

- 2** Issue the CMS QUERY DISK command to verify the VMSES/E code and Software Inventory minidisks are correctly linked and accessed.

#### query disk

Verify the MAINT 5E5 minidisk is accessed as file mode **B**, and is linked **R/O**.

Verify the MAINT 51D minidisk is accessed as file mode **D**, and is linked **R/W**.

**Note:** If another user has the MAINT 51D minidisk linked in write (R/W) mode, you'll obtain only read (R/O) access to this minidisk. If this occurs, you'll need to have that user re-link the 51D disk in read-only (RR) mode; then, you need issue the appropriate LINK and ACCESS commands for the 51D minidisk. Do not continue with these procedures until a R/W link is established to the 51D minidisk.

- 3** If necessary, establish the appropriate access to the VMSES/E minidisks.

- a** Establish read access to the VMSES/E code minidisk.

```
link MAINT 5e5 5e5 rr
access 5e5 b
```

- b** Establish write access to the Software Inventory minidisk.

```
link maint 51d 51d mr
access 51d d
```

- 4** Have the TCP/IP Source Feature for z/VM installation tape mounted and attached to 3TCPIPA0 at virtual address 181. The VMFINS EXEC requires the tape drive to be at virtual address 181.

- 5** Install the TCP/IP Source Feature for z/VM.

**Notes:**

- If you've already created a PPF override file, you should specify your override file name after the **PPF** keyword for the following VMFINS command.
- You may be prompted for additional information during VMFINS INSTALL processing depending on your installation environment. If you're unsure how to respond to a prompt, refer to the “Installing Products with VMFINS” and “Install Scenarios” chapters in the *z/VM: VMSES/E Introduction and Reference* to decide how to proceed.

**vmfins install ppf 3tcpipa1 {source | sourcesfs} (nomemo nolink**

Use **source** to install the TCP/IP Source Feature for z/VM to minidisks; use **sourcesfs** to install to Shared File System directories.

**nolink** prevents VMFINS from linking the appropriate minidisks; it will only access them if they are not accessed.

```

VMFINS2767I Reading VMFINS DEFAULTS B for additional options
VMFINS2760I VMFINS processing started
VMFINS2601R Do you want to create an override for :PPF 3TCPIPA1 SOURCE :PRODID
3TCPIPA1%SOURCE?
Enter 0 (No), 1 (Yes) or 2 (Exit)
0
VMFINS2603I Processing product :PPF 3TCPIPA1 SOURCE :PRODID 3TCPIPA1%SOURCE
VMFREQ2805I Product :PPF 3TCPIPA1 SOURCE :PRODID 3TCPIPA1%SOURCE has passed
requisite checking
VMFINT2603I Installing product :PPF 3TCPIPA1 SOURCE :PRODID 3TCPIPA1%SOURCE
VMFSET2760I VMFSETUP processing started
VMFUTL2205I Minidisk|Directory Assignments:
String Mode Stat Vdev Label/Directory
VMFUTL2205I BASE2 E R/W 2B3 FAL2B3
VMFUTL2205I ----- A R/W 191 FAL191
VMFUTL2205I ----- B R/O 5E5 MNT5E5
VMFUTL2205I ----- D R/W 51D MNT51D
VMFUTL2205I ----- S R/O 190 MNT190
VMFUTL2205I ----- Y/S R/O 19E MNT19E
VMFSET2760I VMFSETUP processing completed successfully
VMFREC2760I VMFREC processing started
VMFREC1852I Volume 1 of 1 of INS TAPE 0110
VMFREC1851I (1 of 1) VMFRCALL processing SOURCE
VMFRCA2159I Loading part(s) to BASE2 2B3 (E)
VMFRCA2159I Loaded nnn part(s) to BASE2 2B3 (E)
VMFREC2760I VMFREC processing completed successfully
VMFINT2603I Product installed
VMFINS2760I VMFINS processing completed successfully

```

- 6** Review the install message log (\$VMFINS \$MSGLOG). If necessary, correct any problems before going on. For information about handling specific error messages, see *z/VM: System Messages and Codes* or use on-line HELP.

## vmfview install

## 6.4.4 Place the TCP/IP Source Feature for z/VM Into Production

The TCP/IP Source Feature for z/VM does not contain any executable files, so there is nothing to put into production or configure. The source files remain on the SOURCE disk.

### 6.4.4.1 Unpack the TCP/IP Source Feature for z/VM Source Files

The TCP/IP Source Feature for z/VM has been installed with packed source files. If you wish to unpack the source files, log on 3TCPIPA0 and invoke the CMS COPYFILE command with the UNPACK option.

**Note:** The SOURCE minidisk (or for SFS, the number of 4K blocks allocated for the 3TCPIPA0 user ID) must be sufficiently large to store the unpacked files.

**access 2b3 e**  
**copyfile *fn ft* e (unpack olddate**

Access the SOURCE directory if you installed to the Shared File System.

If **all** files on the disk are packed, you can unpack them all by specifying \* \* for *fn ft*; otherwise, you should unpack TCP/IP Source Feature for z/VM files on an individual basis.

**The TCP/IP Source Feature for z/VM is now installed and built on your system.**

---

## 7.0 Service Instructions

This section contains the procedures to install corrective (COR) and preventive service to the TCP/IP Feature feature, using VMSES/E. Service for the TCP/IP Source and TCP/IP NFS Server features is included in the service for the TCP/IP Feature for z/VM. Preventive service (for only the TCP/IP Feature for z/VM) is delivered via a Recommended Service Upgrade (RSU) tape.

To become more familiar with service using VMSES/E, you should read the introductory chapters in:

- *z/VM: VMSES/E Introduction and Reference (GC24-5947)*

These manuals also contain the command syntax for the VMSES/E commands listed in the procedure.

**Each step of the servicing instructions must be followed. Do not skip any step unless otherwise directed.**

Throughout these instructions, the use of IBM-supplied default minidisk addresses and user IDs is assumed. If you used different user IDs, minidisk addresses, or SFS directories to install the TCP/IP Feature for z/VM, adapt these instructions as needed for your environment.

**Note!**

The sample console output presented throughout these instructions was produced on a z/VM Version 3 Release 1.0 system and assumes that you installed to minidisks using the default PPF and component names.

---

### 7.1 VMSES/E Service Process Overview

The following is a brief description of the main steps in servicing TCP/IP Feature for z/VM and the features using VMSES/E.

- Merging Service

Use the VMFMRDSK command to clear the alternate apply disk before receiving new service. This allows you to easily remove the new service if a serious problem is found.

- Receiving Service

The VMFREC command receives service from the delivery media and places it on the DELTA disk.

- Applying Service

The VMFAPPLY command updates the version vector table (VVT), which identifies the service level of all the serviced parts. In addition, AUX files are generated from the VVT for parts that require them.

- Reapplying Local Service (if applicable)

All local service (mods) must be entered into the software inventory to allow VMSES/E to track the changes and build them into the system. Refer to Chapter 7 of the *z/VM: Service Guide* for this procedure.

- **Building New Levels**

The build task generates the serviced level of an object and places the new object on a test BUILD disk.

- **Placing the New Service into Production**

Once the service is satisfactorily tested it should be put into production by copying the new service to the production disks. Customized files that have been serviced must be customized again.



---

## 7.2 Servicing the TCP/IP Feature for z/VM

### 7.2.1 Important Service Notes

- Service for the TCP/IP Source and TCP/IP NFS Server features is included with service for the TCP/IP Feature for z/VM.

#### 7.2.1.1 PPF Override Considerations

- The *ppfname* used throughout these servicing instructions is **3TCPIPA0**, which assumes you are using the PPF supplied by IBM for TCP/IP Feature for z/VM. If you have your own PPF override file for the TCP/IP Feature for z/VM, you should use your file's *ppfname* instead of **3TCPIPA0**. The *ppfname* you use should be used **throughout** the rest of this procedure.

#### 7.2.1.2 Language Environment Run-time Library Considerations

- If you service TCP/IP Feature C components, the Language Environment run-time library must be available when you build serviced objects using the VMFBLD command. If the Language Environment run-time library does not reside on a system minidisk automatically accessed by VMSES/E (such as the MAINT 19E minidisk), you need to ensure the appropriate minidisk is available (perhaps through the use of a PPF override).

#### 7.2.1.3 Installing RSU and COR Service — Where to Begin

##### Quick Index for Service Instructions

Select the service instructions you should use, based on the type of service you are installing:

- TCP/IP Feature RSU Service — Begin with 7.2.2, “Preventive (RSU) Service for the TCP/IP Feature for z/VM”
- TCP/IP Feature COR Service — Begin with 7.2.3, “Corrective (COR) service for the TCP/IP Feature for z/VM” on page 74

### 7.2.2 Preventive (RSU) Service for the TCP/IP Feature for z/VM

Preventive service is available periodically on the Recommended Service Upgrade (RSU) tape. Each RSU is cumulative and contains selected important PTFs. The service on the RSU is in pre-applied, pre-built format, and includes the serviced files, the objects that were rebuilt using these files, and the updated software inventory. This makes installing the new service much quicker, but will require that you re-apply any reach-ahead service (service that you have applied that is not on the RSU). RSU tapes are in install format, so the VMFINS command is used to load the tape.

## 7.2.2.1 Prepare to Receive Service

### Electronic Service (Envelope File)

If you have received the RSU electronically or on CD-ROM, follow the appropriate instructions to retrieve and decompact the envelope file to your A-disk. Decompression is currently done by using the DETERSE module, which will produce a decompacted file that has a file name format of RPTF*num* and a file type of SERVLINK. You will need to supply this file name with the VMFINS commands that follow.

**1** Log on the TCP/IP Feature service user ID, **3TCPIPA0**.

The PROFILE EXEC provided for this user ID (as part of the z/VM Version 3 Release 1.0 System DDR) contains ACCESS commands for the required VMSES/E minidisks — the VMSES/E code minidisk (MAINT 5E5, by default) and the VMSES/E Software Inventory minidisk (MAINT 51D, by default).

**2** Issue the CMS QUERY DISK command to verify the VMSES/E code and Software Inventory minidisks are correctly linked and accessed.

**query disk**

Verify the MAINT 5E5 minidisk is accessed as file mode **B**, and is linked **R/O**.

Verify the MAINT 51D minidisk is accessed as file mode **D**, and is linked **R/W**.

**Note:** If another user has the MAINT 51D minidisk linked in write (R/W) mode, you'll obtain only read (R/O) access to this minidisk. If this occurs, you'll need to have that user re-link the 51D disk in read-only (RR) mode; then, you need issue the appropriate LINK and ACCESS commands for the 51D minidisk. Do not continue with these procedures until a R/W link is established to the 51D minidisk.

**3** If necessary, establish the appropriate access to the VMSES/E minidisks.

**a** Establish read access to the VMSES/E code minidisk.

```
link MAINT 5e5 5e5 rr
access 5e5 b
```

**b** Establish write access to the Software Inventory minidisk.

```
link maint 51d 51d mr
access 51d d
```

**4** Have the TCP/IP Feature for z/VM RSU tape mounted and attached as 181 to **3TCPIPA0**.

- 5 Receive the product documentation (3TCPIPA0 MEMO) to the 51D minidisk and identify the products and components that have service on the tape. The product documentation contains the amount of storage needed to receive the service on the RSU. Use this to ensure there is enough space on the service disks or directories.

**a** If receiving the RSU from **tape**, issue:

**vmfins install info (nomemo**

**nomemo** will load but not print the memo.

**b** If receiving the RSU from an **envelope** file, issue:

**vmfins install info (nomemo env rptfnum**

**nomemo** will load but not print the memo.

- 6 Clear the alternate APPLY disk to ensure that you have a clean disk for new service.

**vmfmrdsd 3tcpipa0 {tcpip | tcpipsfs} apply (setup**

Use **tcpip** if the TCP/IP Feature default minidisk environment has been maintained; use **tcpipsfs** if the service minidisks were moved to Shared File System directories.

This command copies the alternate APPLY disk to the production APPLY disk and then clears the alternate APPLY disk.

- 7 Review the merge message log (\$VMFMRD \$MSGLOG). If necessary, correct any problems before going on. For information about handling specific error messages, see *z/VM: System Messages and Codes* or use on-line HELP.

**vmfview mrd**

- 8 Invoke the VMFPSU command to obtain additional information about the service contained on the RSU and how it will affect your local modifications. This command creates an output file, *appid PSUPLAN*, which you can review. See the *z/VM: Service Guide* for an explanation of this file.

**vmfpsu 3tcpipa0 {tcpip | tcpipsfs}**

Use **tcpip** if the TCP/IP Feature default minidisk environment has been maintained; use **tcpipsfs** if the service minidisks were moved to Shared File System directories.

This command produces an output file comparing the service on the RSU to the service on your system. The file name is *appid PSUPLAN*, where *appid* is specified in the PPF file.

### 7.2.2.2 Receive the Service

#### **1** Receive the service on the RSU.

Since the RSU contains pre-applied, pre-built service in install format, the VMFINS command will load the new service to the DELTA disk, the updated apply service inventory to the APPLY disk, and the pre-built objects to the appropriate test build disks.

**a** If receiving the RSU from **tape**, issue:

**vmfins install ppf 3tcpipa0 {tcpip | tcpipsfs} (nomemo nolink**

Use **tcpip** if the TCP/IP Feature default minidisk environment has been maintained; use **tcpipsfs** if the service minidisks were moved to Shared File System directories.

**nolink** prevents VMFINS from linking the appropriate minidisks; it will only access them if not accessed.

**b** If receiving the RSU from an **envelope** file, issue:

**vmfins install ppf 3tcpipa0 {tcpip | tcpipsfs} (nomemo nolink env rptfnum override no**

Use **tcpip** if the TCP/IP Feature default minidisk environment has been maintained; use **tcpipsfs** if the service minidisks were moved to Shared File System directories.

**nolink** prevents VMFINS from linking the appropriate minidisks; it will only access them if not accessed.

- 2 Review the install message log (\$VMFINS \$MSGLOG). If necessary, correct any problems before going on. For information about handling specific error messages, see *z/VM: System Messages and Codes* or use on-line HELP.

## vmfview install

### 7.2.2.3 Apply the Service

Since the service on the RSU is already pre-applied, this step will just reapply any reach-ahead service (service that is on your system but not on the RSU).

- 1 Reapply reach-ahead service.

#### vmfapply ppf 3tcpipa0 {tcpip | tcpipsfs}

Use **tcpip** if the TCP/IP Feature default minidisk environment has been maintained; use **tcpipsfs** if the service minidisks were moved to Shared File System directories.

This command reapplies the reach-ahead service on your system. The version vector table (VVT) is updated with all serviced parts and all necessary AUX files are generated on the alternate apply disk.

- 2 Review the apply message log (\$VMFAPP \$MSGLOG). If necessary, correct any problems before going on. For information about handling specific error messages, see *z/VM: System Messages and Codes* or use on-line HELP.

## vmfview apply

- 3 Re-work and re-apply local service, if it has been affected by the service on the RSU.
  - a. Affected local modifications are indicated in the output file created by the VMFPSU command invoked in a step 8 on page 67.
  - b. For information on re-working local modifications, refer to Chapter 7 in the *z/VM: Service Guide*. Follow the steps that are applicable to your local modification.

The following substitutions may need to be made:

- **zvm** should be: **3tcpipa0**
- *compname* should be: **tcpip** or **tcpipsfs**
- *appid* should be: **3tcpipa0**
- *fm-local* should be the file mode of the 2C4 minidisk
- *fm-applyalt* should be the file mode of the 2A6 minidisk

Keep in mind that when you get to the following step in the *z/VM: Service Guide*:

- “Rebuilding Objects”

you should return to using this program directory and continue with 7.2.2.4, “Update the Build Status Table.”

#### 7.2.2.4 Update the Build Status Table

**1** Update the Build Status Table with serviced parts.

**vmfbld ppf 3tcpipa0 {tcpip | tcpipsfs} (status**

Use **tcpip** if the TCP/IP Feature default minidisk environment has been maintained; use **tcpipsfs** if the service minidisks were moved to Shared File System directories.

This command updates the Build Status Table to determine what remains to be built.

**Note - \$PPF Service**

If a \$PPF file has been serviced you will get the following prompt:

```
VMFBLD2185R The following source product parameter files have been
              serviced:
VMFBLD2185R 3TCPIPA0 $PPF
VMFBLD2185R When source product parameter files are serviced, all
              product parameter files built from them must be recompiled
              using VMFPPF before VMFBLD can be run.
VMFBLD2185R Enter zero (0) to have the serviced source product
              parameter files built to your A-disk and exit VMFBLD so
              you can recompile your product parameter files with VMFPPF.
VMFBLD2185R Enter one (1) to continue only if you have already
              recompiled your product parameter files with VMFPPF.
```

**0**

Enter a 0 and complete the following steps before you continue.

```
VMFBLD2188I Building 3TCPIPA0 $PPF
              on 191 (A) from level $PFnnnnn
```

**vmfppf 3tcpipa0 \***

**Note:** If you've created your own PPF override, use your PPF name instead of 3TCPIPA0.

**copyfile 3tcpipa0 \$ppf a = d (olddate replace**

**erase 3tcpipa0 \$ppf a**

**Note:** **Do not** use your own PPF name in place of 3TCPIPA0 for these COPYFILE and ERASE commands.

**vmfbld ppf 3tcpipa0 {tcpip | tcpipsfs} (status**

**1**

Re-issue VMFBLD to complete updating the build status table.

Use **tcpip** if the TCP/IP Feature default minidisk environment has been maintained; use **tcpipsfs** if the service minidisks were moved to Shared File System directories. When you receive the prompt that was previously displayed, enter a 1 to continue.

- 2 Use VMFVIEW to review the build status messages, and see what objects need to be built.

**vmfview build**

### 7.2.2.5 Build Serviced Objects

The RSU contains pre-built service, so this step will build only objects that were affected by any reach-ahead and local service that was reapplied.

- 1 Ensure Language Environment run-time library support is available if you are applying service to functions that require this support.
- 2 Rebuild TCP/IP Feature for z/VM serviced parts.

**vmfbld ppf 3tcpipa0 {tcpip | tcpipsfs} (serviced**

Use **tcpip** if the TCP/IP Feature default minidisk environment has been maintained; use **tcpipsfs** if the service minidisks were moved to Shared File System directories.

- 3 Review the build message log (\$VMFBLD \$MSGLOG). If necessary, correct any problems before going on. For information about handling specific error messages, see *z/VM: System Messages and Codes* or use on-line HELP.

**vmfview build**

**Note:** The message that follows will be included in the build message log for each TCP/IP Feature (binary) system file that has been serviced. This is a normal occurrence, since system files are built by the TCP2PROD command when they are placed into production.

```
ST:VMFBLD1851I Cannot process TCPBLxxx EXCnnnnn because it is flagged to
ST:                be bypassed
```



## 7.2.2.6 Test the New Service

You should thoroughly test new service before you put it into production. To do this, add the links shown below for each server machine you're using, so that the server and client code test build minidisks are linked instead of their production counterparts. In so doing, you need to first detach the 591 and 592 production minidisks. The required LINK commands are:

```
LINK 3TCPIPA0 491 591 RR
LINK 3TCPIPA0 492 592 RR
```

When you test new service, you should consult any applicable documentation (for example, that provided with the APARs that comprise the service) to account for changes specific to new or changed function. Also, your testing may require TCP/IP services affected by service to be shutdown and restarted, possibly more than once.

**Note:** Before you shutdown any TCP/IP servers, ensure any applicable conditions or guidelines for your environment have been followed.

For information on shutting down TCP/IP servers, see “Starting and Stopping TCP/IP Services” in Chapter 3 of *z/VM: TCP/IP Level 3A0 Planning and Customization*.

## 7.2.2.7 Place the Service into Production

When you have thoroughly tested the new service, you should put the service into production and complete the remainder of the TCP/IP service procedure.

### RSU Service Instructions — Where To Next...

See 7.2.4, “Place the New TCP/IP Feature for z/VM Service Into Production” on page 81 for instructions on completing the installation of RSU service.

## 7.2.3 Corrective (COR) service for the TCP/IP Feature for z/VM

Corrective service for the TCP/IP Feature for z/VM is provided in COR format via tape or electronic envelope. It is installed using VMFREC, VMFAPPLY, and VMFBLD.

### Electronic Service (Envelope File)

If you have received service electronically or on CD-ROM, follow the appropriate instructions to retrieve and decompact the envelope files to your A-disk. Decompaction is currently done by using the DETERSE module, which will produce decompacted files with the following file name formats:

- VLST*num*, for the documentation envelope
- VPTF*num*, for the service envelope.

The file type for both of these files is (and must be) SERVLINK. You will need to supply these file names with the VMFREC commands that follow.

### TCP/IP Feature System File Considerations

If the corrective service you intend to apply affects TCP/IP Feature (binary) system files, you may first need to *commit* the current level of service for these files **before** the relevant new service can be received and applied. For more information, see Appendix C, “PTF Commit Procedure for TCP/IP Feature System Files” on page 104.

### 7.2.3.1 Prepare to Receive Service

- 1** Log on the TCP/IP Feature service user ID, **3TCPIPA0**.

The PROFILE EXEC provided for this user ID (as part of the z/VM Version 3 Release 1.0 System DDR) contains ACCESS commands for the required VMSES/E minidisks — the VMSES/E code minidisk (MAINT 5E5, by default) and the VMSES/E Software Inventory minidisk (MAINT 51D, by default).

- 2** Issue the CMS QUERY DISK command to verify the VMSES/E code and Software Inventory minidisks are correctly linked and accessed.

#### query disk

Verify the MAINT 5E5 minidisk is accessed as file mode **B**, and is linked **R/O**.

Verify the MAINT 51D minidisk is accessed as file mode **D**, and is linked **R/W**.

**Note:** If another user has the MAINT 51D minidisk linked in write (R/W) mode, you'll obtain only read (R/O) access to this minidisk. If this occurs, you'll need to have that user re-link the 51D disk in read-only (RR) mode;

then, you need issue the appropriate LINK and ACCESS commands for the 51D minidisk. Do not continue with these procedures until a R/W link is established to the 51D minidisk.

**3** If necessary, establish the appropriate access to the VMSES/E minidisks.

**a** Establish read access to the VMSES/E code minidisk.

**link MAINT 5e5 5e5 rr**  
**access 5e5 b**

**b** Establish write access to the Software Inventory minidisk.

**link maint 51d 51d mr**  
**access 51d d**

**4** Have the TCP/IP Feature for z/VM corrective (COR) service tape mounted and attached as 181 to **3TCPIPA0**.

**5** Establish the correct minidisk access order.

**vmfsetup 3tcpipa0 {tcpip | tcpipsfs}**

Use **tcpip** if the TCP/IP Feature default minidisk environment has been maintained; use **tcpipsfs** if the service minidisks were moved to Shared File System directories.

**6** Receive the documentation. VMFREC, with the INFO option, loads the documentation and displays a list of all the products on the tape.

**a** If receiving service from **tape**, issue:

**vmfrec info**

This command will load the service memo to the 191 disk.

**b** If receiving service from an **envelope** file, issue:

**vmfrec info (env vlstnum**

This command will load the service memo to the 191 disk.

**7** Review the receive message log (\$VMFREC \$MSGLOG). If necessary, correct any problems before going on. For information about handling specific error messages, see *z/VM: System Messages and Codes* or use on-line HELP.

## vmfview receive

Also make note of which products and components have service on the tape. To do this, use the PF5 key to show all status messages which identify the products on the tape.

- 8 Clear the alternate APPLY disk to ensure that you have a clean disk for new service.

## vmfmrdsk 3tcpipa0 {tcpip | tcpipsfs} apply

Use **tcpip** if the TCP/IP Feature default minidisk environment has been maintained; use **tcpipsfs** if the service minidisks were moved to Shared File System directories.

This command copies the alternate APPLY disk to the production APPLY disk and then clears the alternate APPLY disk.

- 9 Review the merge message log (\$VMFMRD \$MSGLOG). If necessary, correct any problems before going on. For information about handling specific error messages, see *z/VM: System Messages and Codes* or use on-line HELP.

## vmfview mrd

### 7.2.3.2 Receive the Service

- 1 Receive the service.

**a** If receiving service from **tape**, issue:

## vmfrec ppf 3tcpipa0 {tcpip | tcpipsfs}

Use **tcpip** if the TCP/IP Feature default minidisk environment has been maintained; use **tcpipsfs** if the service minidisks were moved to Shared File System directories.

This command receives service from your service tape. All new service is loaded to the DELTA disk.

**b** If receiving service from an **envelope** file, issue:

## vmfrec ppf 3tcpipa0 {tcpip | tcpipsfs} (env vptfnum

Use **tcpip** if the TCP/IP Feature default minidisk environment has been maintained; use **tcpipsfs** if the service minidisks were moved to Shared File System directories.

This command receives service from your service envelope. All new service is loaded to the DELTA disk.

- 2 Review the receive message log (\$VMFREC \$MSGLOG). If necessary, correct any problems before going on. For information about handling specific error messages, see *z/VM: System Messages and Codes* or use on-line HELP.

**vmfview receive**

### 7.2.3.3 Apply the Service

- 1 Apply the new service.

**vmfapply ppf 3tcpipa0 {tcpip | tcpipsfs}**

Use **tcpip** if the TCP/IP Feature default minidisk environment has been maintained; use **tcpipsfs** if the service minidisks were moved to Shared File System directories.

This command applies the service that you just received. The version vector table (VVT) is updated with all serviced parts and all necessary AUX files are generated on the alternate apply disk.

- 2 Review the apply message log (\$VMFAPP \$MSGLOG). If necessary, correct any problems before going on. For information about handling specific error messages, see *z/VM: System Messages and Codes* or use on-line HELP.

**vmfview apply**

#### Note - Local Modifications

If you receive message VMFAPP2120W, you need to re-apply any local modifications before building the new TCP/IP Feature for z/VM. Refer to chapter 7 in the *z/VM: Service Guide*. Follow the steps that are applicable to your local modification.

The following substitutions may need to be made:

- **zvm** should be: **3tcpipa0**
- *compname* should be: **tcpip** or **tcpipsfs**
- *appid* should be: **3tcpipa0**
- *fm-local* should be the file mode of the 2C4 minidisk
- *fm-applyalt* should be the file mode of the 2A6 minidisk

Keep in mind that when you get to the following step in the *z/VM: Service Guide*:

- “Rebuilding Objects”

you should return to using this program directory and continue with 7.2.3.4, “Update the Build Status Table” on page 78.

### 7.2.3.4 Update the Build Status Table

- 1 Update the Build Status Table with serviced parts.

**vmfbld ppf 3tcpipa0 {tcpip | tcpipsfs} (status**

Use **tcpip** if the TCP/IP Feature default minidisk environment has been maintained; use **tcpipsfs** if the service minidisks were moved to Shared File System directories.

This command updates the Build Status Table to include objects that have to be built due to the new service just applied.

**Note - \$PPF Service**

If a \$PPF file has been serviced you will get the following prompt:

```
VMFBLD2185R The following source product parameter files have been
              serviced:
VMFBLD2185R 3TCPIPA0 $PPF
VMFBLD2185R When source product parameter files are serviced, all
              product parameter files built from them must be recompiled
              using VMFPPF before VMFBLD can be run.
VMFBLD2185R Enter zero (0) to have the serviced source product
              parameter files built to your A-disk and exit VMFBLD so
              you can recompile your product parameter files with VMFPPF.
VMFBLD2185R Enter one (1) to continue only if you have already
              recompiled your product parameter files with VMFPPF.
```

**0**

Enter a 0 and complete the following steps before you continue.

```
VMFBLD2188I Building 3TCPIPA0 $PPF
              on 191 (A) from level $PFnnnnn
```

**vmfppf 3tcpipa0 \***

**Note:** If you've created your own PPF override, use your PPF name instead of 3TCPIPA0.

**copyfile 3tcpipa0 \$ppf a = d (olddate replace**

**erase 3tcpipa0 \$ppf a**

**Note:** **Do not** use your own PPF name in place of 3TCPIPA0 for these COPYFILE and ERASE commands.

**vmfbld ppf 3tcpipa0 {tcpip | tcpipsfs} (status setup**

**1**

Re-issue VMFBLD to complete updating the build status table.

Use **tcpip** if the TCP/IP Feature default minidisk environment has been maintained; use **tcpipsfs** if the service minidisks were moved to Shared File System directories. When you receive the prompt that was previously displayed, enter a 1 to continue.

- 2 Use VMFVIEW to review the build status messages, and see what objects need to be built.

**vmfview build**

### 7.2.3.5 Build Serviced Objects

- 1 Ensure Language Environment run-time library support is available if you are applying service to functions that require this support.
- 2 Rebuild TCP/IP Feature for z/VM serviced parts.

**vmfbld ppf 3tcpipa0 {tcpip | tcpipsfs} (serviced**

Use **tcpip** if the TCP/IP Feature default minidisk environment has been maintained; use **tcpipsfs** if the service minidisks were moved to Shared File System directories.

- 3 Review the build message log (\$VMFBLD \$MSGLOG). If necessary, correct any problems before going on. For information about handling specific error messages, see *z/VM: System Messages and Codes* or use on-line HELP.

**vmfview build**

**Note:** The message that follows will be included in the build message log for each TCP/IP Feature (binary) system file that has been serviced. This is a normal occurrence, since system files are built by the TCP2PROD command when they are placed into production.

```
ST:VMFBLD1851I Cannot process TCPBLxxx EXCnnnnn because it is flagged to  
ST:                be bypassed
```

### 7.2.3.6 Test the New Service

You should thoroughly test new service before you put it into production. To do this, add the links shown below for each server machine you're using, so that the server and client code test build minidisks are linked instead of their production counterparts. In so doing, you need to first detach the 591 and 592 production minidisks. The required LINK commands are:



LINK 3TCPIPA0 491 591 RR  
LINK 3TCPIPA0 492 592 RR

When you test new service, you should consult any applicable documentation (for example, that provided with the APARs that comprise the service) to account for changes specific to new or changed function. Also, your testing may require TCP/IP services affected by service to be shutdown and restarted, possibly more than once.

**Note:** Before you shutdown any TCP/IP servers, ensure any applicable conditions or guidelines for your environment have been followed.

For information on shutting down TCP/IP servers, see “Starting and Stopping TCP/IP Services” in Chapter 3 of *z/VM: TCP/IP Level 3A0 Planning and Customization*.

## 7.2.4 Place the New TCP/IP Feature for z/VM Service Into Production

### 7.2.4.1 3TCPIPA0 CATALOG File Modification Notes

Before you continue with the steps in this section, it is recommended that you verify the correctness of any 3TCPIPA0 CATALOG file modifications that may have been made for your environment. The 3TCPIPA0 CATALOG file is used by the TCP2PROD command to copy files to TCP/IP Feature minidisks. See Appendix A, “The TCP2PROD Command” on page 93 for information about the TCP2PROD command and TCP/IP Feature catalog files.

Any new files that can be processed will be identified within a serviced 3TCPIPA0 CATALOG file.

The various definition sections of the 3TCPIPA0 CATALOG file, and the files associated with each section, are briefly described here:

Section	Description
<b>TCPRUN</b>	Non-customizable feature run-time files. For reference purposes, files that can be processed using the TCPRUN section of this file are listed in Figure 14 on page 46 of 6.2.4.7, “TCP/IP Feature for z/VM Product and Sample Configuration Files” on page 45.
<b>TCPCONFIG</b>	Customizable sample files. For reference purposes, files that can be processed using the TCPCONFIG section are listed in Figure 15 on page 48.
<b>TCPSYSFILE</b>	Non-customizable feature (binary) system files. For reference purposes, files that can be processed using the TCPSYSFILE section are listed in Figure 16 on page 50.

## 7.2.4.2 Copy Serviced TCP/IP Feature Run-time Files Into Production

When the new service has been thoroughly tested, it should be copied to the production build disks to place it into production.

### 1 Shutdown TCP/IP services.

#### Note - Shutting Down TCP/IP Services

Before you shutdown any TCP/IP servers, ensure any applicable conditions or guidelines for your environment have been followed.

For information on shutting down TCP/IP servers, see “Starting and Stopping TCP/IP Services” in Chapter 3 of *z/VM: TCP/IP Level 3A0 Planning and Customization*.

### 2 Log on the TCP/IP Feature service user ID, **3TCPIPA0**.

The PROFILE EXEC provided for this user ID (as part of the z/VM Version 3 Release 1.0 System DDR) contains ACCESS commands for the required VMSES/E minidisks — the VMSES/E code minidisk (MAINT 5E5, by default) and the VMSES/E Software Inventory minidisk (MAINT 51D, by default).

### 3 Issue the CMS QUERY DISK command to verify the VMSES/E code and Software Inventory minidisks are correctly linked and accessed.

#### query disk

Verify the MAINT 5E5 minidisk is accessed as file mode **B**, and is linked **R/O**.

Verify the MAINT 51D minidisk is accessed as file mode **D**, and is linked **R/W**.

**Note:** If another user has the MAINT 51D minidisk linked in write (R/W) mode, you'll obtain only read (R/O) access to this minidisk. If this occurs, you'll need to have that user re-link the 51D disk in read-only (RR) mode; then, you need issue the appropriate LINK and ACCESS commands for the 51D minidisk. Do not continue with these procedures until a R/W link is established to the 51D minidisk.

**4** If necessary, establish the appropriate access to the VMSES/E minidisks.

**a** Establish read access to the VMSES/E code minidisk.

```
link MAINT 5e5 5e5 rr
access 5e5 b
```

**b** Establish write access to the Software Inventory minidisk.

```
link maint 51d 51d mr
access 51d d
```

**5** Access the 3TCPIPA0 491 minidisk.

```
access 491 i
```

The 491 minidisk is where the TCP2PROD EXEC and 3TCPIPA0 CATALOG files reside.

**6** Review the 3TCPIPA0 CATALOG file to verify its correctness, as recommended in 7.2.4.1, “3TCPIPA0 CATALOG File Modification Notes” on page 81. Ensure that any changes you may have made to this file remain in effect, and that any local service used to customize this file has been properly re-worked and applied.

**7** (Optional) If necessary, establish a write link to any TCP/IP Feature production or server minidisks which are not yet linked in this mode.

LINK statements for the various TCP/IP Feature minidisks are present in the 3TCPIPA0 directory entry that was defined as part of the z/VM Version 3 Release 1.0 System DDR that is already installed.

If you have changed the default installation user ID or use different minidisk addresses in your environment, you may need to manually link the necessary TCP/IP production and server minidisks. See Figure 14 on page 46 and Figure 15 on page 48 for the link addresses you should use. If you have a PPF override that has changed any of these addresses, use your values.

```
link tcpipid vdev1 vdev2 mr
```

**Note:** If another user already has the *vdev1* minidisk linked in write (R/W) mode, you'll obtain only read (R/O) access to this minidisk. If this occurs, you'll need to have that user re-link the *vdev1* disk in read-only (RR) mode, after which you need to re-issue the above LINK command. Do not continue with these procedures until a R/W link is established to the *vdev1* minidisk.

- 8 Copy serviced TCP/IP Feature files into production using the TCP2PROD command. The files copied via the following command are those identified in the TCPRUN section of the 3TCPIPA0 CATALOG file.

#### Verifying Your Environment

When you perform this step, you should first issue the TCP2PROD command with the **TEST** option, to verify that all resources can be accessed and that the appropriate files will be processed. See Appendix A, “The TCP2PROD Command” on page 93 for information about the TCP2PROD command and TCP/IP Feature catalog files.

With the **TEST** option in effect, **no files are copied into production.**

Resolve any reported problems, then invoke TCP2PROD without the TEST option, as illustrated below.

```
tcp2prod 3tcpipa0 {tcpip | tcpipfs} 3tcpipa0 tcprun (setup
```

Use **tcpip** if the TCP/IP Feature default minidisk environment has been maintained; use **tcpipfs** if the service minidisks were moved to Shared File System directories.

- 9 Review the TCP2PROD message log (TCP2PROD \$MSGLOG). If necessary, correct any problems before going on.

```
vmfview tcp2prod
```

### 7.2.4.3 Copy New TCP/IP Feature Configuration Files Into Production

Use the TCP2PROD command, as described below, to place any new IBM-supplied sample configuration files into production.

#### Note

When you perform this step, only *new* configuration files are copied into production. Existing (that is, customized) configuration files are *not* replaced.

**Note:** For step 2 below, it's assumed that the TCPCONFIG section of the 3TCPIPA0 CATALOG has been verified, as recommended in 7.2.4.1, “3TCPIPA0 CATALOG File Modification Notes” on page 81. If this is not the case, you should make any necessary changes to this section of the 3TCPIPA0 CATALOG file before you continue with the steps that follow.

- 1 If necessary, establish the appropriate environment, as described by steps 1 through 5, in 7.2.4.2, “Copy Serviced TCP/IP Feature Run-time Files Into Production” on page 82.
- 2 Copy serviced TCP/IP Feature configuration files into production using the TCP2PROD command. For reference purposes, files that can be processed using the TCPCONFIG section are listed in Figure 15 on page 48.

#### Verifying Your Environment

When you perform this step, you should first issue the TCP2PROD command with the **TEST** option, to verify that all resources can be accessed and that the appropriate files will be processed. See Appendix A, “The TCP2PROD Command” on page 93 for information about the TCP2PROD command and TCP/IP Feature catalog files.

With the **TEST** option in effect, **no files are copied into production**. Resolve any reported problems, then invoke TCP2PROD without the TEST option, as illustrated below.

**tcp2prod 3tcpipa0 {tcpip | tcpipsfs} 3tcpipa0 tcpconfig (setup**

Use **tcpip** if the TCP/IP Feature default minidisk environment has been maintained; use **tcpipsfs** if the service minidisks were moved to Shared File System directories.

- 3 Review the TCP2PROD message log (TCP2PROD \$MSGLOG). If necessary, correct any problems before going on.

**vmfview tcp2prod**

#### RSU Service Instructions — Where To Next...

If you are installing **RSU** service, skip the section that follows and continue with the instructions provided in 7.2.4.5, “Configure the TCP/IP Feature for z/VM” on page 88.

## 7.2.4.4 Build and Copy Serviced TCP/IP Feature System Files Into Production

Use the TCP2PROD command, as described below, to build and then place any serviced TCP/IP Feature system files into production.

### Corrective Service-specific Processing

The steps described in this section are necessary **only** if Corrective (**COR**) service has been applied that has affected one or more TCP/IP (binary) system files (as listed in Figure 16 on page 50).

#### Notes:

1. For step 3 that follows, it's assumed that the TCPSYSFILE section of the 3TCPIPA0 CATALOG has been verified, as recommended in 7.2.4.1, "3TCPIPA0 CATALOG File Modification Notes" on page 81. If this is not the case, you should make any necessary changes to this section of the 3TCPIPA0 CATALOG file before you continue with the steps that follow.
2. You can avoid the need to perform step 2 that follows by creating a local modification to the 3TCPIPA0 CATALOG file which causes the DTCLFSR system file to be bypassed. See Appendix B, "Modifying TCP/IP Feature CATALOG Files" on page 100 for more information about how to change the 3TCPIPA0 CATALOG file in this manner. If this action is pursued, the entry that requires modification is shown here:

```
:TCPSYSFILE.  
...  
&DELTZ  &DISK201  DTCLFSR  TCPBLFSR =          DATABIN  SSLSERV  
...  
:ETCPSYSFILE.
```

- 1** If necessary, establish the appropriate environment, as described by steps 1 through 5, in 7.2.4.2, "Copy Serviced TCP/IP Feature Run-time Files Into Production" on page 82.

- 2 Update the service build table to modify DTCLFSR-related entries in the 3TCPIPA0 SRVBLDS file. These changes will prevent processing of the DTCLFSR system file, which is not used due to SSL-related packaging changes. Note that the commands shown here span two lines due to formatting restrictions:

```
vmfsim modify 3tcpipa0 srvbllds tdata :bldlist tcpblfsr :object bldlist  
:stat built.yy/mm/dd.nn:nn:nn.user-id (replace
```

```
vmfsim modify 3tcpipa0 srvbllds tdata :bldlist tcpblfsr :object dtclfsr.flsbin  
:stat built.yy/mm/dd.nn:nn:nn.user-id (replace
```

where:

- *yy/mm/dd* and *nn:nn:nn* reflect the (approximate) date and time these commands are issued
- *user-id* is the user ID from which these commands are issued.

- 3 Copy serviced TCP/IP Feature system files into production using the TCP2PROD command.

#### Verifying Your Environment

When you perform this step, you should first issue the TCP2PROD command with the **TEST** option, to verify that all resources can be accessed and that the appropriate files will be processed. See Appendix A, “The TCP2PROD Command” on page 93 for information about the TCP2PROD command and TCP/IP Feature catalog files.

With the **TEST** option in effect, **no files are copied into production.**

Note that TEST mode processing for TCP/IP Feature system files is somewhat limited (as compared with conventional run-time and configuration files) due to the processing requirements and characteristics of these files.

Resolve any reported problems, then invoke TCP2PROD without the TEST option, as illustrated below.

```
tcp2prod 3tcpipa0 {tcpip | tcpipfs} 3tcpipa0 tcpsysfile (setup
```

Use **tcpip** if the TCP/IP Feature default minidisk environment has been maintained; use **tcpipsfs** if the service minidisks were moved to Shared File System directories.

**Note:** For each *serviced* system file that is processed, the prompt that follows is issued before a production-level system file is created:

```
DTCPRD3004I Accessing target resource...
DTCPRD3097I Accessing resource vaddr (owner_id vaddr) with file mode fm2
DTCPRD3062I Decompacting file filename FLBnnnnn fm1 to file mode A...
DTCPRD3055I About to RESERVE target minidisk owner_id vaddr for system file:
filename DATABIN fm2
This will erase all files on this disk. Do you want to continue?
Enter 0 (No), 1 (Yes), 2 (Exit)
1
DTCPRD3060I Installing file filename DATABIN fm2 on minidisk owner_id vaddr
DTCPRD3068I Updating build status information for file: filename FL$BIN
DTCPRD3068I Updating build status information for file: TCPBLxxx (BLDLIST)
```

**4** Review the TCP2PROD message log (TCP2PROD \$MSGLOG). If necessary, correct any problems before going on.

**vmfview tcp2prod**

#### 7.2.4.5 Configure the TCP/IP Feature for z/VM

If any of the configuration files you have previously customized have been affected by service, you should review the updated *sample* equivalents of the those files, and determine if any changes are required to your production files.

See *z/VM: TCP/IP Level 3A0 Planning and Customization* (SC24-5981) for detailed explanations about how to configure the TCP/IP Feature servers. To accommodate service-related changes, you may at times need to consult APAR-specific documentation as well.



#### 7.2.4.6 (Optional) Copy Serviced TCP/IP Client Code to the z/VM Product Code Disk

If you previously copied TCP/IP Feature client code to the z/VM product code disk, you should replace the appropriate files with their serviced counterparts. See Appendix H, “Copying TCP/IP Feature Client Code to the Y-Disk” on page 127 for additional information and instructions concerning this process.

#### 7.2.4.7 Re-Initialize TCP/IP Services

After you have completed any necessary configuration, you need to re-initialize the appropriate TCP/IP servers.

For information on starting TCP/IP servers, see “Starting and Stopping TCP/IP Services” in Chapter 3 of *z/VM: TCP/IP Level 3A0 Planning and Customization*.

#### 7.2.4.8 (Optional) Verify the RSU Service Level

The procedures in this section describe how to determine the current TCP/IP Feature RSU service level by using either of these commands:

- TCP/IP NETSTAT LEVEL
- VMSES/E VMFSIM QUERY

The service contained on each RSU constitutes a new service level; this service level is updated in the system inventory when RSU service is installed.

**Note:** Corrective (COR) service does not affect the TCP/IP Feature RSU service level. However, you should record and use this service level when corrective (COR) service is ordered.

To use the TCP/IP **NETSTAT LEVEL** command, the TCPIP server (or, the TCP/IP *stack*) must be operating, and the TCP/IP NETSTAT command must be available. (In general, any user ID that has the TCPMAINT 592 minidisk accessed can issue this command).

**access 592 *fm***  
**netstat level**

where *fm* is an available file mode.

```
VM TCP/IP Netstat Level 3A0  
IBM 2064; z/VM Version 3 Release 1.0, service level 0101 (nn-BIT), VM TCP/IP Level  
3A0; RSU 0101
```

The last part of the returned output indicates the TCP/IP RSU service level; for this example, this is: **0101**

To use the **VMFSIM QUERY** command, the VMSES/E code (MAINT 5E5) and Software Inventory (MAINT 51D) minidisks must be accessed (thus, it is more readily issued while logged on the TCP/IP installation user ID, **3TCPIPA0**).

**vmfsim query vm sysrecs tdata :ppf 3tcpipa0 :stat**

```
VMFSIP2408I RESULTS FOR  
TDATA :PPF 3TCPIPA0 :STAT  
:PPF 3TCPIPA0 TCPIP  
:STAT RECEIVED.04/01/01.01:23:45.3TCPIPA0.RSU-0101
```

The last part of the status line indicates the TCP/IP RSU service level; for this example, this is: **0101**

**You have finished servicing the TCP/IP Feature for z/VM.**

---

### **7.3 Servicing the TCP/IP NFS Server Feature for z/VM**

The TCP/IP NFS Server Feature for z/VM does not have separate service. All service for this feature is provided through the service process for the TCP/IP Feature for z/VM, using its PPF and components. See the 7.2, “Servicing the TCP/IP Feature for z/VM” on page 65 for a description of the service procedures for the TCP/IP Feature for z/VM.

**You have finished servicing the TCP/IP NFS Server Feature for z/VM.**

---

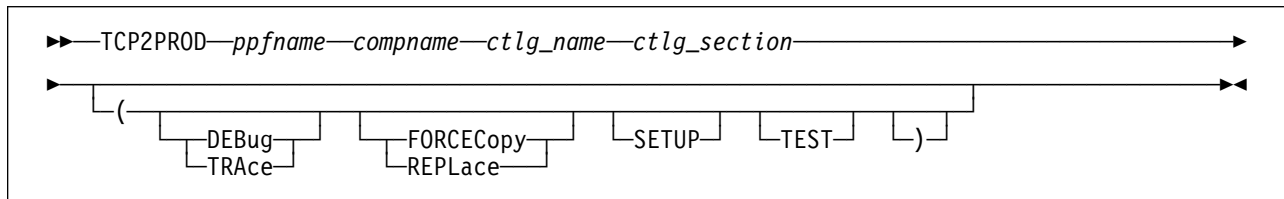
## 7.4 Servicing the TCP/IP Source Feature for z/VM

The TCP/IP Source Feature for z/VM does not have separate service. All service is provided through the service process for the TCP/IP Feature for z/VM, using its PPF and components. See the 7.2, “Servicing the TCP/IP Feature for z/VM” on page 65 for a description of the service procedures for the TCP/IP Feature for z/VM.

**You have finished servicing the TCP/IP Source Feature for z/VM.**

## Appendix A. The TCP2PROD Command

### A.1 TCP2PROD



#### A.1.1 Purpose

Use the TCP2PROD command to help you place TCP/IP Feature for z/VM files into production. The TCP2PROD command uses a TCP/IP Feature CATALOG file to determine which product files are to be placed into production, as well as which minidisks and SFS directories are to be used.

When TCP/IP Feature (binary) system files are processed, TCP2PROD also serves as a VMSES/E *part handler*, in that it manages build status information (maintained in the 3TCPIPA0 SRVBLDS file) for these types of files.

**Note:** The TCP2PROD command is intended for use by the 3TCPIPA0 user ID, and should only be used when you install or service the TCP/IP Feature and its features.

#### A.1.2 Operands

*ppfname* is the name of the usable form product parameter file to be used for this purpose. The file type must be PPF.

*compname* is the name of the component as specified on the :COMPNAME. tag in the product parameter file; *compname* is a 1- to 16-character alphanumeric identifier.

The PPF Variable Declarations (:DCL.) section defined for *compname* determines from which source minidisks and SFS directories product files are copied; likewise for the target minidisks to which these files are copied.

*ctlg\_name* is the name of the product catalog file to be processed. The file type must be CATALOG.

## TCP2PROD

*ctlg\_section* is the definition section of the catalog file to be processed. The value specified by *ctlg\_section* is used as a “root” for the *begin* and *end* tags that define each section of grouped entries (or, records) within a catalog file.

### A.1.3 Options

#### *DEBUG*

#### *TRACE*

causes additional messages to be issued to provide information for debugging purposes. The TRACE option is synonymous with the DEBUG option.

#### *FORCECopy*

#### *REPLace*

causes files identified within a CATALOG *configuration file* definition section to be copied to their configured name and type, regardless of whether such a configuration file already exists. The REPLACE option is synonymous with the FORCECOPY option.

#### *SETUP*

causes a VMSES/E VMFSETUP command to be issued as part of TCP2PROD processing, so that the correct operational environment is established. The *ppfname* and *compname* supplied when TCP2PROD is invoked are used as operands for the VMFSETUP command.

#### *TEST*

causes processing to be performed so that no files are placed into production via the VMSES/E VMFCOPY command (for conventional files), or through TCP2PROD-inherent processing (used for system files). This option allows you to verify that minidisks and SFS directories can be accessed as required, and that the appropriate catalog file entries will be processed. Additional messages are issued in some cases to clarify what processing would be performed if the TEST option was not used.

**Note:** TEST mode processing for TCP/IP Feature system files is somewhat limited (as compared with that for conventional run-time and configuration files) due to the processing requirements and characteristics of these files.

### A.1.4 Usage Notes

- **TCP2PROD does not issue LINK commands** for minidisks as part of its processing.  
  
If necessary, the appropriate LINK statements should be added to the 3TCPIPA0 directory entry. Alternatively, the required LINK commands can be added to the PROFILE EXEC of the 3TCPIPA0 user ID.
- The first CATALOG file found in the CMS search order that matches the name specified by *ctlg\_name* is used.

- CATALOG file entries which are found to be unusable are bypassed. Warning messages are issued when entries are bypassed for this reason.
- A minidisk or directory must be accessed at file mode A with Read/Write (R/W) status, for use as temporary work space and for message logging.
- By default, a file identified within a *configuration file* definition section of a CATALOG file is copied to its configured name and type only if the configured file does not already exist.

A catalog section is considered to be a *configuration file* definition section if its corresponding *begin* tag contains the string, **CONFIG**. For more information about CATALOG file structure and content, see A.2, "TCP/IP for VM CATALOG Files" on page 96.

- TCP2PROD processes system files by first decompacting service-level files to a temporary work space (file mode A) and then creating an appropriate binary image file under a unique CMS file name (for which a corresponding CMS minidisk has been reserved). At the same time, TCP2PROD serves as a VMSES/E *part handler*, in that it manages build status information (maintained in the 3TCPIPA0 SRVBLDS file) for these types of files. By design, only files that are identified within a *system file* definition section of a CATALOG file are processed in this manner.

A catalog section is considered to be a *system file* definition section if its corresponding *begin* tag contains the string, **SYSFILE**. For more information about CATALOG file structure and content, see A.2, "TCP/IP for VM CATALOG Files" on page 96.

- If TCP2PROD is invoked with *ppfname* specified as a question mark (?), the command syntax is displayed.

### A.1.5 The TCP2PROD \$MSGLOG File

Informational, warning and error messages issued to the console by TCP2PROD are maintained in a message log, TCP2PROD \$MSGLOG. This log file is written to the minidisk or directory accessed at file mode A, and can be viewed using the VMSES/E VMFVIEW command.

The TCP2PROD \$MSGLOG is cumulative. The most recent entries are appended at the **top** of the file. Separator headers that include date and time stamps are inserted in the log with each TCP2PROD invocation so that newer log entries can be distinguished from older ones.

No messages are logged until an initial validation of TCP2PROD operands is complete.

## TCP/IP for VM CATALOG Files

### A.1.6 Return Codes

The possible return codes produced by TCP2PROD are indicated below.

---

**0**

**Explanation:** Successful execution. No processing errors were encountered.

---

**1**

**Explanation:** Incorrect invocation. TCP2PROD was invoked with an incorrect number of operands. A message that identifies the missing operand is displayed, in addition to the command syntax.

---

**2**

**Explanation:** Internal error. If return code 2 is returned, processing status is unknown. You should contact the TCP/IP Feature support group for problem determination and assistance.

---

**4**

**Explanation:** Errors encountered, with warnings issued. The errors encountered may have caused processing to complete with only partial success. Review the TCP2PROD \$MSGLOG for warning messages which identify any problems that were encountered.

---

**8**

**Explanation:** Errors encountered. Processing has not completed successfully. Review the TCP2PROD \$MSGLOG for messages regarding the problems encountered.

---

## A.2 TCP/IP for VM CATALOG Files

### A.2.1 Purpose

The TCP/IP Feature CATALOG files are used by TCP2PROD to determine which TCP/IP Feature product files are to be placed into production, as well as which minidisks and SFS directories are to be used for this process.

The CATALOG files provided for use with the TCP/IP Feature for z/VM are listed in Figure 17.

<b>Catalog File</b>	<b>Corresponding TCP/IP Feature</b>
3TCPIPA0 CATALOG	TCP/IP Feature (Base)



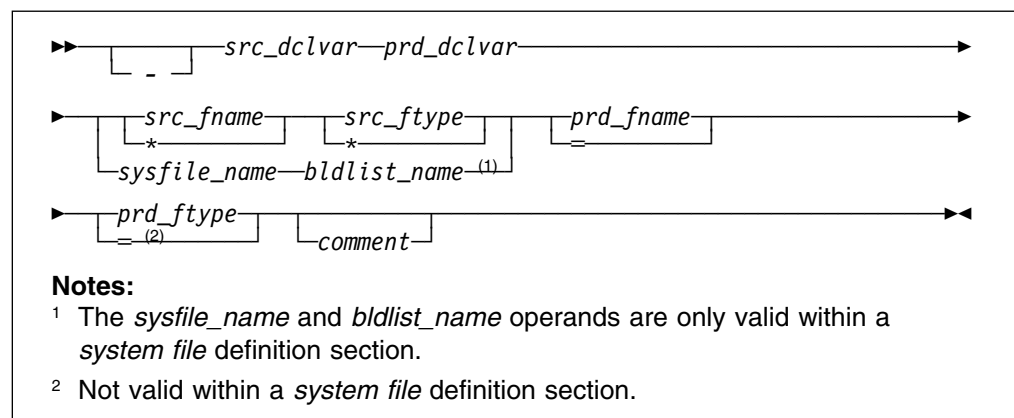
Distinct sections within the CATALOG file are used to identify *product* files, such as the TCPIP MODULE, and *sample* files which need to be customized, such as the TCPIP SDATA file. Each definition section is delimited by a pair of *begin* and *end* tags. For example, the definition section for TCP/IP Feature sample configuration files is defined using the :TCPCONFIG. and :ETCPCONFIG. tags.

The entries within each section provide information sufficient for TCP2PROD to process a group of files associated with a given TCP/IP feature.

Unless marked otherwise, all entries within a given section are processed and the corresponding files are placed into production via the VMSES/E VMFCOPY command (for conventional files) or through additional processing inherent to the TCP2PROD command (for system files).

### A.2.2 CATALOG Record Syntax

The syntax for TCP/IP Feature CATALOG file entries is indicated below.



### A.2.3 Operands

- is the entry bypass character, a hyphen (-). Use this bypass character at the beginning of a file entry to indicate that it should *not* be processed by TCP2PROD.

**Note:** It's recommended that all file entries remain present within the various TCP/IP Feature CATALOG files. If a TCP/IP Feature file is not used in your environment, its corresponding entry should be bypassed as described above. Doing so will also help readily distinguish entries which correspond to TCP/IP Feature product files from comment lines.

*src\_dclvar* is a PPF :DCL. variable name for the minidisk or SFS directory where the source file resides.

## TCP/IP for VM CATALOG Files

- prd\_dclvar* is a PPF :DCL. variable name for the minidisk or SFS directory where the production or configured file is to reside.
- src\_fname* is the source file name. An asterisk (\*) can be used to indicate that all files of the type specified by *src\_ftype* are to be processed. When this wildcard (\*) is used, the production or configured file name remains unchanged from the source file name.
- src\_ftype* is the source file type. An asterisk (\*) can be used to indicate that all files of the name specified by *src\_fname* are to be processed. When this wildcard (\*) is used, the production or configured file type remains unchanged from the source file type.
- prd\_fname* is the configured file name. If the source file name is specified using an asterisk (\*), *prd\_fname* must be specified to maintain the correct entry format. In such cases, an equal sign (=) should be used; a value other than an equal sign is processed as if an equal sign had been specified.
- prd\_ftype* is the production or configured file type. If the source file type is specified using an asterisk (\*), *prd\_ftype* must be specified to maintain the correct entry format. In such cases, an equal sign (=) should be used; a value other than an equal sign is processed as if an equal sign had been specified.
- Note:** An equal sign cannot be used to specify a production file type within a *system file* definition section; a “conventional” CMS file type must be specified within this type of definition section.
- sysfile\_name* is the file name of a (binary) source system file. This operand is only valid within a *system file* definition section.
- bldlist\_name* is the name of the VMSES/E build list with which a preceding *sysfile\_name* file is associated. For TCP/IP Feature system files, a one-to-one correspondence exists for a given build list and a given system file. This operand is only valid within a *system file* definition section.
- comment* commentary text that is ignored by TCP2PROD during processing.

### Notes:

1. All operands must be separated by at least one space.
2. Comment lines may be included in CATALOG files. Comment lines must begin with an asterisk (\*); these lines are ignored during TCP2PROD processing.

### A.2.4 Usage Notes

- TCP/IP Feature CATALOG file changes that are required for your environment should be made via a VMSES/E local modification, so that any service-related changes can be reported during the VMSES/E service process. See Appendix B, “Modifying TCP/IP Feature CATALOG Files” on page 100 for more information about how to change the TCP/IP Feature CATALOG files in this manner.
- The source and target minidisk/directory variable names used within this file correspond to those used within an applicable TCP/IP feature PPF file (or an override variation of that file). If any changes are made to the Variable Declarations (:DCL.) section of a TCP/IP Feature PPF file via a PPF override, you may need to incorporate similar changes within the various CATALOG files (through a separate VMSES/E local modification) to allow for the correct resolution of PPF :DCL. variable names.

---

## Appendix B. Modifying TCP/IP Feature CATALOG Files

This appendix describes how to create a VMSES/E local modification for TCP/IP Feature base or feature CATALOG files, which are provided as replacement-maintained (or, “full-source”) objects. This procedure can also be used to modify other replacement-maintained TCP/IP Feature files, such as sample files.

The example provided in this appendix describes how to modify the 3TCPIPA0 CATALOG file, so that files associated with the BOOTP server will not be processed by the TCP2PROD EXEC.

For more information about installing and maintaining local modifications, see Chapters 5, 6, and 7 of the *z/VM: Service Guide* (GC24-5946).

### Notes:

1. Throughout this procedure, references are made to several TCP/IP Feature installation minidisks. Default addresses, and equivalent SFS directories (when applicable), for these minidisks are listed in the tables in 5.3, “DASD Storage and User ID Requirements” on page 23.
2. VMSES/E local modifications require a modification identifier, or *modid*, to be associated with the parts affected by a change; *modid* is a locally-determined value. It should begin with **L**, and is followed by up to 4 alphanumeric characters that identify a specific local modification. For example, L0001.

### **1** Log on the installation user ID, **3TCPIPA0**.

The PROFILE EXEC provided for this user ID (as part of the z/VM Version 3 Release 1.0 System DDR) contains ACCESS commands for the required VMSES/E minidisks — the VMSES/E code minidisk (MAINT 5E5, by default) and the VMSES/E Software Inventory minidisk (MAINT 51D, by default).

### **2** Issue the CMS QUERY DISK command to verify the VMSES/E code and Software Inventory minidisks are correctly linked and accessed.

#### query disk

Verify the MAINT 5E5 minidisk is accessed as file mode **B**, and is linked **R/O**.

Verify the MAINT 51D minidisk is accessed as file mode **D**, and is linked **R/W**.

**Note:** If another user has the MAINT 51D minidisk linked in write (R/W) mode, you'll obtain only read (R/O) access to this minidisk. If this occurs, you'll need to have that user re-link the 51D disk in read-only (RR) mode; then, you need issue the appropriate LINK and ACCESS commands for the 51D minidisk. Do not continue with these procedures until a R/W link is established to the 51D minidisk.

**3** If necessary, establish the appropriate access to the VMSES/E minidisks.

**a** Establish read access to the VMSES/E code minidisk.

```
link MAINT 5e5 5e5 rr
access 5e5 b
```

**b** Establish write access to the Software Inventory minidisk.

```
link maint 51d 51d mr
access 51d d
```

**4** Create a usable PPF file.

**Note:** This VMFPPF step is necessary **only** if you have made changes to the PPF file since it was last compiled.

```
vmfppf ppfname compname
```

where:

- *ppfname* is the name of a PPF file that corresponds to the configuration file to be modified. For example, **3tcpipa0**. If you have created an override PPF file, use your override file name.
- *compname* is the feature component name with which the configuration file is associated. For example, for the TCP/IP Feature (3TCPIPA0), you would use either: **tcpip** or **tcpipsfs**

**5** Establish the correct minidisk access order.

```
vmfsetup ppfname compname
```

*ppfname* and *compname* are the same as described in step 4.

- 6** Copy the highest level of the file to be modified to the local modification (LOCALMOD) minidisk, with the correct file type for the replacement part.

**vmfrepl** *fn ft ppfname compname (\$select logmod modid outmode localmod*

where:

- *fn* and *ft* are the real file name and file type of the file being modified. For example:  
3TCPIPA0 CATALOG
- *ppfname* and *compname* are the same as described in step 4 on page 101.
- *modid* is the local modification identifier for this part. (*modid* is described in note 2 on page 100).
- **localmod** is the symbolic name for the local modification disk in the :MDA. section of the 3TCPIPA0 PPF file.

- 7** XEDIT the part just copied to the LOCALMOD disk, so you can make (and then file) your changes.

**xedit** *fn ftabbrv-modid fm-local*

The file ID (*fn ftabbrv-modid fm-local*) is returned in message VMFREP2507I. For example:  
3TCPIPA0 CATL0001 E

## 8 Make your changes to the displayed file.

For example, assume you've chosen to not use the BOOTP server provided with TCP/IP Feature, because you are instead using the supplied DHCP server. To avoid TCP2PROD processing of the files associated with the BOOTP server, you need to bypass the 3TCPIPA0 CATALOG file shown here:

```
:TCPRUN.  
...  
- &BLD1Z &DISK4 TCPROFIL EXEC PROFILE = BOOTPD  
...  
:ETCPRUN.  
...  
...  
:TCPCONFIG.  
...  
- &BLD0Z &BLD4Z BOOTPTAB SAMPLE ETC BOOTPTAB BOOTPD  
...  
:ETCPCONFIG.
```

## 9 When you have completed your changes, save the replacement part.

====> file

## 10 Rebuild the Modified Objects

### Note

When you have completed steps 6 through 9 for all of the local modifications necessary for this component, then rebuild these objects.

Build the modified (or, serviced) configuration file(s).

**vmfbld ppf ppfname compname (serviced**

---

## Appendix C. PTF Commit Procedure for TCP/IP Feature System Files

As TCP/IP Feature (binary) system files are updated through corrective service, the storage space required to maintain multiple PTF levels for these parts is likely to become constrained. To allow new service for these files to be received and applied, it will at times be necessary to *commit* existing PTF service — this causes a service status of 'COMMITTED' to be reflected for a PTF, and its corresponding parts are deleted.

**Note:** For some environments, it may be necessary to commit existing maintenance prior to receiving *any* new corrective service that affects TCP/IP Feature system files, depending on how much **DELTA** disk storage has been allocated for maintaining serviced files.

The TCP/IP Feature system files for which commit processing is likely to be required are listed in Figure 18:

System File Identifier
DTCSOLF FLSBIN DTCSOLS FLSBIN

The procedure that follows illustrates how to use the VMSES/E VMFREM command to commit service associated with a specific PTF for a TCP/IP Feature system file.

**1** Log on the installation user ID, **3TCPIPA0**.

The PROFILE EXEC provided for this user ID (as part of the z/VM Version 3 Release 1.0 System DDR) contains ACCESS commands for the required VMSES/E minidisks — the VMSES/E code minidisk (MAINT 5E5, by default) and the VMSES/E Software Inventory minidisk (MAINT 51D, by default).

**2** Issue the CMS QUERY DISK command to verify the VMSES/E code and Software Inventory minidisks are correctly linked and accessed.

**query disk**

Verify the MAINT 5E5 minidisk is accessed as file mode **B**, and is linked **R/O**.

Verify the MAINT 51D minidisk is accessed as file mode **D**, and is linked **R/W**.

**Note:** If another user has the MAINT 51D minidisk linked in write (R/W) mode, you'll obtain only read (R/O) access to this minidisk. If this occurs, you'll need to have that user re-link the 51D disk in read-only (RR) mode; then, you need issue the appropriate LINK and ACCESS commands for the



51D minidisk. Do not continue with these procedures until a R/W link is established to the 51D minidisk.

**3** If necessary, establish the appropriate access to the VMSES/E minidisks.

**a** Establish read access to the VMSES/E code minidisk.

```
link MAINT 5e5 5e5 rr
access 5e5 b
```

**b** Establish write access to the Software Inventory minidisk.

```
link maint 51d 51d mr
access 51d d
```

**4** Create a usable PPF file.

**Note:** This VMFPPF step is necessary **only** if you have made changes to the PPF file since it was last compiled.

**vmfppf** *ppfname compname*

where:

- *ppfname* is the name of the PPF file used for TCP/IP maintenance purposes at your installation. For example, **3tcpipa0**. If you have created an override for the PPF file in question, use your override file name.
- *compname* is the feature component name with which the source file is associated. For example, for the TCP/IP Feature (3TCPIPA0), you would use either: **tcpip** or **tcpipsfs**

**5** Establish the correct minidisk access order.

**vmfsetup** *ppfname compname*

where *ppfname* and *compname* are the same as described in step 4.

**6** Identify the PTF-numbered parts (files) for which commit processing is appropriate and generate a file that lists the PTFs that are associated with these parts. The REXX and CMS Pipelines code that follows will accomplish this task.

**Note:** When you first perform the steps provided in this appendix, you will need to create (or obtain) an exec that incorporates this processing.

An enhanced version of the **TCPCMLST EXEC** is available from the TCP/IP support group which provides notification as to when a base-level (FLB00000)

file can safely be removed from the DELTA disk, in addition to creating the PTF removal list. The source for the enhanced TCPCLST EXEC is also available from the TCP/IP Feature home page on the World Wide Web. The URL for this home page is:

**<http://www.vm.ibm.com/related/tcpip/>**

```

/*-----*/
/* TCPCLST EXEC -- (Minimum format/required processing) */
/*-----*/
Arg delta_fm .
out_file = '3TCPIPA0 $REMLIST A' /* Define output file */
rc = 0
If (Datatype(delta_fm,'U') <> 1) /* Verify filemode OK */
Then Exit 1
'PIPE' ,
'Command LISTFILE * FLB*' delta_fm , /* List FLBnnnn parts */
'| Spec w2 1' , /* FLBnnnn ftype only */
'| Change 1-3 /FLB/UQ/' , /* Change to PTF nums */
'| Sort 1-7' , /* Order list */
'| Nlocate 1-7 /'UQ00000'/' , /* Discard base level */
'| Stem outdata.' , /* Save data to check */
'| >' out_file /* Write output file */
cmd_rc = rc /* Save PIPE retcode */
If (Left(Word(outdata.1,1),2) <> 'UQ') /* If erroneous data, */
Then 'PIPE' 'Command ERASE' out_file '| Hole' /* erase output file */
If (outdata.0 = 0) /* Adjust retcode if */
Then cmd_rc = 28 /* no data produced. */
Exit cmd_rc

```

Invoke the TCPCLST exec to create the 3TCPIPA0 \$REMLIST file that will be used in the next step.

**tcpclst** *fm-delta*

where:

- *fm-delta* is the file mode at which the 3TCPIPA0 **DELTA** minidisk or directory is accessed.

- 7** Commit the PTF service for the parts in question, as identified within the 3TCPIPA0 \$REMLIST file.

**vmfrem ppf *ppfname compname* list 3tcpipa0 \$remlist a (commit**

where:

- *ppfname* and *compname* are the same as described in step 4 on page 105.

Upon successful completion of the VMFREM command, the parts associated with the PTF in question are erased and its status in the receive software inventory table (*recid* SRVRECS) is set to 'COMMITTED'.

---

## Appendix D. Making Local Modifications to TCP/IP Feature Modules

This appendix provides information to assist you in making local modifications to TCP/IP Feature for z/VM modules, and is intended to supplement the local modification process described in Chapter 6, “Procedures for Local Service” of the *z/VM: Service Guide* (GC24-5946). The process described herein is oriented toward local service that is implemented through:

- TEXT file **replacement** — when an existing TCP/IP TEXT file is fully replaced and the file name of the existing TEXT file remains unchanged
- **local updates** to the source for an existing TEXT file (so that a “local replacement” is effectively created)
- TEXT file **substitution** or **addition**, in which an existing TCP/IP TEXT file is fully replaced by one or more new, differently-named TEXT files.

With regard to the latter two types of modification, additional steps to accommodate changes to TCP/IP C and Pascal-based modules or added TEXT files are provided.

### When to Use This Information

In general, the VMSES/E local modification process as documented in Chapter 6, “Procedures for Local Service” of the *z/VM: Service Guide* should be used when local modifications are made. *This is especially true* for **common** part types — such as EXEC and ASSEMBLE files — for which VMSES/E itself provides appropriate part handlers.

However, C or Pascal-specific VMSES/E part handlers are not available for TCP/IP **C** and **Pascal** source parts, so local modifications to TCP/IP Feature modules based on these languages cannot be processed completely within the VMSES/E environment. Instead, the VMFC and VMFPAS execs provided with the TCP/IP Feature can be used to facilitate local modifications to its C and Pascal-based modules. Additional steps which make use of these execs, are included in this appendix to help you modify TCP/IP C or Pascal-based objects.

Information about the VMFC and VMFPAS execs can be found in Chapter 31, “Using Source Code Libraries,” of *z/VM: TCP/IP Level 3A0 Planning and Customization* (SC24-5981).

Note that VMFC and VMFPAS are **not** “VMSES/E-compliant” part handlers. They do not process source files in the same manner as VMSES/E part handlers (for example, by using VMFEXUPD); nor do they produce object (TEXT) files that comply with VMSES/E maintenance naming conventions (such as file abbreviations and part numbering). Therefore, several manual steps are required to ensure the TEXT files produced by VMFC and VMFPAS can be used as part of the VMSES/E maintenance process.

For detailed information about installing and maintaining local modifications, see Chapters 5, 6, and 7 of the *z/VM: Service Guide*.

## Notes:

1. Throughout this procedure, references are made to several TCP/IP Feature installation minidisks. Default addresses, and equivalent SFS directories (when applicable), for these minidisks are listed in the tables in 5.3, “DASD Storage and User ID Requirements” on page 23.
2. VMSES/E local modifications require a modification identifier, or *modid*, to be associated with the parts affected by a change; *modid* is a locally-determined value. It should begin with **L**, and is followed by up to 4 alphanumeric characters that identify a specific local modification. For example: L0002

### D.1.1 Prepare for Local Service

- 1** Log on the installation user ID, **3TCPIPA0**.

The PROFILE EXEC provided for this user ID (as part of the z/VM Version 3 Release 1.0 System DDR) contains ACCESS commands for the required VMSES/E minidisks — the VMSES/E code minidisk (MAINT 5E5, by default) and the VMSES/E Software Inventory minidisk (MAINT 51D, by default).

- 2** Issue the CMS QUERY DISK command to verify the VMSES/E code and Software Inventory minidisks are correctly linked and accessed.

#### query disk

Verify the MAINT 5E5 minidisk is accessed as file mode **B**, and is linked **R/O**.

Verify the MAINT 51D minidisk is accessed as file mode **D**, and is linked **R/W**.

**Note:** If another user has the MAINT 51D minidisk linked in write (R/W) mode, you'll obtain only read (R/O) access to this minidisk. If this occurs, you'll need to have that user re-link the 51D disk in read-only (RR) mode; then, you need issue the appropriate LINK and ACCESS commands for the 51D minidisk. Do not continue with these procedures until a R/W link is established to the 51D minidisk.

- 3** If necessary, establish the appropriate access to the VMSES/E minidisks.

- a** Establish read access to the VMSES/E code minidisk.

```
link MAINT 5e5 5e5 rr
access 5e5 b
```

- b** Establish write access to the Software Inventory minidisk.

```
link maint 51d 51d mr
access 51d d
```

#### 4 Create a usable PPF file.

**Note:** This VMFPPF step is necessary **only** if you have made changes to the PPF file since it was last compiled.

**vmfppf** *ppfname compname*

where:

- *ppfname* is the name of a PPF file that corresponds to the source file to be modified. For example, **3tcpipa0**. If you have created an override for the PPF file in question, use your override file name.
- *compname* is the feature component name with which the source file is associated. For example, for the TCP/IP Feature (3TCPIPA0), you would use either: **tcpip** or **tcpipsfs**

#### 5 Establish the correct minidisk access order.

**vmfsetup** *ppfname compname*

where *ppfname* and *compname* are the same as described in step 4.

#### 6 Access the TCP/IP Feature Source minidisk (or SFS directory).

**access 2b3** *fm*

where *fm* is an available file mode, perhaps "C."

## D.1.2 Receive the Local Service

### D.1.2.1 Create the Replacement TEXT File

Create the replacement TEXT file (or, "text deck"). How you perform this step depends on the nature of the TEXT file itself.

- If your modification is comprised of a **replacement** TEXT file, you need only to copy this file to an appropriately named file which VMSES/E can then use.
- If your modification is comprised of a **new** (additional) TEXT file, you need to copy this file to an appropriately named file and update the build list that corresponds to the module affected by your changes.

If either of the above cases are applicable, skip the steps provided for "C and Pascal Source Modifications," and create the required *TXTmodid* file, as described in D.1.2.5, "Copy the TEXT File for Use by VMSES/E" on page 113.

- If your modification requires updates to a C or Pascal source file, continue with the steps provided for “C and Pascal Source Modifications.”

### D.1.2.2 Add an Update Record to the AUX File

- 1 Update or create the AUX file for the part being modified (*fn* AUXLCL) and add an entry for the update file; the default AUX file type is “AUXLCL.”

**Note:** Perform this step for each source file you need to modify.

```
xedit fn auxlcl fm-local (noprof
====> input update-ft svclvl lcmoidid comment
====> file
```

where:

- *fn* is the file name of the source-maintained part being modified.
- *fm-local* is the file mode of the TCP/IP Feature local modification (LOCALMOD) disk.
- *update-ft* is the file type of the update file that contains your local modifications.
- *svclvl* is a service level indicator; by convention, the string “LCL” is most often used for local modifications.
- *moidid* is the local modification number (as described in Note 2 on page 109). For example: L0002
- *lcmoidid* is the string “LC” concatenated with *moidid*; for example: LCL0002

### D.1.2.3 Create the Update File for the Part

- 1 Xedit the source file (*fn* C or *fn* PASCAL) with the **CTL** option.

**Note:** Perform this step for each source file you need to modify.

```
xedit fn ft-src fm-src (ctl tcpip
```

where:

- *fn ft-src fm-src* are the file name, file type and file mode of the source file you are modifying.
- **tcpip** is the name of the TCP/IP control file.

- 2 Make your changes to the displayed source file. The original source file is *not* changed.
- 3 When you have completed your changes, save them on the TCP/IP Feature local modification (LOCALMOD) disk. When you enter the FILE command in XEDIT, all of your changes are placed in the update file (*fn update-ft*).

====> file = = *fm-local*

where *fm-local* is the file mode of the TCP/IP Feature local modification (LOCALMOD) disk.

#### D.1.2.4 Create an Updated Replacement TEXT File

- 1 Compile the source to include your updates. Use the VMFC EXEC to compile C source files; use the VMFPAS EXEC to compile PASCAL source files. For more information about the VMFC and VMFPAS execs, see Chapter 31, "Using Source Code Libraries," of *z/VM: TCP/IP Level 3A0 Planning and Customization (SC24-5981)*.

**Note:** Perform this step for each source file you have modified.

**a** To compile **C**-based source files, issue:

**vmfc** *fn tcpip* ( *options*

**OR**

**b** To compile **Pascal**-based source files, issue:

**vmfpas** *fn tcpip* ( *options*

where:

- *fn* is the name of the source file to be compiled.
- **tcpip** is the name of the TCP/IP control file.
- *options* are C or PASCAL compiler options required for your environment.



After successful completion of the VMFC (or VMFPAS) EXEC, a text file (*fn* TXTLCL A) incorporating your local modifications will exist.

**Note:** If you choose to use a different compilation method instead of using the VMFC or VMFPAS execs provided with the TCP/IP Feature, the resulting TEXT file may be named differently than “*fn* TXTLCL.”

---

End of C and Pascal Source Modifications

### D.1.2.5 Copy the TEXT File for Use by VMSES/E

Copy the replacement or new TEXT file (or, “text deck”) to the TCP/IP Feature local modification (LOCALMOD) disk, with the correct file type for the replacement part.

**Note:** Perform this step for each TEXT file affected by your modifications.

**vmfrepl** *fn text ppfname compname fn ft-txt fm-txt* (\$select **logmod modid outmode localmod ftabbr txt**

where:

- *fn* is the file name of the TEXT file in question.
- *ppfname* and *compname* are the same as described in step 4 on page 110.
- *fn ft-txt fm-txt* are the file name, file type and file mode of the TEXT file in question.
- *modid* is the local modification number (as described in Note 2 on page 109). For example: L0002
- **localmod** is the symbolic name for the local modification disk in the :MDA. section of the 3TCIPA0 PPF file.

### D.1.3 (Optional) Modify the Build List for Added TEXT Files

If your local modifications do not require the addition of a new TEXT file, skip the steps provided in “Additional Steps for Adding a New TEXT File,” and continue with the steps provided in D.1.4, “Rebuild the Modified Objects” on page 116.

If your local modifications are implemented within a unique TEXT file (that is, one not supplied by IBM as part of the TCP/IP Feature) you need to perform additional steps for this TEXT file to be incorporated by the affected TCP/IP module. These steps, provided in “Additional Steps for Adding a New TEXT File,” implement build list changes that will accommodate the new TEXT file; they would also be required

if you find it necessary to replace an IBM-supplied TEXT file with one of a different name.

---

Additional Steps for Adding a New TEXT File

### D.1.3.1 Add an Update Record to the Build List AUX File

**1** Update or create the build list AUX file (*fn-blst* AUXLCL) and add an entry for the update file; the default file AUX file type is “AUXLCL.”

A complete list of TCP/IP feature build lists is provided in Appendix F, “TCP/IP Feature Build Lists” on page 120.

```
xedit fn-blst auxlcl fm-local (noprof
====> input update-ft svclvl lcmmodid comment
====> file
```

where:

- *fn-blst* is the file name of the build list used to build the affected module (that is, the build list to which the new TEXT file must be added).
- *fm-local* is the file mode of the TCP/IP Feature local modification (LOCALMOD) disk.
- *update-ft* is the file type of the update file that contains your local modifications.
- *svclvl* is a service level indicator; by convention, the string “LCL” is most often used for local modifications.
- *modid* is the local modification number (as described in Note 2 on page 109). For example: L0002
- *lcm**modid* is the string “LC” concatenated with *modid*; for example: LCL0002

### D.1.3.2 Create the Update File for the Build List

**1** Xedit the build list source file with the **CTL** option.

```
xedit fn-blst $exec fm-src (ctl tcPIP
```

where:

- *fn-blst* and *fm-src* are the file name and file mode of the build list file you are modifying.
- **\$exec** is the file type of the build list.
- **tcPIP** is the name of the TCP/IP control file.

- 2 Make your changes to the displayed source file. The original source file is *not* changed.

Locate the :OBJNAME. tag associated with the name of the module that will include your text file. For example, TCPIP.MODULE. Add a new :PARTID. tag record after the last :PARTID. tag for this object (TCPIP.MODULE), and before its :EOBJNAME. tag.

The new :PARTID. tag should define the file name of the text file you're adding, followed its file type abbreviation, TXT. Use only a single space to separate the tag, the text file name, and the TXT abbreviation, as follows

```
:PARTID. newtxtfn TXT
```

In the example that follows, the TCPBLM91 build list is to be updated to add TEXT files to the TCPIP module. To do this, line 176 is copied (or duplicated); then, in the newly created line(s), the string "TCOFFPR" is changed to the name of the added TEXT file(s).

```

TCPBLM91 EXCnnnnn I2 V 80 Trunc=80 Size=265 Line=178 Col=1 Alt=nn
====>
00087 :EOBJNAME.
00088 *
00089 :OBJNAME. TCPIP.MODULE AMODE 31 FROM TCPIP NOMAP
00090 :BLDREQ. TCPBLCOM.BLDLIST TCPBL492.TCPASCAL.TXTLIB TCPBL492.TCPLAN
00091         TCPBL492.TCPXXA.TXTLIB
00092 :GLOBAL. TXTLIB COMMTXT TCPXXA TCPASCAL TCPLANG
00093 :OPTIONS. CLEAR NOAUTO RLDSAVE NOLIBE NOUNDEF RMODE 24
00094 :PARTID. TCPIP TXT
00095 ----- 80 line(s) not displayed -----
00175 :OPTIONS. LIBE UNDEF RESET VSPASCAL
00176 :PARTID. TCOFFPR TXT
00177 :EOBJNAME.
00178 *

```

- 3 When you have completed your changes, save them on the TCP/IP Feature local modification (LOCALMOD) disk. When you enter the FILE command in XEDIT, all of your changes are placed in the update file (*fn-blst update-ft*).

====> file = = *fm-local*

where *fm-local* is the file mode of the TCP/IP Feature local modification (LOCALMOD) disk.

### D.1.3.3 Create the Updated Replacement Build List

- 1 Create a replacement part from the build list \$EXEC (source) file. To do this, update the build list file with the VMFEXUPD command.

**vmfexupd** *fn-blst* **exec** *ppfname compname* (**outmode localmod \$select logmod**)

where:

- *fn-blst* is the file name of the build list being updated. For example, TCPBLM91
- *ppfname* and *compname* are the same as described in step 4 on page 110.

The VMFEXUPD command records the update for the build list in the local version vector table (VVTLCL), adds the update to the \$SELECT file, and creates the replacement part (*fn-blst* EXEC).

End of Additional Steps for Adding a New TEXT File

## D.1.4 Rebuild the Modified Objects

### Note

When you have completed the steps described in D.1.2, “Receive the Local Service” through D.1.2.5, “Copy the TEXT File for Use by VMSES/E” (or D.1.3.3, “Create the Updated Replacement Build List”) for all of the local modifications necessary for this component, then rebuild the modified objects.

In general, your modification is likely to require other steps associated with the service process to be completed (such as updating the build status table, re-building serviced objects, testing service, and copying the service into production). To complete the service process, continue with one of the steps listed, as appropriate:

- 7.2.2.4, “Update the Build Status Table” on page 70, to complete TCP/IP Feature RSU service
- 7.2.3.4, “Update the Build Status Table” on page 78, to complete TCP/IP Feature COR service

For reference, a sample VMFBLD command is shown below that can be used to rebuild specific objects within a given build list:

```
vmfbld ppf ppfname compname fn-blst fn-mod.module (serviced
```

where:

- *ppfname* and *compname* are the same as described in step 4 on page 110.
- *fn-blst* is the appropriate TCP/IP Feature build list file name. A complete list of TCP/IP Feature build lists is provided in Appendix F, “TCP/IP Feature Build Lists” on page 120.
- *fn-mod* is the file name of module which incorporates the TEXT file that has been modified.

Also, if you have modified only one module, you may want to manually copy it into production, instead of using the TCP2PROD EXEC. If this is the case, use the VMFCOPY command that follows:

```
vmfcopy fn module fm-bld = = fm-prd (oldd repl sprodid 3tcpipa0%tcpip prodid 3tcpipa0%tcpip
```

where:

- *fn* is the name of the modified module.
- *fm-bld* is the file mode of the TCP/IP Feature build disk on which the module was built.
- *fm-prd* is the file mode of the TCP/IP Feature production disk where the module should reside.

The VMFCOPY command will update the VMSES/E PARTCAT file on the appropriate TCP/IP Feature production minidisk.

---

## Appendix E. Modifying TCP/IP NFS Server Feature VMNFS Code

This appendix provides information to assist you in making local modifications to the TCP/IP NFS Server Feature VM NFS server module (VMNFS). Such modifications would be required for the NFS server to use a file handle encryption subroutine different from that in NFSHCIP ASSEMBLE, or to change the way SMSG requests are validated (NFSSMSG C) or failed minidisk link attempts are reported (NFSBADPW C). Certain modifications may also require changes to the TCPBLC91 EXEC, which is the build list used to build the VMNFS module.

- | **Note:** The source for the NFSHCIP ASSEMBLE file can be found on the TCP/IP Feature for z/VM  
| **BASE** minidisk (**3TCPIPA0 2B2**, by default); that for the the TCPBLC91 EXEC is present on the TCP/IP  
| Feature **SOURCE** minidisk (**3TCPIPA0 2B3**, by default). The remainder of the TCP/IP NFS Server  
| Feature source files are provided only as part of the TCP/IP Source Feature for z/VM; such files are  
| shipped in packed format, and may need to be unpacked before they can be used.

For detailed information about installing and maintaining local modifications, see Chapters 5, 6, and 7 of the *z/VM: Service Guide* (GC24-5946).

---

### E.1 Modifying the NFSHCIP ASSEMBLE and TCPBLC91 EXEC Files

If you need to modify the NFSHCIP ASSEMBLE or TCPBLC91 EXEC files, you should follow the steps provided in Chapter 6, “Procedures for Local Service” of the *z/VM: Service Guide* (GC24-5946). In so doing, the following substitutions may need to be made:

- **zvm** should be: **3tcpipa0**
- *compname* should be: **tcpip** or **tcpipsfs**
- *appid* should be: **3tcpipa0**
- *fm-local* should be the file mode of the 2C4 minidisk
- *fm-applyalt* should be the file mode of the 2A6 minidisk

You may also find some of the information provided in Appendix D, “Making Local Modifications to TCP/IP Feature Modules” on page 108 to be useful, such as the steps provided in D.1.1, “Prepare for Local Service.”

Keep in mind that when you get to the following step in the *z/VM: Service Guide*:

- “Rebuilding Objects”

you should return to using this program directory and continue with 7.2.2.4, “Update the Build Status Table” on page 70.

---

## E.2 Modifying VMNFS C Source Files

If you need to modify the source for the NFSSMSG or NFSBADPW C files, use the information provided in Appendix D, “Making Local Modifications to TCP/IP Feature Modules” on page 108, especially that which is specific to “C and Pascal Source Modifications.”

## Appendix F. TCP/IP Feature Build Lists

This appendix provides a complete list of the VMSES/E build lists used to maintain the TCP/IP Feature for z/VM product. This information has been provided to help you determine which build list to use with VMSES/E commands when you need to build or service specific TCP/IP Feature objects, and assist you with making local modifications. For more information about build list content and formats, see the *z/VM: VMSES/E Introduction and Reference (GC24-5947)*.

The build lists identified in the tables that follow can be found on the 3TCPIPA0 2B2 (BASE1) minidisk. However, before using the information in a build list, the 3TCPIPA0 2D2 (DELTA) minidisk should be checked to determine whether that build list has been updated by service, so that the most current file is referenced.

Also, note that the minidisks shown under the "Build String" headings are default values for 3TCPIPA0 minidisks. If a PPF override has been used in your environment to change Build String minidisks or SFS directories, use your values when determining which files are affected by a build list.

### F.1 TCP/IP Feature for z/VM Build Lists

The Figure 19 lists the VMSES/E build lists used with the TCP/IP Feature for z/VM, and provides general information about the objects (files) managed by each build list.

<i>Figure 19 (Page 1 of 2). VMSES/E Build Lists - TCP/IP Feature for z/VM</i>			
<b>Build List Name</b>	<b>VMSES/E Part Handler</b>	<b>Build String (Minidisk)</b>	<b>Build List Description / Affected Objects</b>
TCPBL491	VMFBDCOM	BUILD1 (491)	Full-replacement objects built to the 491 minidisk
TCPBL492	VMFBDCOM	BUILD3 (492)	Full-replacement objects built to the 492 minidisk
TCPBLALL	VMFBDMLB	BUILD3 (492)	ALLMACRO MACLIB build list
TCPBLTRP	VMFBDMLB	BUILD3 (492)	TFTPRP MACLIB build list
TCPBLCOM	VMFBDTLB	BUILD3 (492)	COMMTXT TXTLIB build list
TCPBLGDD	VMFBDTLB	BUILD3 (492)	GDDMXD TXTLIB build list
TCPBLXAW	VMFBDTLB	BUILD3 (492)	XAWLIB TXTLIB build list
TCPBLDPI	VMFBDTLB	BUILD3 (492)	DPILIB TXTLIB build list
TCPBLRPC	VMFBDTLB	BUILD3 (492)	RPCLIB TXTLIB build list
TCPBLRPT	VMFBPMD	BUILD3 (492)	VMRPC TXTLIB build list
<b>Notes:</b>			
1. Language Environment run-time library support must be available when building objects identified in this build list.			



Figure 19 (Page 2 of 2). VMSES/E Build Lists - TCP/IP Feature for z/VM

Build List Name	VMSES/E Part Handler	Build String (Minidisk)	Build List Description / Affected Objects
TCPBLDOLD	VMFBDTLB	BUILD3 (492)	OLDXLIB TXTLIB build list
TCPBLXTL	VMFBDTLB	BUILD3 (492)	XTLIB TXTLIB build list
TCPBLX11	VMFBDTLB	BUILD3 (492)	X11LIB TXTLIB build list
TCPBLSNA	VMFBDLLB	BUILD1 (491)	SNALINK LOADLIB build list
TCPBLXNX	VMFBDLLB	BUILD1 (491)	XNX25 LOADLIB build list
TCPBLSNM	VMFBDLLB	BUILD1 (491)	SNMPLIB LOADLIB build list
TCPBLM91	VMFBDMOD	BUILD1 (491)	MODULE objects built to the 491 minidisk
TCPBLM92	VMFBDMOD	BUILD3 (492)	MODULE objects built to the 492 minidisk
TCPBLC91 (1*)	VMFBDMOD	BUILD1 (491)	C-based MODULE objects built to the 491 minidisk
TCPBLC92 (1*)	VMFBDMOD	BUILD3 (492)	C-based MODULE objects built to the 492 minidisk
TCPBLMPR (1*)	VMFBDTLB	BUILD1 (491)	MPROUTE MODULE build list
TCPBLNFC (1*)	VMFBDTLB	BUILD3 (492)	NFS (Client) MODULE build list
TCPBLCSL	VMFBDCLB	BUILD1 (491)	TCPCSLIB CSLIB build list
TCPBLFSF	TCP2PROD	N/A	DTCSSLF system file build list
TCPBLFSS	TCP2PROD	N/A	DTCSSLS system file build list
TCPBLFSR	TCP2PROD	N/A	DTCLFSR system file build list
TCKBLDES	VMFBDTLB	BUILD3 (492)	DES TXTLIB build list
TCKBLKRB	VMFBDTLB	BUILD3 (492)	KRB TXTLIB build list
TCKBLKDB	VMFBDTLB	BUILD3 (492)	KDB TXTLIB build list
TCKBLBPL	VMFBDTLB	BUILD3 (492)	BPLDBM TXTLIB build list
TCKBLC91 (1*)	VMFBDMOD	BUILD1 (491)	Kerberos C-based MODULE objects built to the 491 minidisk
TCKBLC92 (1*)	VMFBDMOD	BUILD3 (492)	Kerberos C-based MODULE objects built to the 492 minidisk
<b>Notes:</b>			
1. Language Environment run-time library support must be available when building objects identified in this build list.			

---

## Appendix G. Moving the TCP/IP Feature to SFS Directories

By default, the TCP/IP Feature for z/VM is installed to minidisks, as part of the z/VM Version 3 Release 1.0 System DDR, during the initial install of z/VM Version 3 Release 1.0 itself. However, you can move certain TCP/IP Feature minidisks — the service minidisks — to Shared File System (SFS) directories. Refer to Figure 10 in 5.3.1, “DASD Requirements for the TCP/IP Feature for z/VM” on page 26 to see which minidisks can reside in SFS space. When service disks are moved to SFS directories, you can use either the default file pool (VMSYS) or a file pool of your choosing.

A summary of the steps necessary to move TCP/IP Feature service minidisks to SFS space are:

- Allocate space in the user-defined (or default) file pool
- Provide the installation user ID, 3TCPIPA0, access to the file pool
- Create the necessary TCP/IP Feature subdirectories
- Copy files from minidisks to the new SFS subdirectories
- Create a PPF override, if not using default file pool or subdirectory names.

### Where to Next

You need to do one of the following:

- If you want to place the TCP/IP Feature into SFS directories using **3TCPIPA0 PPF file defaults**, you need to continue with instructions provided in the *z/VM: Installation Guide*, in the appendix titled “Moving Components to SFS Directories.”
- If you want to place the TCP/IP Feature into your own **user-defined file pool or SFS directories**, continue with the instructions in G.1.1, “Setup the SFS File Pool and Required Directories” on page 123.

After you have chosen one of the previous options and completed the steps required (whether provided in this appendix or in the *z/VM: Installation Guide*) you need to return to 6.2.3, “Enable the TCP/IP Feature” on page 38 and complete the installation of the TCP/IP Feature for z/VM.

## G.1.1 Setup the SFS File Pool and Required Directories

### Notes:

1. The steps that follow help you determine TCP/IP Feature file pool space requirements, enroll the 3TCPIPA0 user ID in a file pool, and define the required SFS directories. If not all of these steps are required — for example, you are changing only SFS directory names — adapt these instructions as needed for your environment.
2. The steps that follow assume the use of a user-defined SFS file pool name; if you are using the z/VM default file pool name (VMSYS), you should substitute “VMSYS” in place of the text “user-defined file pool” or the variable *userfilepool*, when appropriate.
3. For information about planning for, generating, and managing a file pool and server, see *z/VM: CMS File Pool Planning, Administration, and Operation* (SC24-5949).

- 1** Determine the number of 4K blocks required for your SFS directories by adding up the 4K block requirements for each SFS directory you plan to use. If you intend to use *all* of the TCP/IP Feature SFS directory defaults, the 4K block requirements are summarized in Figure 10 on page 26.

This information will be used when the 3TCPIPA0 user ID is enrolled in the user-defined file pool.

- 2** Enroll user 3TCPIPA0 in the user-defined file pool, using the ENROLL USER command:

```
ENROLL USER 3TCPIPA0 userfilepool: (BLOCKS blocks)
```

where *blocks* is the number of 4K blocks you calculated in the previous step.

**Note:** This must be done from a user ID that is an administrator for the user-defined file pool.

- 3** Determine if there are enough blocks available in the file pool to accommodate the TCP/IP Feature for z/VM. This information can be obtained via the QUERY FILEPOOL STATUS command. Near the end of the output from this command is a list of minidisks in the file pool and the number of blocks free. If the number of blocks free is smaller than the total number needed to install the TCP/IP Feature, you need to add space to the file pool before you continue with this process. See *z/VM: CMS File Pool Planning, Administration, and Operation* for information about adding space to a file pool.

- 4 Use the CREATE DIRECTORY command to create the required subdirectories; the default subdirectories are listed in Figure 10 on page 26. If necessary, refer to the *z/VM: CMS Command Reference* for more information about the CREATE DIRECTORY command.

```
set filepool userfilepool:
create directory dirid
create directory dirid
```

*dirid* is the name of the SFS directory you're creating, such as the default names:

```
create directory userfilepool:3tcpipa0.tcpip
create directory userfilepool:3tcpipa0.tcpip.local
create directory userfilepool:3tcpipa0.tcpip.delta
create directory userfilepool:3tcpipa0.tcpip.applyalt
create directory userfilepool:3tcpipa0.tcpip.applyprod
create directory userfilepool:3tcpipa0.tcpip.object
```

- 5 If you intend to use an SFS directory as the work space for the 3TCPIPA0 user ID, include the following IPL control statement in the 3TCPIPA0 directory entry (after the INCLUDE TCPCMSU statement):

```
IPL CMS PARM FILEPOOL userfilepool
```

This will cause CMS to automatically access the 3TCPIPA0's top directory as file mode A.

## G.1.2 Copy Minidisk Files to SFS Directories

- 1 Copy the files from the TCP/IP Feature for z/VM minidisks (or from the VMSYS file pool, if the TCP/IP Feature is already installed there) to your new user-defined SFS file pool and directories, using the VMFCOPY command.

**Note:** Repeat the ACCESS and VMFCOPY commands that follow for each minidisk you need to copy. If necessary, see Figure 10 on page 26 for TCP/IP Feature default minidisk addresses and SFS directory names.

```
access vdev e
access dirid f
vmfcopy * * e = = f (prodid 3tcpipa0%tcpip olddate replace
```

where:

- *vdev* is the minidisk from which you are copying files.
- *dirid* is the name of the (target) SFS directory to which you are copying files.
- ***3tcpipa0%tcpip*** is the PRODID defined within the 3TCPIPA0 PPF file, for both the minidisk and SFS components of the TCP/IP Feature.

The VMFCOPY command will update the VMSES PARTCAT file on the target directory.

### G.1.3 Create a Product Parameter File (PPF) Override

This section provides information to help you create a product parameter file (PPF) override. The example used in this section changes the name of the SFS file pool where TCP/IP Feature files reside. See the *z/VM: VMSES/E Introduction and Reference* for more information about PPF file overrides.

**Note:** Do **not** directly modify the product-supplied 3TCPIPA0 \$PPF or 3TCPIPA0 PPF files to change the VMSYS file pool name or any other installation parameters. If the 3TCPIPA0 \$PPF file is serviced, the existing \$PPF file will be replaced, and any changes to that file will be lost. By creating your own \$PPF override, your updates will be preserved.

The following process describes changing the TCP/IP Feature default file pool name from “VMSYS” to “MYPOOL1”:

- 1 Create a new \$PPF override file or edit an existing override file.

**xedit** *overname \$ppf fm2*

*overname* is the PPF override file name (such as "mytcpip") that you want to use.

*fm* is an appropriate file mode. If you create this file yourself, specify a file mode of A.

If you modify an existing override file, specify a file mode of A or D, based on where the file currently resides (A being the file mode of a R/W 191 minidisk, or equivalent; D, that of the MAINT 51D minidisk).

- 2** Create (or modify as required) the Variable Declarations (:DCL.) section for the **TCPIPSFS** override area so it resembles the :DCL. section that follows:

```

:OVERLIST. TCPIPSFS
:TCPIPSFS. TCPIP 3TCPIPA0
*
*=====
* Override for TCPIPSFS SFS Component -- File Pool Name Change      *
*=====
:DCL. UPDATE
&191  DIR  MYPOOL1:3TCPIPA0                * A-disk
&LMODZ DIR  MYPOOL1:3TCPIPA0.TCPIP.LOCAL   * Local modifications
&DELTZ DIR  MYPOOL1:3TCPIPA0.TCPIP.DELTA   * Product service
&APPLX DIR  MYPOOL1:3TCPIPA0.TCPIP.APPLYALT * AUX and Inv files (ALT)
&APPLZ DIR  MYPOOL1:3TCPIPA0.TCPIP.APPLYPROD * AUX and Inventory file
&BAS1Z DIR  MYPOOL1:3TCPIPA0.TCPIP.OBJECT  * Base disk
&BAS2Z DIR  MYPOOL1:3TCPIPA0.TCPIP.SOURCE  * Source code (Optional)
:EDCL.
:END.

```

This override will update the TCPIPSFS override area :DCL. section of the 3TCPIPA0 \$PPF file.

- 3** Save your changes when they are complete.

====> file = = fm

where *fm* is the file mode where your override file resides (or A if you are creating one).

- 4** If your \$PPF override file was created at file mode A, copy it to file mode D — the Software Inventory minidisk (MAINT 51D).

copyfile overname \$ppf fm = = d (olddate

- 5** Compile your changes to create the usable PPF file, **overname PPF**.

vmfppf overname tcpipsfs

where *overname* is the file name of your \$PPF override file.

Now that the **overname PPF** file has been created, specify “*overname*” instead of “3TCPIPA0” as the PPF name to be used for any VMSES/E commands that require a PPF name.

---

## Appendix H. Copying TCP/IP Feature Client Code to the Y-Disk

To simplify access to TCP/IP client functions for your user community, you may find it desirable to copy all, or a subset of, the TCP/IP Feature client code to the z/VM product code disk. Doing so will avoid the need for users to additionally link and access the TCPMAINT 592 minidisk.

As well, applications that use certain programming interfaces may require TCP/IP-specific information to be available for proper operation. For example, information defined in the TCPIP DATA file is referenced by:

- the OpenEdition VM Sockets Library to correctly identify the TCP/IP virtual machine when the TCPIPID environment variable is not set. See the *C for VM/ESA: Library Reference* (SC23-3908) for more information.
- the VMTCPDT routine (which resides in the VMMLIB TXTLIB that is associated with the VMLIB Callable Services Library (CSL). See the *z/VM: CMS Callable Services Reference* (SC24-5959) for more information about the VMTCPDT CSL routine.
- various functions provided as part of the CMS REXX Socket library. See the *z/VM: REXX/VM Reference* (SC24-5963) for more information.

To copy TCP/IP Feature for z/VM client files to the Product Code minidisk (typically the MAINT 19E minidisk — the Y-disk), use the following procedure **after** you have installed TCP/IP Feature and *all* TCP/IP features used in your environment.

### Warning - File Overlap Considerations

**Before** you copy *any* TCP/IP Feature for z/VM client files to the Y-disk (or a similar “*common use*” minidisk), you should first determine whether any conflicts exist between the TCP/IP client files you choose to copy, and those present on the target (Y-disk) minidisk. If any file conflicts are found, these should be addressed and resolved with respect to your installation environment before you continue with the procedure that follows.

### Notes:

1. You will need to repeat this procedure each time you apply service to the TCP/IP feature.
2. Use discretion when wildcards (\*) are used for both the *fn* (file name) and *ft* (file type) parameters of the VMFCOPY commands shown in this section, since files that exist on the Y-disk can be replaced with similarly-named TCP/IP counterpart files. The overlay of certain files may be warranted in some cases, and may be undesirable for others.

An example of this latter case is cited here. Both the TCP/IP Feature for z/VM and the IBM Language Environment for MVS & VM, Version 1 Release 8 have several **H** files that are identically named, but differ in content. These files are:

FCNTL	H	IF	H	IN	H	INET	H
IOCTL	H	NETDB	H	RESOURCE	H	SOCKET	H
STRINGS	H	TTYDEV	H	TYPES	H	UIO	H

An overlay of the IBM Language Environment **H** files (already present on the Y-disk) by their TCP/IP counterparts may create problems when applications that expect (and rely upon) the content of the IBM Language Environment files are developed or rebuilt.

3. Before copying TCP/IP Feature files to another minidisk, ensure adequate storage space is available to maintain the files you have selected.

- 1** Log on the **MAINT** (or equivalent) user ID.

- 2** Process TCP/IP Feature files used by or available to TCP/IP clients.

**link tcpmaint 592 592 rr**  
**access 592 e**  
**access 19e f**

**Note:** If the Y-disk is not defined as the 19E minidisk in your environment, substitute the appropriate address for this minidisk.

**vmfcopy *fn ft e = = f2* (olddate replace sprodid 3tcpipa0%tcpip prodid 3tcpipa0%tcpip**

The VMFCOPY command will update the VMSES/E PARTCAT file on the Y-disk.

Wildcards (\*) can be substituted for *fn* (file name) and *ft* (file type), but should be used with discretion.

- 3** (Optional) Erase any TCP/IP Feature files that you do not want on the Y-disk — for example, any MAP files that correspond to TCP/IP Feature modules re-built during service. Refer to the VMSES/E PARTCAT file on Y-disk to determine which files are associated with TCP/IP Feature.

**Note:** Additional information about the various TCP/IP Feature client files, and their association with specific TCP/IP functions, is available on-line via the TCP/IP Feature home page on the World Wide Web. The URL for this home page is:

<http://www.vm.ibm.com/related/tcpip/>

**vmferase file *filename filetype f***

See the *z/VM: VMSES/E Introduction and Reference* for more information about the VMFERASE command and options that may help you remove specific files.



- 4 Re-save the CMS saved system, to return the Y-disk to shared status. See the “Placing (Serviced) Components into Production” section of the *z/VM: Service Guide* for detailed information about how to save the CMS saved system.

---

## Appendix I. Disabling the TCP/IP and TCP/IP NFS Server Features

If you are running z/VM Version 3 Release 1.0, or higher, and had previously enabled either the TCP/IP NFS Server Feature or the TCP/IP Feature for z/VM, but are no longer authorized for one of these features, you must explicitly disable that feature. To disable the TCP/IP Feature, perform the steps that follow.

- 1** Logon the installation user ID, **3TCPIPA0**.  
The PROFILE EXEC provided for this user ID (as part of the z/VM Version 3 Release 1.0 System DDR) contains ACCESS commands for the required VMSES/E minidisks — the VMSES/E code minidisk (MAINT 5E5, by default) and the VMSES/E Software Inventory minidisk (MAINT 51D, by default).
- 2** Issue the CMS QUERY DISK command to verify the VMSES/E code and Software Inventory minidisks are correctly linked and accessed.

### query disk

Verify the MAINT 5E5 minidisk is accessed as file mode **B**, and is linked **R/O**.

Verify the MAINT 51D minidisk is accessed as file mode **D**, and is linked **R/W**.

**Note:** If another user has the MAINT 51D minidisk linked in write (R/W) mode, you'll obtain only read (R/O) access to this minidisk. If this occurs, you'll need to have that user re-link the 51D disk in read-only (RR) mode; then, you need issue the appropriate LINK and ACCESS commands for the 51D minidisk. Do not continue with these procedures until a R/W link is established to the 51D minidisk.

- 3** If necessary, establish the appropriate access to the VMSES/E minidisks.
  - a** Establish read access to the VMSES/E code minidisk.  
**link MAINT 5e5 5e5 rr**  
**access 5e5 b**
  - b** Establish write access to the Software Inventory minidisk.  
**link maint 51d 51d mr**  
**access 51d d**

#### 4 Set the TCP/IP Feature for z/VM to the disabled state.

**vmfins disable ppf** *ppfname compname*

If you are disabling the TCP/IP Feature, specify **3TCPIPA0** for *ppfname*. For *compname*, use **tcpip** if the TCP/IP Feature default minidisk environment has been maintained; use **tcpipsfs** if the service disks were moved to Shared File System directories.

If you are disabling the TCP/IP NFS Server Feature, specify **3TCNFSA0** for *ppfname*. For *compname*, use **nocomp**.

This command will:

- set the TCP/IP or TCP/IP NFS Server Feature as DISABLED in VMSES/E and CP
- create a *ppfname* PRODSYS file on the A-disk, which contains a CP system configuration PRODUCT statement with a state of: DISABLED

#### 5 In order to **retain the DISABLED state** for the TCP/IP or TCP/IP NFS Server Feature, **update the CP system configuration file** with the information in the *ppfname* PRODSYS file.

The *ppfname* PRODSYS file was created by VMFINS in the previous step. You need to update the CP system configuration file so that the TCP/IP or TCP/IP NFS Server Feature will be in the DISABLED state the next time you IPL your system. If you are not authorized to update the CP system configuration file, contact your CP system administrator.

The CP system configuration file resides on the CP “parm” disk (the MAINT CF1 or CF2 minidisk); by default, this file is named: SYSTEM CONFIG. The necessary updates need to be made to the “PRODUCT ENABLE/DISABLE INFORMATION” section of this file. You will need to replace the current ENABLED entry for PRODID 3TCPIPA0 or PRODID 3TCNFSA0 (the TCP/IP or TCP/IP NFS Server Feature) with the entry from the *ppfname* PRODSYS file (created by VMFINS in the previous step). If necessary, refer to the *z/VM: Planning and Administration* manual for more information about the CP system configuration file.

#### 6 Shutdown the applicable server (TCPIP or VMNFS), if this has not already been done.

---

## Notices

IBM® may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation  
Licensing  
2-31 Roppongi 3-chome, Minato-ku  
Tokyo 106, Japan

**The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:** INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes to the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation  
Mail Station P300  
2455 South Road  
Poughkeepsie, NY 12601-5400  
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities on non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

#### COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to IBM programming interfaces. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

---

## Trademarks and Service Marks

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

FunctionPac  
IBM  
IBMLink  
Language Environment  
RACF  
VM/ESA  
z/VM

Other company, product, and service names may be trademarks or service marks of others.

# Reader's Comments

## TCP/IP Feature for z/VM

You may use this form to comment about this document, its organization, or subject matter. Please understand that your feedback is of importance to IBM, but IBM makes no promises to always provide a response to your feedback. If you prefer to provide feedback electronically, please e-mail your comments to: **vmtcpip@vnet.ibm.com**

For each of the topics below please indicate your satisfaction level by circling your choice from the rating scale. If a statement does not apply, please circle N.

RATING SCALE						
<b>very satisfied</b>	←—————→				<b>very dissatisfied</b>	<b>not applicable</b>
<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>N</b>	

	Satisfaction					
	1	2	3	4	5	N
Ease of product installation						
Time required to install the product						
Contents of program directory						
Readability and organization of program directory tasks						
Necessity of all installation tasks						
Accuracy of the definition of the installation tasks						
Technical level of the installation tasks						
Installation verification procedure						
Ease of customizing the product						
Ease of migrating the product from a previous release						
Ease of putting the system into production after installation						
Ease of installing service						

- Did you order this product as an independent product or as part of a package?

- Independent
- Package

What type of package was ordered?

- CustomPac
  - FunctionPac®
  - SystemPac
- System Delivery Offering (SDO)
- Other - Please specify type: \_\_\_\_\_

- Is this the first time your organization has installed this product?
  - Yes
  - No
- Were the people who did the installation experienced with the installation of VM products using VMSES/E?
  - Yes
    - How many years of experience do they have? \_\_\_\_\_
  - No
- How long did it take to install this product? \_\_\_\_\_
- If you have any comments to make about your ratings above, or any other aspect of the product installation, please list them below:

---

---

---

---

---

---

---

---

---

---

Please provide the following contact information:

---

Name and Job Title

---

Organization

---

Address

---

Telephone

---

**Thank you for your participation.**

Please send the completed form to the following address, or give to your IBM representative who will forward it to the TCP/IP Feature Development group:

IBM Corporation  
 TCP/IP for VM Development  
 Dept. G79G  
 1701 North Street  
 Endicott, NY 13760







Program Number: 5654-A17

Printed in U.S.A.

G110-4710-00

