

z/OS Communications Server



IP Messages: Volume 2 (EZB, EZD)

Version 1 Release 6

z/OS Communications Server



IP Messages: Volume 2 (EZB, EZD)

Version 1 Release 6

Note:

Before using this information and the product it supports, be sure to read the general information under “Notices” on page 287.

Fifth Edition (September 2004)

This edition applies to Version 1 Release 6 of z/OS (5694-A01) and Version 1 Release 6 of z/OS.e (5655-G52) and to all subsequent releases and modifications until otherwise indicated in new editions.

IBM welcomes your comments. You may send your comments to the following address.

International Business Machines Corporation
Attn: z/OS Communications Server Information Development
Department AKCA, Building 501
P.O. Box 12195, 3039 Cornwallis Road
Research Triangle Park, North Carolina 27709-2195

You can send us comments electronically by using one of the following methods:

Fax (USA and Canada):

1+919-254-4028

Internet e-mail:

- comsvrcf@us.ibm.com

World Wide Web:

<http://www.ibm.com/servers/eserver/zseries/zos/webqs.html>

If you would like a reply, be sure to include your name, address, telephone number, or FAX number. Make sure to include the following in your comment or note:

- Title and order number of this document
- Page number or topic related to your comment

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 1994, 2004. All rights reserved.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

About this document	v
Who should read this document.	v
How this document is organized	v
How to use this document	vi
Determining if a publication is current	vi
How to contact IBM service	vi
Conventions and terminology used in this document	vii
The IBM TCP/IP message standards	vii
Terms and abbreviations used in this document	ix
Clarification of notes.	x
Prerequisite and related information	xi
Required information	xi
Related information.	xi
How to send your comments	xv
 Summary of changes	 xvii
 Chapter 1. EZB0xxxx messages.	 1
 Chapter 2. EZB1xxxx messages	 71
 Chapter 3. EZB2xxxx messages	 87
 Chapter 4. EZB3xxxx messages.	 137
 Chapter 5. EZB4xxxx messages.	 217
 Chapter 6. EZB6xxxx messages.	 247
 Chapter 7. EZB8xxxx messages.	 249
 Chapter 8. EZB9xxxx messages.	 251
Common messages	251
 Chapter 9. EZD0xxxx messages.	 253
 Chapter 10. EZD1xxxx messages	 257
 Appendix A. Message ranges and the components that issue them	 267
Volume 1 ranges	267
Volume 2 ranges	267
Volume 3 ranges	268
Volume 4 ranges	268
 Appendix B. Related protocol specifications (RFCs).	 271
Internet Drafts	280
 Appendix C. Information APARs	 281
Information APARs for IP documents	281
Information APARs for SNA documents	282
Other information APARs	283

Appendix D. Accessibility	285
Using assistive technologies	285
Keyboard navigation of the user interface	285
z/OS information	285
Notices	287
Trademarks	295
Bibliography.	297
z/OS Communications Server information	297
z/OS Communications Server library	297
Communicating Your Comments to IBM	303

About this document

This document describes the Internet Protocol (IP) messages that occur in z/OS® Communications Server. The information in this document supports both IPv6 and IPv4. Unless explicitly noted, information describes IPv4 networking protocol. IPv6 support is qualified within the text.

For information about how to set up, initialize, and customize your Transmission Control Protocol/Internet Protocol (TCP/IP) services system, refer to the *z/OS Communications Server: IP Configuration Reference*, the *z/OS Communications Server: IP Configuration Guide* and the *z/OS Communications Server: IP Programmer's Reference*. For information about how to use the applications on your TCP/IP system, refer to *z/OS Communications Server: IP User's Guide and Commands*.

This document supports z/OS.e.

Who should read this document

This document assists TCP/IP operators, system programmers, and users to:

- Analyze a problem
- Classify the problem as a specific type
- Describe the problem to the IBM® Software Support Center

Familiarity with TCP/IP concepts and terms is assumed.

How this document is organized

The messages are listed in alphanumeric order by message ID. For each message ID, the book contains the text and a description of the message. This book contains the following chapters:

- Chapter 1, "EZB0xxxx messages," on page 1 contains messages in the range of EZB0600I—EZB0999I.
- Chapter 2, "EZB1xxxx messages," on page 71 contains messages in the range of EZB1000I—EZB1230I.
- Chapter 3, "EZB2xxxx messages," on page 87 contains messages in the range of EZB2000I—EZB2498E.
- Chapter 4, "EZB3xxxx messages," on page 137 contains messages in the range of EZB3000I—EZB3999I.
- Chapter 5, "EZB4xxxx messages," on page 217 contains messages in the range of EZB4000I—EZB4198I.
- Chapter 6, "EZB6xxxx messages," on page 247 contains message EZB6473I.
- Chapter 7, "EZB8xxxx messages," on page 249 contains message EZB801I.
- Chapter 8, "EZB9xxxx messages," on page 251 contains common messages that are called by several application and function components. These messages are in the range of EZB9395I—EZB9428I.
- Chapter 9, "EZD0xxxx messages," on page 253 contains messages in the range of EZD0000I—EZD0801I.
- Chapter 10, "EZD1xxxx messages," on page 257 contains messages in the range of EZD1161I—EZD1162I.

- Appendix A, "Message ranges and the components that issue them," on page 267 contains a list of all message ranges and components.
- Appendix B, "Related protocol specifications (RFCs)," on page 271 lists the related protocol specifications for TCP/IP.
- Appendix C, "Information APARs," on page 281 lists information APARs for IP and SNA documents.
- Appendix D, "Accessibility," on page 285 describes accessibility features to help users with physical disabilities.
- "Notices" on page 287 contains notices and trademarks used in this document.
- "Bibliography" on page 297 contains descriptions of the documents in the z/OS Communications Server library.

How to use this document

To use this document, you should be familiar with z/OS TCP/IP Services and the TCP/IP suite of protocols.

Determining if a publication is current

As needed, IBM updates its publications with new and changed information. For a given publication, updates to the hardcopy and associated BookManager® softcopy are usually available at the same time. Sometimes, however, the updates to hardcopy and softcopy are available at different times. The following information describes how to determine if you are looking at the most current copy of a publication:

- At the end of a publication's order number there is a dash followed by two digits, often referred to as the dash level. A publication with a higher dash level is more current than one with a lower dash level. For example, in the publication order number GC28-1747-07, the dash level 07 means that the publication is more current than previous levels, such as 05 or 04.
- If a hardcopy publication and a softcopy publication have the same dash level, it is possible that the softcopy publication is more current than the hardcopy publication. Check the dates shown in the Summary of Changes. The softcopy publication might have a more recently dated Summary of Changes than the hardcopy publication.
- To compare softcopy publications, you can check the last two characters of the publication's filename (also called the book name). The higher the number, the more recent the publication. Also, next to the publication titles in the CD-ROM booklet and the readme files, there is an asterisk (*) that indicates whether a publication is new or changed.

How to contact IBM service

For immediate assistance, visit this Web site:
<http://www.software.ibm.com/network/commserver/support/>

Most problems can be resolved at this Web site, where you can submit questions and problem reports electronically, as well as access a variety of diagnosis information.

For telephone assistance in problem diagnosis and resolution (in the United States or Puerto Rico), call the IBM Software Support Center anytime (1-800-IBM-SERV). You will receive a return call within 8 business hours (Monday – Friday, 8:00 a.m. – 5:00 p.m., local customer time).

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

If you would like to provide feedback on this publication, see “Communicating Your Comments to IBM” on page 303.

Conventions and terminology used in this document

For definitions of the terms and abbreviations used in this document, you can view the latest IBM terminology at the IBM Terminology Web site.

The IBM TCP/IP message standards

This section describes the message numbering conventions used in the IP Message manuals.

Current numbering convention

The following diagram shows the message numbering convention currently in use.

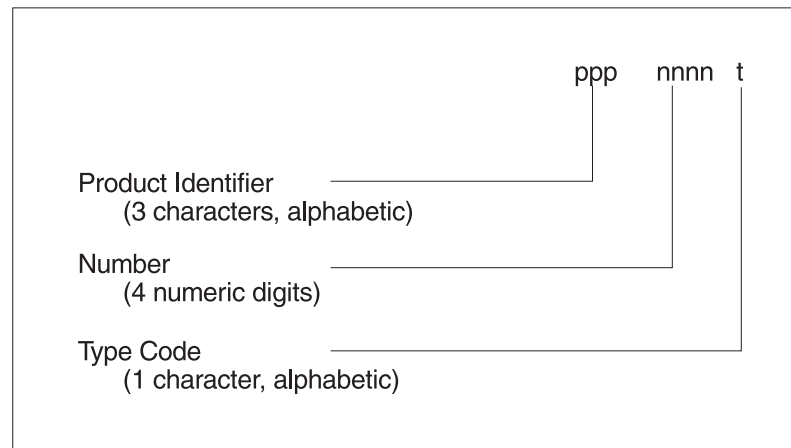


Figure 1. Sample message formats

The *product identifiers* (ppp) for TCP/IP are EZA, EZB, EYZ, EZZ and SNM. The *number* (nnnn) indicates a unique 4-digit numeric value assigned to the message by product. The *type* (t) indicates the action assigned to the message. The message ID is followed by a single space and the message text.

For example:

EZZ0902I Use a host name after HOST option.

The various type code characters are defined below.

Letter	Meaning
--------	---------

A	Immediate Action required.
E	Eventual Action required.
D	Immediate Decision required.
I	Informational.

Older numbering convention

The following diagram shows the older message numbering convention that was used for TCP/IP applications.

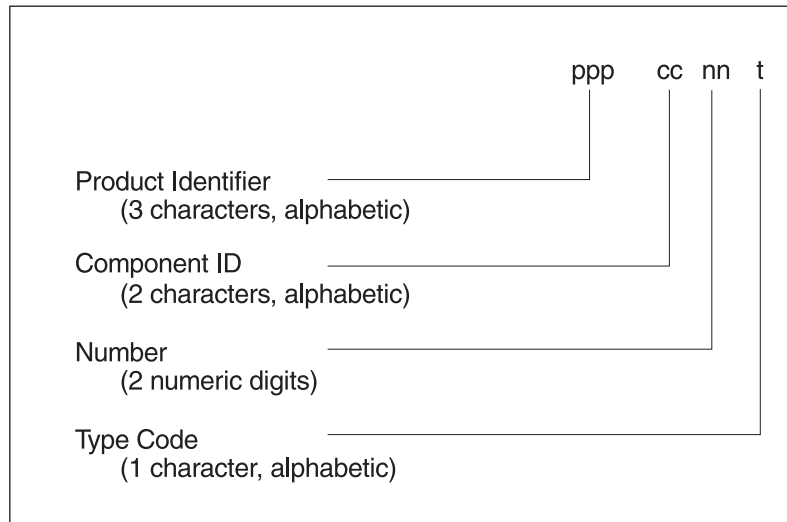


Figure 2. Sample message formats

The various type code characters are defined below.

Type Code

Severity

E or S A recoverable error occurred. Follow the instructions given in the document to fix the problem.

E,T, or U

An unrecoverable error that terminated the program occurred. You might need to contact the IBM Software Support Center.

I Informational message.

N Nondisplay. The message text, but not the message ID, appeared.

W Warning. A condition that might subsequently cause an error has occurred.

X Concatenation. This message is adjoined to one or more messages.

The component identifiers used by some messages with the EZY prefix are listed in the following table.

Table 1. EZY message component identifiers

EZY component ID	Component name
FS	FTP Server
FT	FTP Server
RC	orexec Client
RD	rexecd Daemon
RG	RPCGEN
RS	RSHD Client
TE	Telnet Server - Executive
TO	Telnet Server - Other

Table 1. EZY message component identifiers (continued)

EZY component ID	Component name
TS	Telnet Server - State
TU	Telnet Server - Utility
TY	Telnet - System
XM	OSF/Motif
XN	OSF/Motif
XO	OSF/Motif
XP	OSF/Motif
XQ	OSF/Motif
XR	OSF/Motif
XU	OSF/Motif
XW	X Window System

Message ranges by volume

The IP Messages are distributed among volumes as follows:

Volume 1

EZA Messages

Volume 2

EZB and EZD Messages

Volume 3

EZY Messages

Volume 4

EZZ and SNM Messages

Terms and abbreviations used in this document

This section describes various terms that are used throughout all the IP Messages documents.

How the term “internet” is used in this document

In this document, an internet is a logical collection of networks supported by routers, gateways, bridges, hosts, and various layers of protocols, which permit the network to function as a large, virtual network.

Note: The term “internet” is used as a generic term for a TCP/IP network, and should not be confused with the Internet, which consists of large national backbone networks (such as MILNET, NSFnet, and CREN) and a myriad of regional and local campus networks worldwide.

How the document distinguishes “TCPIP” from “TCP/IP”

The abbreviation TCPIP is used to refer to the specific address space on which the Transmission Control Protocol/Internet Protocol product resides (for example, “TCPIP continues...”). The abbreviation TCP/IP refers to the product itself.

Other abbreviations and terms

The following abbreviations are also used in this document:

NCS Network Computing System, which is the Apollo implementation of remote procedure calls.

RPC Sun Microsystems' implementation of remote procedure calls.

SQL IBM Structured Query Language.

SQL/DS

IBM Structured Query Language/Data Systems Version 2 Release 2 or later.

ES/9000®

Enterprise System/9000 processor.

ES/9370™

Enterprise System/9370 processor.

Within the TCP/IP environment, you should also be familiar with the following terms:

Term	Description
------	-------------

Data set	The basic unit of data storage for MVS™. Unless otherwise specified, the use of this term indicates that the MVS host storing your local data set is your local host system.
-----------------	--

Local host	In an internet, any computer to which an end user or a functional unit is connected without the use of the internet.
-------------------	--

Remote host	In an internet, any host on the network that requires a physical link to interconnect with the network.
--------------------	---

User, local user	Either servers or clients (address spaces) on the local MVS system.
-------------------------	---

The variable *hlq* is used in this document to represent the high-level qualifier of a data set name. The default high-level qualifier for your installation was set when the product was installed and customized. Because TCP/IP for MVS uses both implicit and explicit data set allocation and provides you with methods of overriding the default, this variable can have many possible values. The *z/OS Communications Server: IP Configuration Reference* and the *z/OS Communications Server: New Function Summary* provide more details about data set names and the use of high-level qualifiers in this release. When in doubt, check with your system administrator to determine the value of *hlq* for a particular data set.

Clarification of notes

Information traditionally qualified as **Notes** is further qualified as follows:

Note Supplemental detail

Tip Offers shortcuts or alternative ways of performing an action; a hint

Guideline

Customary way to perform a procedure; stronger request than recommendation

Rule Something you must do; limitations on your actions

Restriction

Indicates certain conditions are not supported; limitations on a product or facility

Requirement

Dependencies, prerequisites

Result Indicates the outcome

Prerequisite and related information

z/OS Communications Server function is described in the z/OS Communications Server library. Descriptions of those documents are listed in “z/OS Communications Server information” on page 297, in the back of this document.

Required information

Before using this product, you should be familiar with TCP/IP, VTAM®, MVS, and UNIX® System Services.

Related information

This section contains subsections on:

- “Softcopy information”
- “Other documents”
- “Redbooks” on page xii
- “Where to find related information on the Internet” on page xiii
- “Accessing z/OS licensed documents on the Internet” on page xiv
- “Using LookAt to look up message explanations” on page xiv

Softcopy information

Softcopy publications are available in the following collections:

Titles	Order Number	Description
<i>z/OS V1R6 Collection</i>	SK3T-4269	This is the CD collection shipped with the z/OS product. It includes the libraries for z/OS V1R6, in both BookManager and PDF formats.
<i>z/OS Software Products Collection</i>	SK3T-4270	This CD includes, in both BookManager and PDF formats, the libraries of z/OS software products that run on z/OS but are not elements and features, as well as the <i>Getting Started with Parallel Sysplex®</i> bookshelf.
<i>z/OS V1R6 and Software Products DVD Collection</i>	SK3T-4271	This collection includes the libraries of z/OS (the element and feature libraries) and the libraries for z/OS software products in both BookManager and PDF format. This collection combines SK3T-4269 and SK3T-4270.
<i>z/OS Licensed Product Library</i>	SK3T-4307	This CD includes the licensed documents in both BookManager and PDF format.
<i>System Center Publication IBM S/390® Redbooks™ Collection</i>	SK2T-2177	This collection contains over 300 ITSO redbooks that apply to the S/390 platform and to host networking arranged into subject bookshelves.

Other documents

For information about z/OS products, refer to *z/OS Information Roadmap* (SA22-7500). The Roadmap describes what level of documents are supplied with each release of z/OS Communications Server, as well as describing each z/OS publication.

Relevant RFCs are listed in an appendix of the IP documents. Architectural specifications for the SNA protocol are listed in an appendix of the SNA documents.

The following table lists documents that may be helpful to readers.

Title	Number
<i>z/OS Integrated Security Services Firewall Technologies</i>	SC24-5922
<i>S/390: OSA-Express Customer's Guide and Reference</i>	SA22-7403
<i>z/OS JES2 Initialization and Tuning Guide</i>	SA22-7532
<i>z/OS MVS Diagnosis: Procedures</i>	GA22-7587
<i>z/OS MVS Diagnosis: Reference</i>	GA22-7588
<i>z/OS MVS Diagnosis: Tools and Service Aids</i>	GA22-7589
<i>z/OS Integrated Security Services LDAP Client Programming</i>	SC24-5924
<i>z/OS Integrated Security Services LDAP Server Administration and Use</i>	SC24-5923
<i>Understanding LDAP</i>	SG24-4986
<i>z/OS UNIX System Services Programming: Assembler Callable Services Reference</i>	SA22-7803
<i>z/OS UNIX System Services Command Reference</i>	SA22-7802
<i>z/OS UNIX System Services User's Guide</i>	SA22-7801
<i>z/OS UNIX System Services Planning</i>	GA22-7800
<i>z/OS MVS Using the Subsystem Interface</i>	SA22-7642
<i>z/OS C/C++ Run-Time Library Reference</i>	SA22-7821
<i>z/OS Program Directory</i>	GI10-0670
<i>DNS and BIND</i> , Fourth Edition, O'Reilly and Associates, 2001	ISBN 0-596-00158-4
<i>Routing in the Internet</i> , Christian Huitema (Prentice Hall PTR, 1995)	ISBN 0-13-132192-7
<i>sendmail</i> , Bryan Costales and Eric Allman, O'Reilly and Associates, 2002	ISBN 1-56592-839-3
<i>TCP/IP Tutorial and Technical Overview</i>	GG24-3376
<i>TCP/IP Illustrated, Volume I: The Protocols</i> , W. Richard Stevens, Addison-Wesley Publishing, 1994	ISBN 0-201-63346-9
<i>TCP/IP Illustrated, Volume II: The Implementation</i> , Gary R. Wright and W. Richard Stevens, Addison-Wesley Publishing, 1995	ISBN 0-201-63354-X
<i>TCP/IP Illustrated, Volume III</i> , W. Richard Stevens, Addison-Wesley Publishing, 1995	ISBN 0-201-63495-3
<i>z/OS Cryptographic Service System Secure Sockets Layer Programming</i>	SC24-5901
<i>SNA Formats</i>	GA27-3136

Redbooks

The following Redbooks may help you as you implement z/OS Communications Server.

Title	Number
<i>TCP/IP Tutorial and Technical Overview</i>	GG24-3376
<i>SNA and TCP/IP Integration</i>	SG24-5291
<i>IBM Communications Server for OS/390® V2R10 TCP/IP Implementation Guide: Volume 1: Configuration and Routing</i>	SG24-5227
<i>IBM Communications Server for OS/390 V2R10 TCP/IP Implementation Guide: Volume 2: UNIX Applications</i>	SG24-5228
<i>IBM Communications Server for OS/390 V2R7 TCP/IP Implementation Guide: Volume 3: MVS Applications</i>	SG24-5229
<i>Secureway Communications Server for OS/390 V2R8 TCP/IP: Guide to Enhancements</i>	SG24-5631

Title	Number
<i>TCP/IP in a Sysplex</i>	SG24-5235
<i>Managing OS/390 TCP/IP with SNMP</i>	SG24-5866
<i>Security in OS/390-based TCP/IP Networks</i>	SG24-5383
<i>IP Network Design Guide</i>	SG24-2580
<i>Migrating Subarea Networks to an IP Infrastructure</i>	SG24-5957
<i>IBM Communication Controller Migration Guide</i>	SG24-6298

Where to find related information on the Internet

z/OS

- <http://www.ibm.com/servers/eserver/zseries/zos/>

z/OS Internet Library

- <http://www.ibm.com/servers/eserver/zseries/zos/bkserv/>

IBM Communications Server product

- <http://www.software.ibm.com/network/commserver/>

IBM Communications Server product support

- <http://www.software.ibm.com/network/commserver/support/>

IBM Systems Center publications

- <http://www.redbooks.ibm.com/>

IBM Systems Center flashes

- <http://www-1.ibm.com/support/techdocs/atmastr.nsf>

RFCs

- <http://www.ietf.org/rfc.html>

Internet drafts

- <http://www.ietf.org/ID.html>

Information about Web addresses can also be found in information APAR II11334.

DNS web sites: For more information about DNS, see the following USENET news groups and mailing:

USENET news groups:

comp.protocols.dns.bind

For BIND mailing lists, see:

- <http://www.isc.org/ml-archives/>
 - BIND Users
 - Subscribe by sending mail to bind-users-request@isc.org.
 - Submit questions or answers to this forum by sending mail to bind-users@isc.org.
 - BIND 9 Users (Note: This list may not be maintained indefinitely.)
 - Subscribe by sending mail to bind9-users-request@isc.org.
 - Submit questions or answers to this forum by sending mail to bind9-users@isc.org.

Note: Any pointers in this publication to Web sites are provided for convenience only and do not in any manner serve as an endorsement of these Web sites.

Accessing z/OS licensed documents on the Internet

z/OS licensed documentation is available on the Internet in PDF format at the IBM Resource Link™ Web site at:

<http://www.ibm.com/servers/resourcelink>

Licensed documents are available only to customers with a z/OS license. Access to these documents requires an IBM Resource Link user ID and password, and a key code. With your z/OS order you received a Memo to Licensees, (GI10-0671), that includes this key code.

To obtain your IBM Resource Link user ID and password, log on to:

<http://www.ibm.com/servers/resourcelink>

To register for access to the z/OS licensed documents:

1. Sign in to Resource Link using your Resource Link user ID and password.
2. Select **User Profiles** located on the left-hand navigation bar.

Note: You cannot access the z/OS licensed documents unless you have registered for access to them and received an e-mail confirmation informing you that your request has been processed.

Printed licensed documents are not available from IBM.

You can use the PDF format on either **z/OS Licensed Product Library CD-ROM** or IBM Resource Link to print licensed documents.

Using LookAt to look up message explanations

LookAt is an online facility that lets you look up explanations for most of the IBM messages you encounter, as well as for some system abends and codes. Using LookAt to find information is faster than a conventional search because in most cases LookAt goes directly to the message explanation.

You can use LookAt from the following locations to find IBM message explanations for z/OS elements and features, z/VM®, VSE/ESA™, and Clusters for AIX® and Linux:

- The Internet. You can access IBM message explanations directly from the LookAt Web site at <http://www.ibm.com/eserver/zseries/zos/bkserv/lookat/>.
- Your z/OS TSO/E host system. You can install code on your z/OS or z/OS.e® systems to access IBM message explanations, using LookAt from a TSO/E command line (for example, TSO/E prompt, ISPF, or z/OS UNIX System Services running OMVS).
- Your Microsoft® Windows® workstation. You can install code to access IBM message explanations on the *z/OS Collection* (SK3T-4269), using LookAt from a Microsoft Windows DOS command line.
- Your wireless handheld device. You can use the LookAt Mobile Edition with a handheld device that has wireless access and an Internet browser (for example, Internet Explorer for Pocket PCs, Blazer, or Eudora for Palm OS, or Opera for Linux handheld devices). Link to the LookAt Mobile Edition from the LookAt Web site.

You can obtain code to install LookAt on your host system or Microsoft Windows workstation from a disk on your *z/OS Collection* (SK3T-4269), or from the LookAt

Web site (click **Download**, and select the platform, release, collection, and location that suit your needs). More information is available in the LOOKAT.ME files available during the download process.

How to send your comments

Your feedback is important in helping to provide the most accurate and high-quality information. If you have any comments about this document or any other z/OS Communications Server documentation:

- Go to the z/OS contact page at:
<http://www.ibm.com/servers/eserver/zseries/zos/webqs.html>
There you will find the feedback page where you can enter and submit your comments.
- Send your comments by e-mail to comsvrcf@us.ibm.com. Be sure to include the name of the document, the part number of the document, the version of z/OS Communications Server, and, if applicable, the specific location of the text you are commenting on (for example, a section number, a page number or a table number).

Summary of changes

Summary of changes for SC31-8784-04 z/OS Version 1 Release 6

This document contains information previously presented in SC31-8784-03, which supports z/OS Version 1 Release 5. The information in this document supports both IPv6 and IPv4. Unless explicitly noted, information describes IPv4 networking protocol. IPv6 support is qualified within the text.

New information

- EZD0011I
- EZD1163I–EZD1170I
- EZD1215I–EZD1227I

This document contains terminology, maintenance, and editorial changes. Technical changes or additions to the text and illustrations are indicated by a vertical line to the left of the change.

Starting with z/OS V1R4, you may notice changes in the style and structure of some content in this document—for example, headings that use uppercase for the first letter of initial words only, and procedures that have a different look and format. The changes are ongoing improvements to the consistency and retrievability of information in our documents.

Summary of changes for SC31-8784-03 z/OS Version 1 Release 5

This document contains information previously presented in SC31-8784-02, which supports z/OS Version 1 Release 4. The information in this document supports both IPv6 and IPv4. Unless explicitly noted, information describes IPv4 networking protocol. IPv6 support is qualified within the text.

New information

- EZD0001I
- EZD0002I
- EZD0003I
- EZD0004I
- EZD0005I
- EZD0006I
- EZD0007I
- EZD0008I
- EZD0009I
- EZD0010I
- EZD0101I
- EZD0800I
- EZD0801I

- EZD1161I
- EZD1162I

This document contains terminology, maintenance, and editorial changes. Technical changes or additions to the text and illustrations are indicated by a vertical line to the left of the change.

Starting with z/OS V1R4, you may notice changes in the style and structure of some content in this document—for example, headings that use uppercase for the first letter of initial words only, and procedures that have a different look and format. The changes are ongoing improvements to the consistency and retrievability of information in our documents.

**Summary of changes
for SC31-8784-02
z/OS Version 1 Release 4**

This document contains information previously presented in SC31-8784-01, which supports z/OS Version 1 Release 2. The information in this document supports both IPv6 and IPv4. Unless explicitly noted, information describes IPv4 networking protocol. IPv6 support is qualified within the text.

New information

- EZB0695I
- EZB0855I–EZB0857I
- EZB1100I

An appendix with z/OS product accessibility information has been added.

This document contains terminology, maintenance, and editorial changes. Technical changes or additions to the text and illustrations are indicated by a vertical line to the left of the change.

Starting with z/OS V1R4, you may notice changes in the style and structure of some content in this document (for example, headings that use uppercase for the first letter of initial words only, and procedures that have a different look and format). The changes are ongoing improvements to the consistency and retrievability of information in our documents.

Also starting with z/OS V1R4, you will notice this document is reorganized by message number to improve retrievability.

This document supports z/OS.e.

Chapter 1. EZB0xxxx messages

EZB0600I *message*

Explanation: This message is received by the server when the dynamic allocation of a data set or the MVS enqueueing (ENQ) function was unsuccessful.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: S99Error

EZB0603I *offset*

Explanation: This message indicates the offset at which the data string buffer is initialized.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DumpString

EZB0606I *buffer position*

Explanation: This message indicates the position at which the data string buffer is initialized.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DumpString

EZB0611E **Data Set name “arguments” invalid.**

Explanation: An argument declared in the *hlq*.LPD.CONFIG data set is incorrect. This message indicates the incorrect argument as declared by the user.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Correct the argument declared in the *hlq*.LPD.CONFIG data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for information about the syntax rules for the *hlq*.LPD.CONFIG data set.

Source Data Set: LPD

Procedure Name: ProcessOperands

EZB0612E Data Set “arguments” not found.

Explanation: The argument declared in the *hlq.LPD.CONFIG* data set was not found. This message indicates the argument as declared by the user.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Correct the argument declared in the *hlq.LPD.CONFIG* data set and restart the program. Refer to the *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: ProcessOperands

EZB0613E Data Set “dataset name” does not contain member “member name”.

Explanation: The data set name specified in the *data set name* parameter of the *hlq.SEZAINST(LPSPROC)* data set does not contain the indicated member.

System Action: LPD ends.

User or Operator Response: None.

System Programmer Response: Specify the correct data set name in the *data set name* parameter of the *hlq.SEZAINST(LPSPROC)* data set. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: ProcessOperands

EZB0614I IBM MVS LPD version *version level*

Explanation: This message indicates the current version and level of LPD for MVS.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: ProcessVersionOption

EZB0615I The option “option” is ambiguous. Use a longer abbreviation.

Explanation: An option specified in a parameter of the *hlq.SEZAINST(LPSPROC)* data set is incomplete. The option was abbreviated; however, the abbreviation is too short to distinguish between 2 correct options.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Specify the correct option in the *hlq.SEZAINST(LPSPROC)* data set and restart the procedure. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: ProcessOptions

EZB0616I The option “option” was not recognized.

Explanation: An option specified in a parameter of the *hlq.SEZAINST(LPSPROC)* data set is not recognized.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check for the correct option parameter of the *hlq*.SEZAINST(LPSPROC) data set and verify that the option specified is supported on your system. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: ProcessOptions

EZB0617I Use the TRACE, TYPE or VERSION options as needed.

Explanation: An incorrect option was specified in a parameter of the *hlq*.SEZAINST(LPSPROC) data set. This message indicates the correct options for this parameter. The following list provides the names and descriptions for these options:

VERSION	Displays the version number.
TYPE	Activates high-level trace facility in the LPD server. Significant events, such as the receipt of a job for printing, are recorded in the SYSOUT data set specified in your LPSPROC data set.
TRACE	Causes a detailed trace of activities within the LPD server to record in the SYSOUT data set specified in your LPSPROC data set. The detailed tracing can also be activated by the DEBUG statement in the configuration data set (<i>hlq</i> .LPD.CONFIG) and by the TRACE command of the SMSG interface.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Enter the correct option and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: ProcessOptions

EZB0619I Data set prefix not specified

Explanation: The prefix name of the TRACE *name* parameter of the *hlq*.SEZAINST(LPSPROC) data set was not specified. The parameter specifies the prefix of the configuration data set. The default is *hlq*. for the LPD.CONFIG data set.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check for the correct data set prefix in the PREFIX *name* parameter of the *hlq*.SEZAINST(LPSPROC) data set. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: ProcessOptions

EZB0620I Program error: Invalid option *rc*

Explanation: An error has occurred during processing of the configuration data set. This message indicates the return code received for this procedure.

System Action: The program ends abnormally.

User or Operator Response: None.

System Programmer Response: Check that the correct option parameters have been declared in the *hlq*.SEZAINST(LPSPROC) data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information. If the error persists, contact the IBM Software Support Center.

Source Data Set: LPD

Procedure Name: ProcessOptions

EZB0621I **LPD starting with port** *port number*

Explanation: This message indicates the port number on which the LPD connection was initiated.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: ProcessArguments

EZB0622E *message*

Explanation: The LPD is shutting down. Refer to the text of the message for further explanation.

System Action: LPD ends.

User or Operator Response: None.

System Programmer Response: Check the message content for an indication of configuration problems. Reinitiate TCPIP, if required.

Source Data Set: LPD

Procedure Name: Restore.

EZB0623I *errmsg (msgnum)*

Explanation: This message indicates the return code received by the Restore procedure. This procedure initiates a cleanup routine and ends TCPIP processing. This message is displayed with EZB0622I.

errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the **EZA** message whose text is displayed in *errmsg*. For more information about this message, see message *EZAmsgnum* in the *z/OS Communications Server: IP Messages Volume 1 (EZA)*.

System Action: LPD ends.

User or Operator Response: None.

System Programmer Response: Respond as indicated by the message *EZAmsgnum*.

Source Data Set: LPD

Procedure Name: Restore

EZB0625E **Out of storage for connections!**

Explanation: There is not enough storage allocated for a connection to complete.

System Action: The program ends abnormally.

User or Operator Response: None.

System Programmer Response: Allocate more storage for connections.

Source Data Set: LPD

Procedure Name: AllocConnection

EZB0626I **Allocated ConnectionBlock at** *address*

Explanation: A connection block was allocated at the indicated IP address. A connection block is used to build a connection record.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: AllocConnection

EZB0627I **Passive open on port** *port number*

Explanation: A passive connection opening was established on the indicated port number.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: GetNewConnection

EZB0628I **Allocated PrinterBlock at** *address*

Explanation: A printer block was allocated at the indicated IP address.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: AllocPrinter

EZB0629I *printer name* **added.**

Explanation: This message indicates the name of the printer that was allocated for the LPD server.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: AllocPrinter

EZB0630I **Spool allocate RC** *rc*, **Class** *default class*, **DEST** *NJEdest*, **ID** *NJE ID*, **OUTPUT** *default spool*

EZB0631I *string*

Explanation: The Network Job Entry system (NJE) is being used as the remote printing application for LPD. This message indicates the SYSOUT class, the name of the NJE node, the device user ID, the output name, and the value of the data buffer.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: CompletePrinter

EZB0632I *string*

Explanation: This message indicates the size of the data buffer.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: CompletePrinter

EZB0633I **Use DEST and IDENTIFIER for MVS.**

Explanation: The DEST and IDENTIFIER parameters should be used for this MVS system. The DEST option sets the destination node. The default is the local node. The IDENTIFIER option specifies the device user ID. The default is SYSTEM.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: CompletePrinter

EZB0635I **Spool allocate RC** *spool rc*, **Class** *default class*, **OUTPUT** *default spool*

EZB0636I *string*

Explanation: The Network Job Entry system (NJE) is being used as the remote printing application for LPD. This message indicates the SYSOUT class, the name of the NJE node, the device user ID, the output name, and the value of the data buffer.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: CompletePrinter

EZB0637I *string*

Explanation: This message indicates the size of the data buffer.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: CompletePrinter

EZB0640I **The errors reported above prevent startup of** *dataset name*

Explanation: The indicated data set name could not be opened. See the previous messages, which provide more specific information about this error.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: This message is displayed with more specific error messages. Respond as indicated by the previous messages.

Source Data Set: LPD

Procedure Name: CompletePrinter

EZB0641I Service printer name defined with address address

Explanation: This message indicates the service name and the address space that were declared in the *name* parameter of the SERVICE statement of the *hlq*.LPD.CONFIG data set. The *name* parameter specifies a service where connections are accepted and acknowledged.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: CompletePrinter

EZB0642I Use a class which is one letter.

Explanation: The CLASS=*class* parameter of the service defined in the SERVICE statement of the *hlq*.LPD.CONFIG data set is incorrect. This parameter specifies the SYSOUT class. The default is A for printers and B for punches.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Specify the correct class in the CLASS=*class* parameter and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: SetPrinterRSCSDefault

EZB0643I Use a class which is one letter or digit.

Explanation: The CLASS=*class* parameter of the service defined in the SERVICE statement of the *hlq*.LPD.CONFIG data set is incorrect. This parameter specifies the SYSOUT class. The default is A for printers and B for punches.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Specify the correct class in the CLASS=*class* parameter and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: SetPrinterRSCSDefault

EZB0644I Use a number after the PRIORITY keyword.

Explanation: An unexpected character was received after the PRIORITY=*priority* parameter of the service defined in the SERVICE statement of the *hlq*.LPD.CONFIG data set. The PRIORITY parameter specifies the transmission priority. The default is 50.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Correct the entry declared in the PRIORITY=*priority* parameter and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: SetPrinterRSCSDefault

EZB0645I SPOOL only valid for PRINTER or PUNCH

Explanation: SPOOL is valid for the PRINTER or PUNCH services only. These services are specified in the SERVICE statement of the *hlq*.LPD.CONFIG data set.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Correct the parameters declared for the services specified in the SERVICE statement. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: SetPrinterRSCSDefault

EZB0646I Could not spool dataset name device. Return code rc

Explanation: The attempt to spool to the data set defined in the *name* parameter of the SERVICE statement was unsuccessful. This message indicates the return code received after this procedure.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check for the correct data set name specified in the *name* parameter of the SERVICE statement in the *hlq*.LPD.CONFIG data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: SetPrinterRSCSDefault

EZB0647I Only use CLASS, DEST, IDENTIFIER, OTHERS, PRIORITY, SPOOL or TAG as qualifiers.

Explanation: An incorrect parameter was specified in the RSCS statement of the *hlq*.LPD.CONFIG data set. The following list provides a description for these parameters:

CLASS	The SYSOUT class. The default is A for printers and B for punches.
DEST	Specifies the destination node ID. The default is the node on which LPSERVE is running.
IDENTIFIER	Specifies the device user ID. The default is SYSTEM.
OTHERS	The option is ignored by MVS LPSERVE.
PRIORITY	Specifies the transmission priority. The default is 50.
SPOOL	Supplies the operands for the CP SPOOL command that will be used on the virtual printer or punch that is defined for each new job.
TAG	Supplies the operands for the CP TAG command that will be used on the virtual printer or punch that is defined for each new job.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Specify the correct parameter in the RSCS statement of the *hlq*.LPD.CONFIG data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: SetPrinterRSCSDefault

EZB0648I Use a class which is one letter.

Explanation: The CLASS option of the NJE parameter for the SERVICE statement in the *hlq*.LPD.CONFIG data set is not correct. The default is A for printers and B for punches.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Specify the correct CLASS option for the NJE parameter in the SERVICE statement and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: SetPrinterRSCSDefault

EZB0649I Use a class which is one letter or digit.

Explanation: The CLASS option of the NJE parameter for the SERVICE statement in the *hlq*.LPD.CONFIG data set is not correct. The default is A for printers and B for punches.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Specify the correct CLASS option for the NJE parameter in the SERVICE statement and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: SetPrinterRSCSDefault

EZB0650I SPOOL only valid for PRINTER or PUNCH

Explanation: SPOOL is valid for the PRINTER or PUNCH services only. These services are specified in the SERVICE statement of the *hlq*.LPD.CONFIG data set.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Correct the parameters declared for the services specified in the SERVICE statement. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: SetPrinterRSCSDefault

EZB0651I Could not spool dataset name device. Return code rc

Explanation: The attempt to spool to the data set defined in the *name* parameter of the SERVICE statement was unsuccessful. This message indicates the return code received after this procedure.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check for the correct data set name specified in the *name* parameter of the SERVICE statement in the *hlq*.LPD.CONFIG data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: SetPrinterRSCSDefault

EZB0652I Only use CLASS, DEST, SPOOL or TAG as qualifiers.

Explanation: An incorrect parameter was specified in the LOCAL statement of the *hlq*.LPD.CONFIG data set. The following list provides a description for these parameters:

CLASS	The SYSOUT class. The default is A for printers and B for punches.
DEST	Specifies the destination node ID. The default is the node on which LPSERVE is running.
SPOOL	Supplies the operands for the CP SPOOL command that will be used on the virtual printer or punch that is defined for each new job.

EZB0653I • EZB0656I

TAG Supplies the operands for the CP TAG command that will be used on the virtual printer or punch that is defined for each new job.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Specify the correct parameter in the RSCS statement of the *hlq*.LPD.CONFIG data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: SetPrinterRSCSDefault

EZB0653I The keyword “keyword” is not a keyword.

Explanation: An incorrect parameter was declared in the SMTP statement of the *hlq*.LPD.CONFIG data set. The SMTP statement specifies the SMTP server name, CLASS, and DEST options.

System Action: LPD continues. Failed job will not work unless the SMTP statement is defined and LPD is restarted.

User or Operator Response: None.

System Programmer Response: Correct the parameter declared in the SMTP statement of the *hlq*.LPD.CONFIG data set, and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: ProcessSMTPOptions

EZB0654I The keyword “keyword” is too short. Use a longer abbreviation.

Explanation: The indicated option was abbreviated. However, the abbreviation is too short to distinguish between 2 correct options.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Specify the correct option and restart the procedure. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: ProcessLocalandRCSCOptions

EZB0655I Use the SMTP service machine name after “SMTP”.

Explanation: An incorrect parameter was specified in the SMTP statement of the *hlq*.LPD.CONFIG data set. This statement specifies the SMTP server name, CLASS, and DEST options.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Specify the correct name in the *server_name* parameter of the SMTP statement of the *hlq*.LPD.CONFIG data set and restart the program. If this parameter is omitted, the default is SMTP. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: ProcessSMTPOptions

EZB0656I The keyword “keyword” is not a keyword.

Explanation: An incorrect parameter was declared in the SMTP statement of the *hlq*.LPD.CONFIG data set. The SMTP statement specifies the SMTP server name, CLASS, and DEST options.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Correct the parameter declared in the SMTP statement of the *hlq*.LPD.CONFIG data set, and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: ProcessSMTPOptions

EZB0657I The keyword “keyword” is too short. Use a longer abbreviation.

Explanation: A parameter of the SMTP statement in the *hlq*.LPD.CONFIG data set is incorrect. This parameter was abbreviated; however, the abbreviation is too short to distinguish between two correct parameters.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Specify the correct parameter in the SMTP statement in the *hlq*.LPD.CONFIG data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: ProcessSMTPOptions

EZB0658I Use a class which is one letter.

Explanation: The class specified in the CLASS=*class* parameter of the SMTP statement of the *hlq*.LPD.CONFIG data set is incorrect. This parameter specifies the SYSOUT class. The default is A for printers and B for punches. Valid values for this parameter are A through Z.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Correct the class declared in the CLASS=*class* parameter of the SMTP statement in the *hlq*.LPD.CONFIG data set and restart the program.

Source Data Set: LPD

Procedure Name: ProcessSMTPOptions

EZB0659I Use a class which is one letter or digit.

Explanation: The class specified in the CLASS=*class* parameter of the SMTP statement of the *hlq*.LPD.CONFIG data set is incorrect. This parameter specifies the SYSOUT class. The default is A for printers and B for punches. Valid values for this parameter are A through Z and 0 through 9.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Correct the class declared in the CLASS=*class* parameter of the SMTP statement in the *hlq*.LPD.CONFIG data set and restart the program.

Source Data Set: LPD

Procedure Name: ProcessSMTPOptions

EZB0660I Only use CLASS, DEST, IDENTIFIER or OTHERS as qualifiers.

Explanation: A parameter of the SMTP statement in the *hlq*.LPD.CONFIG data set is incorrect. This message indicates the correct parameters for this statement.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Specify the correct parameter for the SMTP statement of the *hlq*.LPD.CONFIG data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: ProcessSMTPOptions

EZB0665I Cannot open configuration file.

Explanation: The configuration data set could not be opened.

System Action: LPD ends.

User or Operator Response: None.

System Programmer Response: Check that the name of the configuration data set was specified correctly. Check that the data set is available to the server. Check that the server has authority to access the data set.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0666I Use option after you have defined a SERVICE.

Explanation: A parameter of the SERVICE statement in the *hlq*.LPD.CONFIG data set was not found. The SERVICE statement specifies a service for which connections are accepted and acknowledged. The following provides a description of the valid parameters for the SERVICE statement:

name The service name must be one to eight characters in length. Only characters permitted in MVS data set names are valid. This value is case-sensitive.

PRINTER Specifies that the service is to a printer.

PUNCH Specifies that the service is to a punch device.

NONE Specifies that the service is not currently in use.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Specify the correct parameter for in the SERVICE statement of the *hlq*.LPD.CONFIG data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0667I The keyword "keyword" is ambiguous. Use a longer abbreviation.

Explanation: A statement of the *hlq*.LPD.CONFIG data set is incorrect. An abbreviation was used for this statement; however, the abbreviation is too short to distinguish between two correct statements.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Use the correct statement in the *hlq*.LPD.CONFIG data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0668I The keyword "keyword" was not recognized.

Explanation: A statement in the *hlq*.LPD.CONFIG data set is incorrect.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Correct the statement in the *hlq*.LPD.CONFIG data set and restart the job. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0669I **Use an integer after “DISK or UNIT”.**

Explanation: A number is expected after the DISK or UNIT statements of the *hlq.LPD.CONFIG* data set.

System Action: LPD halts.

User or Operator Response: None.

System Programmer Response: Specify the correct value in the parameters of the DISK or UNIT statement of the *hlq.LPD.CONFIG* data set and restart the program.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0670I **Use either START or END after “EXIT”.**

Explanation: Specify the START or END parameter with the EXIT statement of the *hlq.LPD.CONFIG* data set. The following provides a description for these parameters:

Parameter	Description
START	Specifies that the program is invoked after allocating and opening the output data set, but before anything is written to the data set.
END	Specifies that the program is invoked just before closing the output data set.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Use the correct parameter with the EXIT statement of the *hlq.LPD.CONFIG* data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0671I **Use either START or END after “EXIT”.**

Explanation: An incorrect parameter was specified in the EXIT statement of the *hlq.LPD.CONFIG* data set. The following provides a description for these parameters:

Parameter	Description
START	Specifies that the program is invoked after allocating and opening the output data set, but before anything is written to the data set.
END	Specifies that the program is invoked just before closing the output data set.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Use the correct parameter with the EXIT statement of the *hlq.LPD.CONFIG* data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0672I **Use name after type of EXIT.**

Explanation: The program name specified in the *program* parameter of the EXIT statement in the *hlq.LPD.CONFIG* data set is incorrect or not found. This parameter specifies the name of the program to be invoked after allocating and opening, but before closing, an output data set.

System Action: LPD continues.

EZB0673I • EZB0675I

User or Operator Response: None.

System Programmer Response: Specify the correct program name in the *program* parameter of the EXIT statement in the *hlq*.LPD.CONFIG data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0673I Could not load EXIT *program name*.

Explanation: The program name specified in the *program* parameter of the EXIT statement in the *hlq*.LPD.CONFIG data set could not be accessed. This parameter specifies the name of the program to be invoked after allocating and opening, but before closing, an output data set. This message is displayed when the START parameter was specified with the EXIT statement.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check that the correct program name was specified in the *program* parameter of the EXIT statement in the *hlq*.LPD.CONFIG data set and restart the program.

The library containing the program should be in the system's link list, (LNKLSTxx) or a STEPLIB definition can be used if the library is APF authorized.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0674I Could not load EXIT *program name*.

Explanation: The program name specified in the *program* parameter of the EXIT statement in the *hlq*.LPD.CONFIG data set could not be accessed. This parameter specifies the name of the program to be invoked after allocating and opening, but before closing, an output data set. This message is displayed when the END parameter was specified with the EXIT statement.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check that the correct program name was specified in the *program* parameter of the EXIT statement in the *hlq*.LPD.CONFIG data set and restart the program.

The library containing the program should be in the system's link list, (LNKLSTxx) or a STEPLIB definition can be used if the library is APF authorized.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0675I Use either MAIL or DISCARD after "FAILEDJOB".

Explanation: A parameter specified in the FAILEDJOB statement of the *hlq*.LPD.CONFIG data is incorrect or not found. This statement specifies whether a notice of unsuccessful jobs should be mailed to users or a job is discarded without notification. The following provides a description for these parameters:

MAIL Specifies that notices of unsuccessful jobs are mailed to users.

DISCARD Specifies that unsuccessful jobs are discarded without notice.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Specify the correct parameter in the FAILEDJOB statement of the *hlq*.LPD.CONFIG data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0676I Use an integer after "LINESIZE".

Explanation: The value specified in the *length* parameter of the LINESIZE statement in the *hlq.LPD.CONFIG* data set was not found. This parameter specifies the number of characters in a line on a page. Lines longer than this number are truncated. The default is 132.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Correct the value declared in the *length* parameter of the LINESIZE statement in the *hlq.LPD.CONFIG* data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0677I Use an integer after "LINESIZE"

Explanation: The value specified in the *length* parameter of the LINESIZE statement in the *hlq.LPD.CONFIG* data set is incorrect. This parameter specifies the number of characters in a line on a page. Lines longer than this number are truncated. The default is 132.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Correct the value declared in the *length* parameter of the LINESIZE statement in the *hlq.LPD.CONFIG* data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0678I Use only one of LOCAL, NJE, RSCS, and REMOTE.

Explanation: The destination type parameter for the SERVICE statement in the *hlq.LPD.CONFIG* data set is not correct. The following list provides a description of the valid destination types for this parameter:

LOCAL	Specifies that the data sets are written to the local MVS printer or punch.
NJE	Specifies that the data sets are delivered to the Network Job Entry (NJE) system.
RSCS	Specifies that the data sets are delivered to the Remote Spooling Communications Subsystem (RSCS).
REMOTE	Specifies that the data sets are forwarded to a remote printer.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Correct the destination type parameter for the SERVICE statement in the *hlq.LPD.CONFIG* data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0679I Allocated ObeyBlock at *address*

Explanation: The OBEY statement for the *hlq*.LPD.CONFIG data set has been received. This statement specifies user IDs authorized to use the SMSG interface provided with LPD. This message indicates that an address has been allocated an ObeyBlock, or authorization to use the SMSG interface. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0680I Use an integer after "PAGESIZE".

Explanation: The value specified in the *lines* parameter of the PAGESIZE statement in the *hlq*.LPD.CONFIG data set was not found. This parameter specifies the number of lines on a page. The default is 60.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Specify the correct value in the *lines* parameter of the PAGESIZE statement in the *hlq*.LPD.CONFIG data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0681I Use an integer after "PAGESIZE".

Explanation: The value specified in the *lines* parameter of the PAGESIZE statement in the *hlq*.LPD.CONFIG data set is incorrect. This parameter specifies the number of lines on a page. The default is 60.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Specify the correct value in the *lines* parameter of the PAGESIZE statement in the *hlq*.LPD.CONFIG data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0682I Use a *printer@hostname* after "REMOTE".

Explanation: The destination specified in the *printer@hostname* parameter of the REMOTE statement in the *hlq*.LPD.CONFIG data set was not found. This parameter indicates the destination printer at a specified IP host. This can be an IP name or an IP address.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check for the correct entry in the *printer@hostname* parameter of the REMOTE statement in the *hlq*.LPD.CONFIG data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0683I Use a printer@hostname after "REMOTE".

Explanation: The destination specified in the *printer@hostname* parameter of the REMOTE statement in the *hlq.LPD.CONFIG* data set is incorrect. This parameter indicates the destination printer at a specified IP host. This can be an IP name or an IP address.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check for the correct entry in the *printer@hostname* parameter of the REMOTE statement in the *hlq.LPD.CONFIG* data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0684I Cannot reach destination

Explanation: The destination specified in the *printer@hostname* parameter of the REMOTE statement in the *hlq.LPD.CONFIG* data set could not be reached. This parameter indicates the destination printer at a specified IP host. This can be an IP name or an IP address.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check for the correct destination in the *printer@hostname* parameter of the REMOTE statement, and for the correct service name specified in the *name* parameter of the SERVICE statement in the *hlq.LPD.CONFIG* data set, and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0685I Use only one of LOCAL, NJE, RSCS, and REMOTE.

Explanation: The destination type parameter specified for the SERVICE statement in the *hlq.LPD.CONFIG* data set is not correct. The following list provides a description of the valid service names for this parameter:

LOCAL	Specifies that the data sets are written to the local MVS printer or punch.
NJE	Specifies that the data sets are delivered to the Network Job Entry (NJE) system.
RSCS	Specifies that the data sets are delivered to the Remote Spooling Communications Subsystem (RSCS).
REMOTE	Specifies that the data sets are forwarded to a remote printer.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Correct the destination type parameter declared for the SERVICE statement in the *hlq.LPD.CONFIG* data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0686I Host "host" resolved to address. Printer name is "printer name".

Explanation: This message indicates the host name, IP address, and the printer name to which the destination address was resolved. The destination address is specified in the *printer@hostname* parameter of the REMOTE statement in the *hlq.LPD.CONFIG* data set.

System Action: LPD continues.

EZB0687I • EZB0689I

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0687I Use only one of LOCAL, NJE, RSCS, and REMOTE.

Explanation: The service name specified in the *name* parameter of the SERVICE statement in the *hlq*.LPD.CONFIG data set is incorrect. The following list provides a description of the valid service names for this parameter:

LOCAL

Specifies that the data sets are written to the local MVS printer or punch.

NJE

Specifies that the data sets are delivered to the Network Job Entry (NJE) system.

RSCS

Specifies that the data sets are delivered to the Remote Spooling Communications Subsystem (RSCS).

REMOTE

Specifies that the data sets are forwarded to a remote printer.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Correct the service name declared in the *name* parameter of the SERVICE parameter in the *hlq*.LPD.CONFIG data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0688I Use a printer name and type after "SERVICE".

Explanation: The service name specified in the *name* parameter of the SERVICE statement in the *hlq*.LPD.CONFIG data set was not found. The SERVICE statement specifies the service name for which connections are accepted and acknowledged.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check for the correct name specified in the *name* parameter of the SERVICE statement and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0689I The service "service name" has been described more than once.

Explanation: The service name specified in the *name* parameter of the SERVICE statement in the *hlq*.LPD.CONFIG data set is a duplicate. This statement specifies a service for which connections are accepted and acknowledged.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check for the correct name specified in the *name* parameter of the SERVICE statement and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0690I Use a printer name and type after “SERVICE”.

Explanation: The service name specified in the *name* parameter of the SERVICE statement in the *hlq*.LPD.CONFIG data set was not found. This statement specifies a service for which connections are accepted and acknowledged.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check for the correct name specified in the *name* parameter of the SERVICE statement and restart the program.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0691I Use “PRINTER” or “PUNCH” as a SERVICE type.

Explanation: The parameter specified in the SERVICE statement of the *hlq*.LPD.CONFIG data set is incorrect. This statement indicates a service for which connections are accepted and acknowledged. The following provides a description of the valid parameters for this statement:

name The service name must be one to eight characters in length. Only characters permitted in MVS data set names are valid. This value is case-sensitive.

PRINTER Specifies that the service is to a printer.

PUNCH Specifies that the service is to a punch device.

NONE Specifies that the service is not currently in use.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check that the correct parameter is specified in the SERVICE statement and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: PreparePrinters.

EZB0692I Use a Volume Serial after “VOLUME”.

Explanation: An incorrect value was declared in the VOLUME statement of the *hlq*.LPD.CONFIG data set.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check for the correct entry specified in the VOLUME statement of the *hlq*.LPD.CONFIG data set and restart the program. The correct length of this parameter is six characters.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0693I Use a Table Name after “TRANSLATETABLE”.

Explanation: The name specified in the *name* parameter of the TRANSLATETABLE statement in the *hlq*.LPD.CONFIG data set was not found. This parameter specifies the name of the translation table. If a DBCS conversion parameter is specified, *name* is used to determine which DBCS table to load.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check that the correct name was specified in the *name* parameter of the TRANSLATETABLE statement and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

EZB0694I • EZB0696I

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0694I Could not load Translate Table.

Explanation: The translation table name specified in the *name* parameter of the TRANSLATETABLE statement in the *hlq*.LPD.CONFIG data set could not be loaded. This statement specifies the translation table to be used by the client and is found in the *name*.TCPXLBIN data set.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check that the correct translate table name is specified in the *name* parameter of the TRANSLATETABLE statement, check that the *name*.TCPXLBIN data set is available to the server, and restart the program. If a DBCS conversion parameter is specified in the SMTP.SMTP.CONFIG data set, *name* is used to determine which DBCS translation table to load.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0695I Could not load translate table with name - *name* for printer *printer_name*

Explanation: The Line Printer Daemon (LPD) attempted to load an SBCS translation table corresponding to the name provided by the TRANSLATETABLE or XLATETABLE statement in the *hlq*.LPD.CONFIG data set. This statement specifies the translation table to be used by the remote client for SBCS ASCII to EBCDIC translations.

All data sets in the search order hierarchy for the required translate table data set either do not exist, or do not contain data in the required format for SBCS binary translate tables.

name is an input string that becomes part of the translation table data set name (for example, *hlq.name*.TCPXBLIN).

printer_name is the name of the printer service.

System Action: LPD continues, however the printer service is unavailable.

User or Operator Response: Notify the system programmer.

System Programmer Response: Configure a valid SBCS binary translate table data set in the search order hierarchy for the required SBCS translation table. See the *z/OS Communications Server: IP Configuration Reference* for more information about using translation tables including search order hierarchy and customization.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0696I Program error: Invalid option *option*

Explanation: An incorrect statement was specified in the *hlq*.LPD.CONFIG data set. This message indicates the statement as declared in this data set.

System Action: TCPIP ends.

User or Operator Response: None.

System Programmer Response: Restart TCPIP, correct the statement specified in the *hlq*.LPD.CONFIG data set and restart the program. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0697I ...End of Printer chain...

Explanation: The *hlq*.LPD.CONFIG data set statements have been processed to build the control tables representing the supported printers.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0698I InitEmulation failed

Explanation: The procedure InitEmulation, which starts the process to allow commands to run in non-EC mode machines, was not successful.

System Action: TCPIP ends.

User or Operator Response: None.

System Programmer Response: Run a 3270 type terminal emulator or use a 3270 type display station and restart TCPIP.

Source Data Set: LPD

Procedure Name: PrepareTCP

EZB0699I Starting TCP/IP service connection

Explanation: A connection was initiated to the TCPIP services.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: PrepareTCP

EZB0700I BeginTcpiip: errmsg (msgnum)

Explanation: The procedure BeginTcpiip, which informs the TCPIP address space that you want to start using its services, was unsuccessful.

errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the **EZA** message whose text is displayed in *errmsg*. For more information about this message, see message EZAmsgnum in the *z/OS Communications Server: IP Messages Volume 1 (EZA)*.

System Action: TCPIP ends.

User or Operator Response: None.

System Programmer Response: Respond as indicated by the message EZAmsgnum.

Source Data Set: LPD

Procedure Name: PrepareTCP

EZB0701I **TCP/IP turned on.**

EZB0702I **Host** “*host ID*” **Domain** “*domain ID*” **TCPIP Service Machine** “*service ID*”

Explanation: TCPIP services have been initiated for the indicated host, domain, and printer IDs.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: PrepareTCP

EZB0703I **FSEND failed** *errmsg (msgnum)*

Explanation: The procedure FSEND, which sends data on a TCP connection, was unsuccessful.

errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the **EZA** message whose text is displayed in *errmsg*. For more information about this message, see message EZA*msgnum* in the *z/OS Communications Server: IP Messages Volume 1 (EZA)*.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Respond as indicated by the message EZA*msgnum*.

Source Data Set: LPD

Procedure Name: SendACK

EZB0704I **Abort issued for connection** *connection number*

Explanation: The procedure DoAbortConnection, which shuts down a specific connection, was initiated. This message indicates the connection number for which the TcpAbort procedure was started.

System Action: The TCP connection ends.

User or Operator Response: None.

System Programmer Response: Reinitiate the connection if required.

Source Data Set: LPD

Procedure Name: DoAbortConnection

EZB0705I *date time*

Explanation: This message indicates the current date and time.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: PrintTimeStamp

EZB0706I Terminating connection *connection message*

Explanation: Because of an incorrect return code received from TCP, the connection ends. This message indicates the connection number and the reason for the termination. This message is displayed with EZB0705I.

System Action: This connection ends.

User or Operator Response: None.

System Programmer Response: Reinitiate the connection if required.

Source Data Set: LPD

Procedure Name: TerminateConnection

EZB0707I Adding "message line" to message.

Explanation: The indicated message line was added to the message input buffer.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DoSendQueueList

EZB0708I FSend of response sent

Explanation: The procedure FSend, which sends data on a TCP connection, was initiated.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DoSendQueueList

EZB0710I New command *command code (no operands)*

Explanation: This message indicates the command code that was received from the client.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DoNewCommand

EZB0711I New command *command code data "parameter".*

Explanation: This message indicates the command code and the additional operands that have been received from the client.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DoNewCommand

EZB0712I **Command rejected. Printer “printer” not recognized.**

Explanation: A command from the client was received that contains an operand with an unrecognized printer ID.

System Action: The LPD to LPD connection ends.

User or Operator Response: Reinitiate the connection, check for the correct printer ID, and resubmit the command. Refer to *z/OS Communications Server: IP User's Guide and Commands* for more information.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPD

Procedure Name: DoNewCommand

EZB0713I **Printer “print” not found**

Explanation: The printer specified in the remote printing command is incorrect. This message indicates the printer ID as declared by the user.

System Action: LPD continues.

User or Operator Response: Check for the correct printer ID and resubmit the command. See the *z/OS Communications Server: IP User's Guide and Commands* for more information.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPD

Procedure Name: DoNewCommand

EZB0716I **Job job ID comment printer name site**

Explanation: This message indicates the job ID that was placed onto the queue for the designated printer.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: PrintJobLogLine

EZB0717E **Could not erase dataset name RC=rc**

Explanation: The job submitted by the client could not be erased from the queue after processing. This message indicates the data set name and return code that was passed.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check for error messages in the LPD log and trace.

Source Data Set: LPD

Procedure Name: EraseFile

EZB0718E **Could not open “queue name QUEUE”.**

Explanation: The specified queue could not be opened.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: This message should be preceded by more specific messages. Correct the errors indicated by the preceding messages.

Source Data Set: LPD

Procedure Name: SavePrinterQueue

EZB0719I Allocated JobBlock at *address*

Explanation: The print job block received from the client was allocated at the indicated IP address.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: AllocJob

EZB0720I Site file "*name SITE*" record 1 unreadable.

Explanation: The first record of the HOSTS.SITEINFO data set could not be read.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check that the HOSTS.SITEINFO data set was generated and installed, and that the records have been entered using the correct record format. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: LoadSite

EZB0721I Site file "*name SITE*" record 2 unreadable.

Explanation: The second record of the HOSTS.SITEINFO data set could not be read.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check that the HOSTS.SITEINFO data set was generated and installed and that the records have been entered using the correct record format.

Source Data Set: LPD

Procedure Name: LoadSite

EZB0722I Could not open "*name SITE*".

Explanation: The HOSTS.SITEINFO data set could not be opened. When making changes to the HOSTS.LOCAL data sets, you must generate and install new HOSTS.SITEINFO and HOSTS.ADDRINFO data sets. Use the MAKESITE statement as either a TSO command or a batch job to generate the new data sets.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check that the HOSTS.SITEINFO data set was generated and installed and that the records have been entered using the correct record format.

Source Data Set: LPD

Procedure Name: LoadSite

EZB0723I Allocated StepBlock at *address*

Explanation: A StepBlock was allocated at the indicated IP address.

System Action: LPD continues.

User or Operator Response: None.

EZB0724E • EZB0727I

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: AllocStep

EZB0724E Could not open control file "*site name job ID printer name*". **Job abandoned.**

Explanation: The control data set specified in the LPR command received by the server could not be opened. This message indicates the site, the job ID, and the remote printer ID.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check that the correct data set name was specified in the LPR command and that the data set is available to the server and reissue the LPR command.

Source Data Set: LPD

Procedure Name: ProcessControlFile

EZB0725I Reloading job *job*.

Explanation: The job ID, specified in the remote printing command, is reloaded.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: LoadJob

EZB0726I Could not open "*site job ID*".

Explanation: The job ID specified in the LPR command received by the server could not be processed because the indicated site could not be opened. This message indicates the site and the job ID specified.

System Action: LPD continues.

User or Operator Response: Make sure the correct destination ID is used for its corresponding parameter and the correct job name is declared in the JOB parameter of the LPR command. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: LoadJob

EZB0727I Job *job ID* abandoned. **Job file too short.**

Explanation: The procedure readln, which reads the data set specified by the client on the LPR command, returned a nonzero return code.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check for the correct job ID specified in the Job *jobname* parameter of the LPR command and submit the command again. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: LoadJob

EZB0728I Job *job ID* abandoned. Job file too short.

Explanation: The procedure readln, which reads the data set specified by the client on the LPR command, returned a nonzero return code.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check for the correct job ID specified in the Job *jobname* parameter of the LPR command and submit the command again. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: LoadJob

EZB0729I Job *job ID* abandoned. Job file too short.

Explanation: The procedure readln, which reads the data set specified by the client on the LPR command, returned a nonzero return code.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check for the correct job ID specified in the Job *jobname* parameter of the LPR command and submit the command again. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: LoadJob

EZB0730I Job *job ID* abandoned. Not enough storage.

Explanation: The job ID specified in the *jobname* parameter of the LPR command received by the server could not be completed because of insufficient storage.

System Action: LPD continues.

User or Operator Response: Inform the system programmer about this message.

System Programmer Response: Check for the storage requirements needed to process the LPR command, allocate more storage, and reissue the command.

Source Data Set: LPD

Procedure Name: LoadJob

EZB0731I Work Queue start

Explanation: The work queue was initiated.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: PrintWorkQueue

EZB0732I *job number job ID*

Explanation: This message indicates the print job received from the client was placed onto the queue for the designated service.

System Action: LPD continues.

EZB0733I • EZB0736I

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: PrintWorkQueue

EZB0733I Work Queue end

Explanation: The work queue is empty. The print server returns to a passive wait state awaiting the next print request.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: PrintWorkQueue

EZB0734I Job *jobnumber*: added to work queue

Explanation: TCPIP displays this message while tracing is on. The specified job number is displayed as it is added to the list of queued jobs.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: QueueJobWork

EZB0735I *stepblock_address datasize action_code dataset name*

Explanation: TCPIP issues this message when tracing is on. The StepBlock address, data size, action code and data set name of the current job are displayed.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DumpStepChain

EZB0736I Could not open *name* QUEUE

Explanation: LPD could not open the specified data set that is queued for printing.

System Action: LPD continues.

User or Operator Response: Check the syntax of the specified data set name or verify that you have authority to print through the system programmer.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPD

Procedure Name: LoadPrinterQueue

EZB0737I Reloading *dataset name* queue

Explanation: LPD issues this message while tracing is on. While attempting to queue the previous data set name, an error was detected. The data set name is reloaded into the list of queued data sets.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: LoadPrinterQueue

EZB0738I Ignoring job *number* from site *name* because there is no SITE file.

Explanation: TCP/IP was unable to locate the site for the specified job number.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Make sure the site was defined using the MAKESITE command.

Source Data Set: LPD

Procedure Name: LoadPrinterQueue

EZB0739I Validating user *user*

Explanation: This message occurs while tracing is on. The userid of the submitter of the print job is validated.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: LoadPrinterQueue

EZB0740I Validation failed. RC *rc*

Explanation: TCP/IP issues this message while tracing is on. While attempting to validate the password for the currently queued job an error was detected.

System Action: LPD continues.

User or Operator Response: Inform the system programmer of the error.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPD

Procedure Name: ValidateJob

EZB0744I *address punch line*

Explanation: LPD displays the address of the punch command text and the punch line number. The punch line text represents the information sent to SMTP from LPD.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: Punchline

EZB0747E Could not allocate SMTP Spool

Explanation: The SMTP device could not allocate the spool for the batch data set submitted using the SMTP command.

System Action: LPD continues.

User or Operator Response: Notify the system programmer.

System Programmer Response: Make sure the SMTP device is activated.

Source Data Set: LPD

Procedure Name: SendFailingMail

EZB0748E Could not open spool to SMTP. Return code was *last error*

Explanation: LPD could not spool a job to the indicated SMTP service. A nonzero return code was returned.

System Action: LPD continues.

User or Operator Response: Notify the system programmer.

System Programmer Response: Make sure the SMTP statement is updated to include accurate information for its parameters in the *hlq*.LPD.CONFIG data set. Also make sure that the MAIL parameter is used with the FAILEDJOB statement. For more information about the SMTP and FAILEDJOB statements, refer to *z/OS Communications Server: IP Configuration Reference*. Check the return code issued in this message and the return codes listed under SMTP in the *z/OS Communications Server: IP and SNA Codes* to further determine and correct the error.

Source Data Set: LPR

Procedure Name: SendFailingMail

EZB0750E Could not deallocate Spool File *ddname* Error code was *rc*

Explanation: The SMTP server was unable to deallocate the specified spool file after attempting to close the SMTP connection.

System Action: LPD continues.

User or Operator Response: Notify the system programmer.

System Programmer Response: Check the return code issued in this message and the return codes listed under SMTP in the *z/OS Communications Server: IP and SNA Codes* to determine and correct the error.

Source Data Set: LPR

Procedure Name: SendFailingMail

EZB0751I Released StepBlock at *address*

Explanation: The SMTP server displays this message as the job at the specified address is deleted from storage.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: RemoveJobFiles

EZB0753I New subcommand *command* (no operands)

Explanation: This message is displayed while tracing is on. A new SMTP subcommand was detected by LPD with no optional operands.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: DoNewSubcommand

EZB0754I **New subcommand** *command* **operands** *operands*.

Explanation: This message is displayed while tracing is on. A new SMTP subcommand was detected by LPD. The new subcommand and optional operands are displayed.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: DoNewSubcommand

EZB0755I **Released StepBlock at** *address*

Explanation: This message occurs while tracing is on. This message is displayed when the previous subcommand is deleted from the stated storage area.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: DoNewSubcommand

EZB0756I **Job number** *number* **is invalid.**

Explanation: The LPD server was unable to queue the specified job number for processing. The connection was terminated.

System Action: LPD continues.

User or Operator Response: Resubmit the job using the LPR command.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPD

Procedure Name: DoNewSubcommand

EZB0757I **Duplicate file name** "*dataset name*".

Explanation: This message indicates that the specified data set was previously recognized by LPD. The job is ignored and the next job is processed.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DoNewSubcommand

EZB0759E Failed to allocate block for *jobnumber* from *site*

Explanation: The service machine was unable to allocate enough storage to process the specified job at the specified site.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DoNewSubCommand

EZB0760E Failed to open "*dataset name*".

Explanation: The server machine's attempt to open the stated data set failed. The connection is terminated.

System Action: LPD continues.

User or Operator Response: Notify the system programmer.

System Programmer Response: Make sure that the specified data set is defined in the MAKESITE tables.

Source Data Set: LPD

Procedure Name: DoNewCommand

EZB0761E Could not open data file "*dataset name*". Job abandoned.

Explanation: The server machine issues this message after attempting to open the data file at the specified site with the specified file type. The job is abandoned.

System Action: LPD continues.

User or Operator Response: Notify the system programmer.

System Programmer Response: Make sure the specified site is defined by the MAKESITE command.

Source Data Set: LPR

Procedure Name: DoSendStep

EZB0762I Sending subcommand *command* with an operand of *operand*

Explanation: LPD issues this message while tracing is on. LPD acknowledges sending the indicated subcommand with the indicated operand.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: DoSendJob

EZB0763I Closing connection *connection*

Explanation: This message is issued while tracing is on. The connection between the client and the LPD server has been closed.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DoCloseConnection

EZB0764I **ACK received on connection** *connection* **for job** *jobnumber* **in state** *jobstate*

Explanation: This message is issued while tracing is on. LPD has received an acknowledgment of the specified connection. The current state of the job is also displayed.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DoNewAck

EZB0765I **ACK in unexpected job state** *jobstate*

Explanation: This message is issued while tracing is on. LPD received an ACK from the remote host, but the job state was bad. See message EZB0764I for job and connection number.

System Action: LPD continues.

User or Operator Response: Check all physical ports and power switch to verify that the printer is ready for printing. If the problem persists, contact your hardware support personnel.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DoNewAck

EZB0766I **NACK has value** *number*

Explanation: This message is issued while tracing is on. The remote host refused to complete processing for this job.

System Action: LPD continues.

User or Operator Response: Check the printer and make sure it is active. If the problem persists, contact your hardware support personnel.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DoNewAck

EZB0767I **Timer cleared for connection** *connection*

Explanation: This message is issued while tracing is on. LPD has successfully cleared the timer for the specified connection.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DoNewData

EZB0768I **Ignoring Data delivered on connection** *connection*

Explanation: This message is issued while tracing is on. The buffer has been exhausted. The data to be delivered is ignored.

System Action: LPD continues.

User or Operator Response: Notify the system programmer.

EZB0769I • EZB0772I

System Programmer Response: Increase the specified buffer size using the DATABUFFERPOOLSIZE statement.

Source Data Set: LPD

Procedure Name: DoNewData

EZB0769I Job *jobnumber* removed from work queue

Explanation: This message is issued while tracing is on. LPD issues this message when the specified job number is removed from the list of queued jobs.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: RemoveJobWork

EZB0770I Job *jobnumber* not found in printer chain.

Explanation: LPD issues this message when the specified job number is not found in the print queue.

System Action: LPD halts.

User or Operator Response: Notify the system programmer.

System Programmer Response: Check the LPD trace. Job may have already printed. Restart LPD.

Source Data Set: LPD

Procedure Name: RemoveJobPrinter

EZB0771I Released JobBlock at *address*

Explanation: LPD issues this message while tracing is on. The storage space allocated for the job block at the specified address was released.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: FreeJob

EZB0772I End Connection *connection* for *errmsg* (*msgnum*)

Explanation: The specified LPD server connection has ended.

errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the **EZA** message whose text is displayed in *errmsg*. For more information about this message, see message EZAm*msgnum* in the *z/OS Communications Server: IP Messages Volume 1 (EZA)*.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Respond as indicated by the message EZAm*msgnum*.

Source Data Set: LPD

Procedure Name: DoEndConnection

EZB0773I Connection *connection* terminated for *errmsg* (*msgnum*)

Explanation: This message indicates that the specified connection has been terminated.

errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the **EZA** message whose text is displayed in *errmsg*. For more information about this message, see message *EZAmmsgnum* in the *z/OS Communications Server: IP Messages Volume 1 (EZA)*.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Respond as indicated by the message *EZAmmsgnum*.

Source Data Set: LPD

Procedure Name: DoEndConnection

EZB0774I Connection ended abruptly.

Explanation: The LPD connection with the remote printer has terminated. LPD received a job and the connection state was not CONNCLOSING.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Review the trace and restart the job. If DEBUG is not on, define DEBUG in the LPD.CONFIG and restart the job.

Source Data Set: LPD

Procedure Name: DoEndConnection

EZB0775I Released StepBlock at *address*

Explanation: This message is issued while tracing is on. The block at the specified address was freed of storage.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DoEndConnection

EZB0776I Released StepBlock at *address*

Explanation: This message is issued while tracing is on. The storage space specified was released because there were no jobs awaiting processing.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DoEndConnection

EZB0777I Released ConnectionBlock at *address*

Explanation: This message is issued while tracing is on. The storage space specified for the LPD connection was released because there were no jobs awaiting processing.

System Action: LPD continues.

EZB0778S • EZB0780I

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DoEndConnection

EZB0778S LPD terminating because *errmsg (msgnum)*

Explanation: This message is displayed while tracing is on. The LPD server has terminated operation.

errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the **EZA** message whose text is displayed in *errmsg*. For more information about this message, see message EZA*msgnum* in the *z/OS Communications Server: IP Messages Volume 1 (EZA)*.

System Action: LPD halts.

User or Operator Response: Notify system programmer.

System Programmer Response: Respond as indicated by the message EZA*msgnum*.

Source Data Set: LPD

Procedure Name: DoEndConnection

EZB0779I New connection state *state* on connection *address* with reason *errmsg (msgnum)*

Explanation: This message indicates that LPD has made a new connection state at the specified address.

errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the **EZA** message whose text is displayed in *errmsg*. For more information about this message, see message EZA*msgnum* in the *z/OS Communications Server: IP Messages Volume 1 (EZA)*.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Respond as indicated by the message EZA*msgnum*.

Source Data Set: LPD

Procedure Name: DoNewConnState

EZB0780I Abort failed *errmsg (msgnum)*

Explanation: This message is issued as a result of the LPD's unsuccessful attempt to disconnect the TCP connection.

errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the **EZA** message whose text is displayed in *errmsg*. For more information about this message, see message EZA*msgnum* in the *z/OS Communications Server: IP Messages Volume 1 (EZA)*.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Respond as indicated by the message EZA*msgnum*.

Source Data Set: LPD

Procedure Name: DoEndConnection

EZB0781I Connection aborted because port number (number) is out of range.

Explanation: This message is issued while tracing is on. The LPD connection was aborted because the stated port number was not defined as a legitimate port. Legitimate ports range from 721 to 731, inclusive.

System Action: LPD continues.

User or Operator Response: Notify the system programmer.

System Programmer Response: Use the PORT statement to define the legitimate ports. For more information about the PORT statement, see the *z/OS Communications Server: IP Configuration Reference*.

Source Data Set: LPD

Procedure Name: DoNewConnState

EZB0782I Connection open. Reading command.

Explanation: This message is issued while tracing is on. The LPD connection is open and now reading the submitted connection state command.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DoNewConnState

EZB0783I Aborting Connection *connection* - Timed out

Explanation: The specified connection was aborted because the timer expired.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Assist the user as necessary. Use the TIMER command to set the timer to an appropriate time limit.

Source Data Set: LPD

Procedure Name: DoConnectionTimeout

EZB0784I Could not retrieve SMSG

Explanation: LPD was unable to retrieve a queued Smsg. Therefore, the message could not be queued for processing.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Refer to message EZB0800I.

Source Data Set: LPD

Procedure Name: ProcessSMSG

EZB0785I Attempted SMSG from "user" ignored.

Explanation: The SMSG from the specified user was ignored. LPD was unable to locate the user ID in the OBEYFILE.

System Action: LPD continues.

User or Operator Response: Notify the system programmer.

EZB0786I • EZB0790I

System Programmer Response: Make sure the specified user ID is listed in the OBEYFILE using the OBEY command.

Source Data Set: LPD

Procedure Name: ProcessMSG

EZB0786I **Command received** *"string"*

Explanation: LPD issues this message when the specified special messages string is received.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: ProcessMSG

EZB0788I **Command not understood.**

Explanation: LPD does not recognize the command issued through the MSG interface.

System Action: LPD continues.

User or Operator Response: Correct the syntax and reissue the command.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: ProcessMSG

EZB0789I **GetNextNote with ShouldWait of** *number*

Explanation: This message is issued while tracing is on. TCP is initializing processing procedures to retrieve the next queued notification. The ShouldWait function is set to either a true or false value depending on whether TCP is to wait for notification before the next one is available.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: ProcessTCP

EZB0790I **GetNextNote returns. Connection** *connection* **Notification** *notification*

Explanation: This message is issued while tracing is on. The connection number and the notification status are returned upon successful completion of the GetNextNote procedure.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: ProcessTCP

EZB0791I New TCP notice arrived

Explanation: This message is issued while tracing is on. TCP has received the next queued notification.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: ProcessTCP

EZB0792I Connection: *connection*

Explanation: This message is issued while tracing is on. This message displays the connection number of the TCP connection.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: ProcessTCP

EZB0793I Notification: *notification*

Explanation: This message is issued while tracing is on. This message displays the notification status of the current TCP connection.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: ProcessTCP

EZB0794I NewState: *newstate*

Explanation: This message is issued while tracing is on and notification has changed. The new state is displayed in this message.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: ProcessTCP

EZB0795I ConnState: *connectionstate*

Explanation: This message is issued while tracing is on. If TCP does not detect a connection state change, this message is displayed.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: ProcessTCP

EZB0796I BytesDelivered: *number*

Explanation: This message is issued while tracing is on. The number of data bytes delivered on the TCP connection are displayed.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: ProcessTCP

EZB0797I SendTurnCode: *errmsg (msgnum)*

Explanation: This message is issued while tracing is on. TCP failed in its attempt to send data on the TCP connection.

errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the **EZA** message whose text is displayed in *errmsg*. For more information about this message, see message **EZAmsgnum** in the *z/OS Communications Server: IP Messages Volume 1 (EZA)*.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Respond as indicated by the message **EZAmsgnum**.

Source Data Set: LPD

Procedure Name: ProcessTCP

EZB0798I Queueing job *jobnumber*

Explanation: This message is issued while tracing is on. The specified job is queued for processing.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: ProcessTCP

EZB0799I Reading additional data on *connection*

Explanation: This message is issued while tracing is on. Processing of the current job continues on the specified connection.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: ProcessTCP

EZB0800I Ignoring TCP/IP notice *notification on connection*

Explanation: This message is issued while tracing is on. The TCP connection has failed or was not recognized. Use the type identifier issued in this message and the *z/OS Communications Server: IP Programmer's Reference* to determine and correct the problem.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPD

Procedure Name: ProcessTCP

EZB0801I Filter *option* not supported. Job abandoned.

Explanation: This message could display for one of the following reasons:

- The option type of the filters parameter used with the LPR command is not supported by the line printer daemon.
- The option type of the filters parameter for the SERVICE statement in the *hlq.LPD.CONFIG* data set is not supported by the line printer daemon.

System Action: LPD continues.

User or Operator Response: Correct the option type of the filters parameter for the LPR command or in the SERVICE statement of the *hlq.LPD.CONFIG* data set. Reissue the job. Refer to the *z/OS Communications Server: IP User's Guide and Commands* for information about the LPR command or the *z/OS Communications Server: IP Configuration Reference* for information about the SERVICE statement.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DoStartStep

EZB0802E Could not open data file *dataset* Job abandoned.

Explanation: LPD was unable to access the indicated data set. The print job is not completed.

System Action: LPD continues.

User or Operator Response: Reissue the print job with a valid data set name or PDS member. See the *z/OS Communications Server: IP User's Guide and Commands* for more information about using the LPR command. None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DoStepStart

EZB0803E Could not define device for job. Unknown type *type* treated as "NONE".

Explanation: LPD encountered a printer type defined in the SERVICE statement that was not valid or unknown.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Correct the device type parameter for the SERVICE statement in *hlq.LPD.CONFIG* data set. For more information on the SERVICE statement, refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: GetJobDevice

EZB0804E *printer kind is unknown address*

Explanation: LPD was not able to recognize the route parameter for the indicated printer. The route parameter was defined to something other than remote, local or NJE.

System Action: LPD continues.

User or Operator Response: Notify the system programmer of this message.

System Programmer Response: Correct the route parameter for the SERVICE statement in the *hlq*.LPD.CONFIG data set. For information on the SERVICE statement, refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: GetJobDevice

EZB0805E *Could not allocate Spool Class class*

Explanation: The LPD server was unable to set a spool file for the specified printer class due to buffer exhaustion. See messages EZB0806I and EZB0807I.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Resubmit the LPD command.

Source Data Set: LPD

Procedure Name: GetJobDevice

EZB0806I *Copies copies, Font font, form form, output printer*

Explanation: LPD issues this message when a nonzero return code is returned during spool file allocation. The number of copies, font size, form, and printer name are provided. See message EZB0807I.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: GetJobDevice

EZB0807I *bufferlength*

Explanation: The length of the data buffer which failed while attempting to allocate a spool file is displayed.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: GetJobDevice

EZB0808I *bufferlength*

Explanation: The length of the alternate data buffer which failed while attempting to allocate a spool file is displayed.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: GetJobDevice

EZB0809E **Could not open spool to destination Return code was error**

Explanation: LPD was unable to open a spool file to the specified printer destination. A return code is passed.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Use the return code listed in this message and *z/OS Communications Server: IP and SNA Codes*.

Source Data Set: LPD

Procedure Name: GetJobDevice

EZB0810I **Spool printertype address for user. Return code was rc**

Explanation: LPD issues this message while tracing is on or a nonzero return code is displayed. The printer type, address, user, and return code are displayed with this message after the spool is allocated.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Check the meaning of the RC issued by CP spool "FOR" command.

Source Data Set: LPD

Procedure Name: GetJobDevice

EZB0811E **printertype kind is unknown deviceaddress**

Explanation: The LPR command has a destination defined that is not known. The destination is defined after the jobname parameter.

System Action: LPD continues.

User or Operator Response: Specify the correct destination in the jobname parameter and reissue the LPR command. For more information on the LPR command see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: GetJobDevice

EZB0812I **Spool TO address superseded by FOR user**

Explanation: The SMTP spool to the default user address was superseded by the specified user id.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: GetJobDevice

EZB0813I Spooling *printer this way* *how*

Explanation: This message is displayed while tracing is on. LPD acknowledges the parameter indicated for the LPR command on the specified printer.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: GetJobDevice

EZB0814E Could not spool *printer address type*. Return code was *rc*

Explanation: This message is issued while tracing is on. The LPD server was not able to spool the printer at the stated address with the specified type. A nonzero return code was returned.

System Action: Printer not spooled.

User or Operator Response: None.

System Programmer Response: Use the return code value to determine the error and reissue the command.

Source Data Set: LPD

Procedure Name: GetJobDevice

EZB0815I Tagging *printer with* *tag*

Explanation: This message is issued while tracing is on. The indicated RSCS tag has been assigned to the indicated printer.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: GetJobDevice

EZB0816E Could not TAG *printer address tag* Return code was *rc*

EZB0817I Response was *response*

Explanation: The LPD server was not able to assign the specified tag to the printer at the indicated address. A nonzero return code was passed.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Use the return code value and the response string provided with message EZB0817I to determine and correct the cause of the error.

Source Data Set: LPD

Procedure Name: GetJobDevice

EZB0819I Job *jobnumber* rescheduled -- no storage

Explanation: This message is issued while tracing is on. LPD was unable to allocate a connection for the specified job number. No storage block was available for a connection.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DoStartSending

EZB0820I Trying to open with local port *port*

Explanation: This message is issued while tracing is on. LPD is attempting to open the specified local port for communication.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DoStartSending

EZB0821I Job *jobnumber* abandoned -- Open failed or no ports

Explanation: This message is issued while tracing is on.

- LPD was unable to open a connection to the specified REMOTE Print Server (LPD) on another host. A TcpOpen error occurred.
- LPD was unable to open a local port to the specified REMOTE Print Server (LPD) on another host. No ports in the range 721–731 were available.

In the message text:

jobnumber The number assigned by the LPR client to this job.

System Action: LPD continues.

User or Operator Response: Resubmit the print job when a connection to the REMOTE print server can be established.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DoStartSending

EZB0822I Sending command *number* with operand *address*

Explanation: This message is issued while tracing is on. LPD is sending the specified command to the specified printer address.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DoOpenSending

EZB0823I Sending ACK at end of data file on *connection* for job *jobnumber*

Explanation: This message is issued while tracing is on. The LPD server has read the dataset on the specified connection for the specified job number and is sending an ACK.

System Action: LPD continues.

User or Operator Response: None.

EZB0824I • EZB0831I

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: DoContinueSending

EZB0824I **ProcessWork starting on job queue**

Explanation: This message is issued while tracing is on. LPD is now processing the next job queued.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: ProcessWork

EZB0825I **Job *jobnumber* for printer dispatched in state *state***

Explanation: This message is issued while tracing is on. The job number, printer, and print state are displayed.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: ProcessWork

EZB0826E **Job *jobnumber* (state) abandoned - Incorrect state.**

Explanation: LPD has terminated the processing of the specified job number because an incorrect job state was detected.

System Action: LPD continues.

User or Operator Response: Resubmit the job using the LPR command.

System Programmer Response: Make sure the failed job is mailed to the user using the MAIL command. For more information see the *z/OS Communications Server: IP Programmer's Reference*.

Source Data Set: LPD

Procedure Name: ProcessWork

EZB0827I **ProcessWork end with queue**

Explanation: This message is issued while tracing is on. LPD has finished processing the queued jobs.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: ProcessWork

EZB0831I **IBM MVS LPD Version *version* on date at time**

Explanation: When the LPD server is initialized this message is displayed. The LPD version number, date, and starting time are displayed.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: LPD MAIN

EZB0833I Could not get identity!

Explanation: The LPD server was unable to identify the user id, host name, TCP service and domain name during its start up procedure. LPD does not start.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Make sure the user id, host name, TCP service, and domain name have all been specified using the LPD command and its subcommands. For more information of the LPD command and its subcommands, refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: LPD

Procedure Name: LPD MAIN

EZB0834I Ready

Explanation: LPD is initialized and ready for processing.

System Action: LPD is initialized.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: LPD MAIN

EZB0835I Ignored data is *string*

Explanation: This message displays while tracing is on; it is preceded by EZB0768I. Data was delivered on the connection, but because the data is unrecognizable, it is ignored. The connection is closed.

System Action: LPD continues.

User or Operator Response: Refer to EZB0768I.

System Programmer Response: Refer to EZB0768I. Use the data from the message to determine if the LPD was sent bad data.

Source Data Set: LPD

Procedure Name: DoNewData

EZB0850I Using Table for DBCS Translate: *data set*

Explanation: The DBCS translation table has been successfully loaded from the binary translation table data set in the search order hierarchy.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: LoadDbcsTables

EZB0851I Could not load DBCS Translate Table: *data set*

Explanation: DBCS conversion is configured for the LPD, but the required DBCS translation table could not be loaded.

System Action: The program continues.

User or Operator Response: None.

System Programmer Response: Configure a valid DBCS binary translate table data set in the search order hierarchy for the required DBCS translation table. Refer to *z/OS Communications Server: IP Configuration Reference* for more information about loading and customizing DBCS translation tables.

Source Data Set: LPD

Procedure Name: LoadDbcsTables

EZB0852I Use a NLS option after "NLSTRANSULATE".

Explanation: An NLS option is expected after the NLSTRANSULATE keyword of the SERVICE statement of the *hlq.LPD.CONFIG* data set.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: Specify the correct value in the parameter of NLSTRANSULATE keyword of the SERVICE statement of the *hlq.LPD.CONFIG* data set and restart the program.

Source Data Set: LPD

Procedure Name: ProcessNlsOptions

EZB0853E Use an integer after *JOBPACING*.

Explanation: A number is expected after the JOBPACING statement in the *hlq.LPD.CONFIG* data set.

System Action: LPD continues using the default value.

User or Operator Response: None.

System Programmer Response: Specify the correct value for the parameter in the JOBPACING statement in the *hlq.LPD.CONFIG* data set and restart the program.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0854E Use an integer after *STEPLIMIT*.

Explanation: A number is expected after the STEPLIMIT statement in the *hlq.LPD.CONFIG* data set.

System Action: LPD continues using in the default value.

User or Operator Response: None.

System Programmer Response: Specify the correct value for the parameter in the STEPLIMIT statement in the *hlq.LPD.CONFIG* data set and restart the program.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0855I Loaded translation table from *dataset* for printer *printer_name*

Explanation: The Line Printer Daemon (LPD) loaded the following SBCS translation table corresponding to the name provided by the TRANSLATETABLE or XLATETABLE statement in the *hlq.LPD.CONFIG* data set. This statement specifies the translation table to be used by the remote client for SBCS ASCII to EBCDIC translations.

dataset is the fully qualified data set name.

printer_name is the name of the printer service.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: PreparePrinters

EZB0856I **Loaded translation table from *dataset***

Explanation: The Line Printer Daemon (LPD) loaded the SBCS translation table to be used if the printer service does not have TRANSLATETABLE or XLATETABLE statements specified.

dataset is the fully qualified data set name.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: ProcessOptions

EZB0857I **Using hardcoded translation tables.**

Explanation: Hardcoded translation tables will be used because the Line Printer Daemon (LPD) was unable to load an SBCS translation table using the data set names of *jobname*.STANDARD.TCPXLBIN or *hlq*.STANDARD.TCPXLBIN. These tables are equivalent to *hlq*.STANDARD.TCPXLBIN, which are shipped with the product.

See the *z/OS Communications Server: IP Configuration Reference* for more information about using translation tables including search order hierarchy and customization.

System Action: LPD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPD

Procedure Name: ProcessOptions

EZB0900I *command name* **version** *version*

Explanation: Displays the line printer command used and the version level of the program. This message is displayed when you use the *version* parameter with the LPR, LPQ, LPRM, and LPRSET commands.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR, LPQ, LPRM, LPRSET

Procedure Name: ProcessVersionOption

EZB0901E **The option *option* is ambiguous. Use a longer abbreviation.**

Explanation: The abbreviated option submitted at the command line is ambiguous.

System Action: The command is terminated.

User or Operator Response: Reissue the desired option with a longer abbreviation at the LPR command line. For a list of valid abbreviations using the LPR command, see the *z/OS Communications Server: IP User's Guide and Commands*.

EZB0902E • EZB0905E

System Programmer Response: None.

Source Data Set: LPR, LPRM

Procedure Name: ProcessOptions

EZB0902E **Use a host name after HOST option.**

Explanation: The HOST option was specified without indicating host name.

System Action: The command is terminated.

User or Operator Response: Reissue a valid host name at the HOST option. For a list of valid host names, contact the system operator.

System Programmer Response: None.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: ProcessOptions

EZB0903E **The option *option* was not recognized.**

Explanation: The program cannot recognize the option you have entered. Valid options will be provided. This message will precede the valid options for your command.

System Action: The command is terminated.

User or Operator Response: Reissue the valid option using the LPR command.

System Programmer Response: None.

Source Data Set: LPQ, LPR, LPRM, LPRSET

Procedure Name: ProcessOptions

EZB0904I **Use the ALL, HOST, PRINTER, TRACE, TYPE or VERSION options as needed.**

Explanation: This message displays valid options that may be used with LPT, LPQ, and LPRM commands.

System Action: The command is terminated.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPQ, LPRM

Procedure Name: ProcessOptions

EZB0905E **Use a printer name after PRINTER option.**

Explanation: The PRINTER option was specified without specifying a printer name.

System Action: The command is terminated.

User or Operator Response: Reissue the LPR command and the PRINTER option using a valid printer. For more information about the LPR command and the PRINTER option, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: None.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: ProcessOptions

EZB0906E Program error: Invalid option *option*.

Explanation: You specified an invalid option. To see a list of options, type HELP followed by your command or see *z/OS Communications Server: IP User's Guide and Commands* for more information.

System Action: The command is terminated.

User or Operator Response: Reissue the LPR command and a valid option.

System Programmer Response: None.

Source Data Set: LPR, LPQ, LPRM, LPRSET

Procedure Name: ProcessOptions

EZB0907E Cannot get a printer name.

Explanation: The program cannot find a default printer name in the *user_id*.LASTING.GLOBALV data set.

System Action: Processing continues.

User or Operator Response: Use the LPRSET command to set up a default printer name. For more information see *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: None.

Source Data Set: LPQ, LPR, LPRM

Procedure Name: ProcessOptions

EZB0908I Printer name from global variable PRINTER = "printer"

Explanation: LPR displays the default printer name, taken from the *user_id*.LASTING.GLOBALV data set. For more information about the *user_id*.LASTING.GLOBALV data set, refer to *z/OS Communications Server: IP Configuration Reference*.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: ProcessOptions

EZB0909E Use the PRINTER option this way: PRINTER nameofprinter<@printerhost>

Explanation: You issued the PRINTER option with the *at-sign* without specifying the printer name.

System Action: The command is terminated.

User or Operator Response: Reissue the printer option using the correct syntax as described in the message.

System Programmer Response: None.

Source Data Set: LPR, LPRM, LPQ

Procedure Name: ProcessOptions

EZB0910E Cannot get the printer host.

Explanation: The program cannot find the default host name from the *user_id*.LASTING.GLOBALV data set.

System Action: The command is terminated.

User or Operator Response: Specify the host name on the command line, or use the LPRSET command to set up a default host name. For more information about the LPRSET command see *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: None.

EZB0911I • EZB0914E

Source Data Set: LPQ, LPR, LPRM

Procedure Name: ProcessOptions

EZB0911I **Host name from global variable PRTHOST = "host "**

Explanation: If the LPR, LPQ, or LPRM command lines do not have a specified printer host, the GLOBALV variables *PRINTER* and *PRTHOST* are reviewed in the *user_id*.LASTING.GLOBALV data set and the name of the host in use is displayed.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: ProcessOptions.

EZB0912E **The printer name is not known.**

Explanation: The program does not recognize the printer name.

System Action: Processing continues.

User or Operator Response: Check the printer name at the server machine, and reissue the command with a valid printer name or use the LPRSET command to set a default printer name at the specified host. See the *z/OS Communications Server: IP User's Guide and Commands* for more information about setting up a default printer in the *user_id*.LASTING.GLOBALV data set.

System Programmer Response: None.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: ProcessOptions

EZB0913E **The host name is not known.**

Explanation: The program cannot determine the host name.

System Action: The command is terminated.

User or Operator Response: Use a valid host name with the command. The LPRSET command can be used to set up a default host name for a default printer. See *z/OS Communications Server: IP User's Guide and Commands* for more information about setting up a default host in the *user_id*.LASTING.GLOBALV data set.

System Programmer Response: None.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: ProcessOptions

EZB0914E **Please specify the printer name and host either as a command option or with the LPRSET command.**

Explanation: The program cannot determine the printer name or the printer host.

System Action: Processing continues.

User or Operator Response: Reissue your command with a valid printer name and host. The LPRSET command can be used to set a default printer and host using the GLOBALV variables in the *user_id*.LASTING.GLOBALV data set. For more information, see *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: If the problem continues, obtain more information using the TYPE or TRACE functions.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: ProcessOptions

EZB0915I Begin “cmd” to printer “printer” at host “foreignhost”

Explanation: This message indicates which printer and remote host are being used for your task. The task is indicated by “cmd” in the message, which can be LPR, LPQ, or LPRM.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: ProcessOptions

EZB0916I Sending command *command* argument: *operand*

Explanation: This message indicates the command was successfully sent to the remote host.

System Action: The system continues processing.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: SendCommand

EZB0917I Command successfully sent

Explanation: The command was successfully sent and acknowledged by the remote print server.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: SendCommand

EZB0918E Command was not sent successfully

Explanation: The command was not acknowledged by the remote host; therefore, the transmission was unsuccessful.

System Action: The command is terminated.

User or Operator Response: Reenter the command using the correct syntax.

System Programmer Response: Assist the operator if necessary.

Source Data Set: LPQ, LPRM

Procedure Name: SendCommand

EZB0919I InitEmulation failed

Explanation: An error occurred initializing EC mode emulation. EC mode emulation is required to run TCP/IP on a non-EC mode machine.

System Action: Processing terminated.

User or Operator Response: Reinitialize EC mode to on. If a problem occurs contact your system programmer.

System Programmer Response: EC-mode emulation can only be accessed when the processor is running. If a program is running in a 24-bit addressing mode, the program can be changed to a 31-bit addressing mode, which causes a branch to a module residing above the 16Mb virtual storage. For more information see *IBM Assembler Language Programming Book*.

EZB0920I • EZB0924I

Source Data Set: LPQ, LPR, LPRM

Procedure Name: LPR Main

EZB0920I Requesting TCP/IP service at *date time*

Explanation: This message indicates the time and date that LPR, LPQ, or LPRM requested TCPIP service. This message is displayed if the TRACE option is specified.

System Action: Process continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: LPR Main

EZB0921I Granted TCP/IP service at *date time*

Explanation: TCPIP service is provided for the command requested.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: LPR Main

EZB0922I Resolving *foreignhost* at *date time*

Explanation: The remote host is being resolved at the indicated time and date.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: DrainConnection, PrintLineWithTabs

EZB0923I Both the printer and the host name are not known.

Explanation: The program could not determine the printer name or the host name.

System Action: The command is terminated.

User or Operator Response: Check the host name and printer name for the correct information and reenter the command. Use the LPRSET command to set a default printer and host name. For more information see *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: None.

Source Data Set: LPQ, LPR, LPRM

Procedure Name: ProcessOptions

EZB0924I Host *host* name resolved to *host_addr* at *date time*

Explanation: The remote host name has been resolved to the indicated Internet address along with the time and date.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: LPR Main

EZB0925I **TCP/IP turned on.**

Explanation: TCPIP services are now ready for use.

System Action: Processing continues with the next request.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: LPR Main

EZB0926I **Host “host” Domain “domain” TCP/IP Service Machine name**

Explanation: This message indicates the host name, its domain name equivalent, and the TCPIP service machine in use.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: LPR Main

EZB0927I **Trying to open with local port *port* to foreign host address *address***

Explanation: This message is issued while tracing is on and LPR is first initiated. TCPIP is attempting to open a connection with the stated local port to the stated foreign host address.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: LPR Main

EZB0928I **Connection open from local port *port* to foreign host address *address***

Explanation: This message is issued while tracing is on. TCPIP indicates that the specified local port has successfully connected to the specified foreign host address.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: LPR Main

EZB0929I **Connected to *host***

Explanation: The local port has completed a successful telecommunication line to the remote port. This means your command has been accepted and transmitted through TCPIP services.

System Action: Processing continues with the next request.

EZB0930I • EZB0934I

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: LPR Main

EZB0930I Connection closed

Explanation: TCPIP services ends the connection when your task has completed processing.

System Action: Processing ends successfully.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: LPR Main, LPQ Main, or LPRM Main

EZB0931I Notification: *SayNotEn*

Explanation: This message provides information about whether the data has been delivered and whether the connection state has been changed.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPQ, LPRM

Procedure Name: DrainConnection, ReceiveData

EZB0932I NewState: *SayConSt*

Explanation: The state of the connection to TCPIP has changed. The new state is indicated in the message.

System Action: The system continues processing.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPQ, LPRM

Procedure Name: DrainConnection, ReceiveData

EZB0933I ConnState: *SayConSt*

Explanation: This message indicates the state of the connection to TCPIP.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPQ, LPRM

Procedure Name: DrainConnection, ReceiveData

EZB0934I BytesToRead: *bytes*

Explanation: This message indicates the number of bytes contained in a transmission from TCPIP.

System Action: The system continues processing.

User or Operator Response: If the data is not acknowledged, TCPIP will retransmit the data to make sure that the data is received.

System Programmer Response: None.

Source Data Set: LPQ, LPRM

Procedure Name: DrainConnection, ReceiveData

EZB0935I BytesDelivered: *bytes*

Explanation: This message indicates the number of bytes delivered for your task, if the trace option is used.

System Action: The system continues processing.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPQ, LPRM

Procedure Name: ReceiveBytes, ReceiveData

EZB0936E BeginTcpi: *errmsg (msgnum)*

Explanation: The function BeginTcpi, which is used to start TCPIP service for LPQ or LPRM, was unsuccessful. *errmsg* is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the EZA message whose text is displayed in *errmsg*. For more information about this message, see message EZAmmsgnum in the *z/OS Communications Server: IP Messages Volume 1 (EZA)*.

System Action: Processing halts.

User or Operator Response: Notify the system programmer of the error.

System Programmer Response: Respond as indicated by the message EZAmmsgnum.

Source Data Set: LPQ, LPRM

Procedure Name: LPR Main, LPQ Main or LPRM Main

EZB0939E Could not get identity! (*errmsg (msgnum)*)

Explanation: TCPIP was unable to identify the user requesting service.

errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the EZA message whose text is displayed in *errmsg*. For more information about this message, see message EZAmmsgnum in the *z/OS Communications Server: IP Messages Volume 1 (EZA)*.

System Action: Processing ends.

User or Operator Response: Respond as indicated by the message EZAmmsgnum.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: LPR Main, LPQ Main or LPRM Main

EZB0940E Unknown host *host*

Explanation: The specified host does not exist or is entered incorrectly.

System Action: Processing halts.

User or Operator Response: Reenter the correct host name or address using the LPRSET command.

System Programmer Response: None.

EZB0941E • EZB0944E

Source Data Set: LPR, LPRM, LPQ

Procedure Name: LPR Main

EZB0941E Handle: *errmsg (msgnum)*

Explanation: The Handle procedure, which specifies what notifications to receive in a given set, was unsuccessful.

errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the **EZA** message whose text is displayed in *errmsg*. For more information about this message, see message *EZAmsgnum* in the *z/OS Communications Server: IP Messages Volume 1 (EZA)*.

System Action: Processing halts.

User or Operator Response: Notify the system programmer of the error.

System Programmer Response: Respond as indicated by the message *EZAmsgnum*.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: LPR Main

EZB0942E Connection to host failed.

Explanation: The attempt to connect to the specified host was not successful.

System Action: Processing halts.

User or Operator Response: Try to restart the connection by issuing the LPD command. For more information on the LPD command check the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: None.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: LPR Main

EZB0943I No local printer ports available now. Bind Conn failed.

Explanation: All of the local printer ports are either busy or not ready.

System Action: Processing halts.

User or Operator Response: Wait and try the command again.

System Programmer Response: None.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: LPR Main

EZB0944E Could not set option (errmsg (msgnum))

Explanation: The procedure that sets an option for a TCP connection was unsuccessful.

errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the **EZA** message whose text is displayed in *errmsg*. For more information about this message, see message *EZAmsgnum* in the *z/OS Communications Server: IP Messages Volume 1 (EZA)*.

System Action: Processing halts.

User or Operator Response: Notify the system programmer.

System Programmer Response: Respond as indicated by the message *EZAmsgnum*.

Source Data Set: LPQ, LPRM LPR, LPQ, LPRM

Procedure Name: LPR Main, LPR Main, LPRM Main

EZB0945E Could not send command *errmsg (msgnum)*

Explanation: The function which sends the commands was unsuccessful.

errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the **EZA** message whose text is displayed in *errmsg*. For more information about this message, see message *EZAmmsgnum* in the *z/OS Communications Server: IP Messages Volume 1 (EZA)*.

System Action: Processing halts.

User or Operator Response: Notify the system programmer

System Programmer Response: Respond as indicated by the message *EZAmmsgnum*.

Source Data Set: LPRM, LPQ

Procedure Name: LPR Main

EZB0946E Failed to read buffer *errmsg (msgnum)*

Explanation: The procedure that attempts to receive the data was unsuccessful.

errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the **EZA** message whose text is displayed in *errmsg*. For more information about this message, see message *EZAmmsgnum* in the *z/OS Communications Server: IP Messages Volume 1 (EZA)*.

System Action: Processing halts.

User or Operator Response: Notify the system programmer.

System Programmer Response: Respond as indicated by the message *EZAmmsgnum*.

Source Data Set: LPRM, LPQ

Procedure Name: LPR Main

EZB0947E TcpClose&colon *errmsg (msgnum)*

Explanation: The function that closes the connection was unsuccessful.

errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the **EZA** message whose text is displayed in *errmsg*. For more information about this message, see message *EZAmmsgnum* in the *z/OS Communications Server: IP Messages Volume 1 (EZA)*.

System Action: Processing halts.

User or Operator Response: Notify the system programmer.

System Programmer Response: Respond as indicated by the message *EZAmmsgnum*.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: LPR Main

EZB0948E Could not abort connection *errmsg (msgnum)*

Explanation: The application was unable to abort a connection.

errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the **EZA** message whose text is displayed in *errmsg*. For more information about this message, see message *EZAmmsgnum* in the *z/OS Communications Server: IP Messages Volume 1 (EZA)*.

System Action: Processing halts.

EZB0949E • EZB0952E

User or Operator Response: Notify the system programmer.

System Programmer Response: Respond as indicated by the message EZAmgnum.

Source Data Set: LPR, LPQ, LPRM

Procedure Name: LPR Main

EZB0949E Failed to send from SendACK. Return Code = rc and Error Number = errno.

Explanation: The SendACK function, used by the server to relay acknowledgment, was not received by the host. The current port number and the user IP address are displayed with this message.

errno is the UNIX System Services Return Code. Return Codes are listed and described in the *z/OS UNIX System Services Messages and Codes*.

System Action: Processing continues.

User or Operator Response: See the *z/OS Communications Server: IP and SNA Codes* for information about TCP/IP return codes and the *z/OS UNIX System Services Messages and Codes* for information about the *errno*.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPR

Procedure Name: SendACK

EZB0950E Failed to send from SendCommand. Return Code = rc and Error Number = errno.

Explanation: The function that sends the acknowledgment was unsuccessful. The command was not sent. The current port number and the user IP address are displayed with this message.

errno is the UNIX System Services Return Code. Return Codes are listed and described in the *z/OS UNIX System Services Messages and Codes*.

System Action: LPR halts.

User or Operator Response: See the *z/OS Communications Server: IP and SNA Codes* for information about TCP/IP return codes and the *z/OS UNIX System Services Messages and Codes* for information about the *errno*.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPR

Procedure Name: SendCommand

EZB0951E Did not receive ACK for receive control file command

Explanation: SendCommand did not receive an acknowledgment code from the receiving host, and the control file command was not received.

System Action: LPR halts.

User or Operator Response: Retry the command making sure the host is ready and the connection is open.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPR

Procedure Name: SendControlFile

EZB0952E Failed to send control line block. Return Code = rc and Error Number = errno.

Explanation: The function that sends the data was unsuccessful. The current port number and the user IP address are displayed with this message.

errno is the UNIX System Services Return Code. Return Codes are listed and described in the *z/OS UNIX System Services Messages and Codes*.

System Action: LPR halts.

User or Operator Response: See the *z/OS Communications Server: IP and SNA Codes* for information about TCP/IP return codes and the *z/OS UNIX System Services Messages and Codes* for information about the *errno*.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPR

Procedure Name: SendControlFile

EZB0953E Did not receive ACK for control file

Explanation: The function TcpFReceive returned an FRECEIVEError, indicating that the receive request was rejected. The control file was not received.

System Action: LPR halts.

User or Operator Response: Make sure the receiving host is ready and retry the request.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPR

Procedure Name: SendControlFile

EZB0954E File could not be opened.

Explanation: LPR was unable to open a data set.

System Action: Processing continues.

User or Operator Response: Verify that the data set is in storage accessible to LPR.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendControlFile

EZB0955E File contains no valid print lines.

Explanation: The data set is empty.

System Action: Processing continues.

User or Operator Response: Make sure the data set you wish to print contains at least one character.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendControlFile

EZB0956E Failed to send data block *errmsg* (*msgnum*)

Explanation: The attempt to send the information was unsuccessful.

errmsg is the text of the message that describes the error.

msgnum is the 4-digit numeric portion of the message identifier of the **EZA** message whose text is displayed in *errmsg*. For more information about this message, see message **EZAmsgnum** in the *z/OS Communications Server: IP Messages Volume 1 (EZA)*.

System Action: Processing continues.

User or Operator Response: Notify the system programmer.

System Programmer Response: Respond as indicated by the message **EZAmsgnum**.

Source Data Set: LPR

Procedure Name: SendLineFlush

EZB0957E Did not receive ACK for receive data file command. Code = *code*.

Explanation: The host did not receive an acknowledgment from the LPR receive data command.

System Action: LPR halts.

User or Operator Response: See *z/OS Communications Server: IP and SNA Codes* for information on return codes and error numbers. The current port number and the user IP address are displayed with this message.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB0958E File could not be opened.

Explanation: The specified data set could not be opened.

System Action: LPR halts.

User or Operator Response: Verify that you have access to the data set and try to open it again.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB0959E Did not receive ACK for data file

Explanation: The remote host did not acknowledge that it received a data set.

System Action: If no acknowledgement is received, the data set is retransmitted. Processing continues.

User or Operator Response: Notify the system programmer of the error.

System Programmer Response: Check why LPD did not send acknowledgement. Starting LPD (server) with "DEBUG" on, will help.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB0960E Did not receive ACK after close

Explanation: The remote host did not send a message acknowledging that it has closed. The connection remains open.

System Action: Processing continues.

User or Operator Response: Try to close the connection again.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: DrainConnection

EZB0961I Control file name is *dataset name*

Explanation: The name of the control data set is displayed.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: MakeFileNames

EZB0962I **Data file name is** *dataset name*

Explanation: The name of the current data set is displayed.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: MakeFileNames

EZB0965E **Use the form:** *command name* **DataSetName.**

Explanation: This message displays the correct format for the indicated command. For more information about the commands, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Action: The command is not processed.

User or Operator Response: Reenter the command using the form given in the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPR

Procedure Name: ProcessOperands

EZB0967E **Data Set name** *"name"* **invalid.**

Explanation: The specified data set name is incorrect.

System Action: The command is not processed.

User or Operator Response: Make sure that the data set name uses the correct format and syntax and that the data set is in storage accessible to the host.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPD, LPR

Procedure Name: ProcessOperands

EZB0968E **Data Set** *"name"* **not found or inaccessible, return code =** *rc*.

Explanation: This message may issue one of the following return codes:

Return code

	Description
4	Data set name not found
12	Data set name is missing
24	Data set is an unprintable VSAM file

If another return code is issued, this indicates that the data set attributes may make the data set inaccessible.

System Action: LPR halts.

User or Operator Response: Reenter the LPR command using the correct data set name. If the data set name is correct, check the data set attributes.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPR

Procedure Name: ProcessOperands

EZB0969E Data Set "*name*" does not contain member "*member*".

Explanation: The specified data set name does not contain the specified member name.

System Action: The LPR command is not processed.

User or Operator Response: Reenter the command using the correct data set name and the correct member name.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: ProcessOperands

EZB0970E Data Set "*name*" invalid organization.

Explanation: The function FindDSName exited abnormally because the file's organization was incorrect.

System Action: Processing continues.

User or Operator Response: Recheck the organization and reenter the file. Notify the system programmer if the problem persists.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPR

Procedure Name: ProcessOperands

EZB0971E Use a matching quotation mark at the end of the string.

Explanation: There is a missing quotation mark at the end of the string.

System Action: The LPR command is not processed.

User or Operator Response: Make sure there are appropriate quotation marks and reenter the string.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: LPRToken

EZB0973E Use either **BINARY** or **NOBINARY** but not both.

Explanation: Either BINARY or NOBINARY can be chosen. Data cannot be sent as BINARY and NOBINARY.

System Action: The LPR command is not processed.

User or Operator Response: Reenter the LPR command specifying either the BINARY or NOBINARY options.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: ProcessOptions

EZB0974E Use a token after the **CLASS** option.

Explanation: After the CLASS option, you must enter a name of a host. If you do not want to name another host name, the CLASS option is not required. For more information on the LPR command and its options, refer to the *z/OS Communications Server: IP User's Guide and Commands*.

System Action: LPR command is not processed.

User or Operator Response: Reenter the LPR command and the CLASS option using a name of a host.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPR

Procedure Name: ProcessOptions

EZB0975E Use a count after the COPIES option.

Explanation: This message indicates that there is no number following the COPIES option. The COPIES option specifies how many copies are to be printed. If this option is not used, one copy will be printed.

System Action: LPR command is not processed.

User or Operator Response: Reenter the LPR command with the COPIES option and fill in a number, specifying the number of copies you want printed. If only one copy is needed, the COPIES option is not required.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: ProcessOptions

EZB0976E Use a letter after the FILTER option.

Explanation: No filter was specified after the filter option in the LPR command. The filter option specifies the type of processing to be done with the data.

System Action: The LPR command is not processed.

User or Operator Response: Reenter the LPR command using the filter option with a valid filter. See *z/OS Communications Server: IP User's Guide and Commands* for more information on the LPR command.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPR

Procedure Name: ProcessOptions

EZB0977E Use a number after the INDENT option.

Explanation: A numeric argument was not specified for the INDENT option of the LPR command.

System Action: LPR command is not processed.

User or Operator Response: Reenter the LPR command, specifying a numeric argument for the INDENT option. For more information about the LPR command and the INDENT option, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: ProcessOptions

EZB0978E Use a token after the JOB option.

Explanation: You must specify a name when using the JOB option. The name is the printing job's description to the remote system.

System Action: The LPR command is not processed.

User or Operator Response: Reenter the LPR command with a name following the JOB option. For more information on the JOB option of the LPR command see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: ProcessOptions

EZB0979E Use a number after the LINECOUNT option.

Explanation: You must enter a number following the LINECOUNT option to specify the number of lines to be printed before a new heading is printed.

System Action: The LPR command is not processed.

User or Operator Response: Reenter the LPR command with the LINECOUNT option using a valid number. For more information see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: ProcessOptions

EZB0980E Use an identification after NAME option.

Explanation: You did not specify a name after the NAME option in the LPR command. (The NAME option specifies the job information to be provided by the remote system in response to a query.)

System Action: The LPR command is not processed.

User or Operator Response: Resubmit the LPR command supplying a name after the NAME option. For more information about the LPR command and the NAME option, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: ProcessOptions

EZB0981E Use a title after the TITLE option.

Explanation: You did not specify the title after the TITLE option in the LPR command.

System Action: The LPR command is not processed.

User or Operator Response: Resubmit the LPR command supplying a name after the TITLE option. For more information about the LPR command and the TITLE option, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: ProcessOptions

EZB0982E Use a number after the TOPMARGIN option.

Explanation: You did not specify a number after the TOPMARGIN option in the LPR command. (This number specifies the number of lines designated for the top margin).

System Action: The LPR command is not processed.

User or Operator Response: Resubmit the LPR command supplying the number of lines to be designated for the top margin. For more information about the LPR command and the TITLE option, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: ProcessOptions

EZB0983E Use a title after the WIDTH option.

Explanation: You did not specify the line width after the WIDTH option in the LPR command.

System Action: The LPR command is not processed.

User or Operator Response: Resubmit the LPR command supplying the line width of the data set. For more information about the LPR command and the WIDTH option see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: ProcessOptions

EZB0984E Use either the NOPOSTSCRIPT or POSTSCRIPT option but not both.

Explanation: You selected both the POSTSCRIPT and the NOPOSTSCRIPT options in the LPR command. You can select only one of these options.

System Action: The LPR command is not processed.

User or Operator Response: Reissue the command with the correct option. For more information about the LPR command and the POSTSCRIPT or NOPOSTSCRIPT options, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: ProcessOptions

EZB0985E Use either the POSTSCRIPT or LANDSCAPE option but not both.

Explanation: You selected both the POSTSCRIPT and LANDSCAPE options in the LPR command. You can select only one of these options.

System Action: The LPR command is not processed.

User or Operator Response: Reissue the command with the correct option. For more information about the LPR command and the POSTSCRIPT or NOPOSTSCRIPT options, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: ProcessOptions

EZB0986E Use either the BINARY OR LANDSCAPE option but not both.

Explanation: You selected both the BINARY and LANDSCAPE options in the LPR command. You may select only one of these options. If you select the LANDSCAPE option, the only valid filter option that you can specify explicitly is the "o" filter.

System Action: The LPR command is not processed.

User or Operator Response: Reissue the command using the correct option. For more information about the LPR command and the BINARY or LANDSCAPE options, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: ProcessOptions

EZB0987E **Use either the FILTER or LANDSCAPE option but not both.**

Explanation: You selected both the FILTER and LANDSCAPE options in the LPR command. You may select only one of these options. If you select the LANDSCAPE option, the only valid filter option that you can specify explicitly is the "o" filter.

System Action: The LPR command is not processed.

User or Operator Response: Reissue the command with the correct option. For more information about the LPR command and the FILTER or LANDSCAPE options, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: ProcessOptions

EZB0988I **PostScript program is *number* bytes.**

Explanation: This message specifies the number of bytes in the PostScript program.

System Action: The LPR command is not processed.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: ProcessOptions

EZB0989I **Use these options.**

EZB0990I *additional options*

EZB0991I *additional options or last options*

Explanation: TCP/IP issues this message when one or more incorrect options is specified in the LPR command. It provides a list of valid options.

System Action: The LPR command is not processed.

User or Operator Response: Reissue the LPR command using valid options.

System Programmer Response: None.

Source Data Set: LPR, LPRSET

Procedure Name: ProcessOptions

EZB0992I **File contains PostScript.**

Explanation: TCP/IP issues this message when tracing is on and PostScript characters are detected in the data set.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendControlFile

EZB0993E Use the program which produced the PostScript to change to landscape display.

Explanation: The LANDSCAPE option is specified in the LPR command and the file to be printed is a PostScript data set.

System Action: The LPR command is not processed.

User or Operator Response: Redefine the data set for landscape display using the original application program that created the POSTSCRIPT file. Then reissue the LPR command without the LANDSCAPE option. See also EZB1010E.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendControlFile

EZB0994E Use the NOPOSTSCRIPT option to prevent special processing of PostScript files.

Explanation: The LPR command contains both POSTSCRIPT and CC options.

System Action: The LPR command is not processed.

User or Operator Response: Reissue the LPR command using the NOPOSTSCRIPT option in place of the POSTSCRIPT option. For more information about the LPR command and the POSTSCRIPT and NOPOSTSCRIPT options, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendControlFile

EZB0995W Ignoring "buffer"

Explanation: TCPIP issues this message when an incorrect carriage control character is detected. The lines containing those incorrect characters are ignored.

System Action: Processing continues.

User or Operator Response: Correct the carriage control characters in the data set.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendControlFile

EZB0996W *number* lines with invalid carriage control characters deleted.

Explanation: TCPIP issues this message when incorrect carriage control characters are detected in the data set being processed. The lines containing the incorrect carriage control characters are deleted.

System Action: Processing continues.

User or Operator Response: Correct the carriage control characters in the data set and resubmit the job.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendControlFile

EZB0997I Byte size check starts at *date time*

Explanation: TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. The date and time are recorded before performing a byte size check of the data set.

System Action: Processing continues.

User or Operator Response: None.

EZB0998I • EZB0999I

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB0998I **Byte size check ends at** *date time*

Explanation: TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. The date and time are recorded after performing a byte size check of the data set.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB0999I **Send command starts at** *date time*

Explanation: TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. Arguments are passed prior to the actual data transfer. The date and time are recorded before those arguments are passed.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendDataFile

Chapter 2. EZB1xxxx messages

EZB1000I **Send command ends at** *date time*

Explanation: TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. After arguments are passed and acknowledgments are received, the date and time are recorded.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB1001I **Send data starts at** *date time*

Explanation: TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. Before the data set is sent to the remote printer, the date and time are recorded.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB1002I **Send data ends at** *date time*

Explanation: TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. When the data set is received by the remote printer, the date and time are recorded.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB1003I **Send ACK starts at** *date time*

Explanation: TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. TCPIP initiates an acknowledgment process. The date and time are recorded.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB1004I Send ACK ends at *date time*

Explanation: TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. When the requested acknowledgments are received, the acknowledgment process is complete; the date and time are recorded.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB1005I Draining the connection.

Explanation: This message occurs while the data is sent from the host to the remote printer and tracing is on. Before the connection between the host and the remote printer closes, the information is taken in and processed by the printer.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: DrainConnection

EZB1006E Host *host* did not accept printer name *printer*

Explanation: The indicated host did not recognize the indicated printer.

System Action: The LPR command is not processed.

User or Operator Response: Resubmit the job using the correct host printer name.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: DrainConnection

EZB1007I Connection still receiving - aborting it.

Explanation: TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. The remote host continues to receive information. When all the information is received, the connection is aborted.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: DrainConnection

EZB1008I Connection aborted.

Explanation: TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. TCPIP ended the connection with the remote host.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: DrainConnection

EZB1009I Data file sent.

Explanation: TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. The data set has been successfully sent.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB1010E Use the program which produced the PostScript to control pagination.

Explanation: This message follows EZB0993E and is issued when the LANDSCAPE option is specified in the LPR command and the file to be printed is a PostScript data set.

System Action: The LPR command is not processed.

User or Operator Response: Redefine the data set to control pagination using the original application program that created the POSTSCRIPT file. Then reissue the LPR command without the LANDSCAPE option.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendControlFile

EZB1011I Queuing control line *"number"*

Explanation: TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. The commands are retrieved in the order in which they were submitted.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: QueueControlLine

EZB1012I Receiving ACK

Explanation: TCPIP issues this message when tracing is on and a data set is sent to a remote printer using the LPR command. TCPIP is receiving acknowledgments from the remote host.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: ReceiveByte

EZB1013I ReceiveACK: *word* for byte value *rc*

Explanation: TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. TCPIP is receiving acknowledgments from the remote host.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: ReceiveByte

EZB1014I Sending ACK

Explanation: TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. The remote host is sending acknowledgments to TCPIP.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendACK

EZB1015I ACK successfully sent

Explanation: TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. The acknowledgment process has been successfully completed.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendACK

EZB1016E Did not receive an ACK for command. Code=*rc*

Explanation: TCPIP issues this message when tracing is on and a data set is sent to the remote printer using the LPR command. TCPIP did not receive an acknowledgment from the remote host. The return code is displayed.

System Action: The LPR command is not processed.

User or Operator Response: Resubmit the LPR command.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendCommand

EZB1017I Control data sent

Explanation: A control data set was sent to the remote host. The control data set contains the printer setup information.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendControlFile

EZB1018I Control file sent

Explanation: A control data set was sent to the remote host. The control data set contains the information to be printed.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendControlFile

EZB1019E Use the form: LPRSET nameofPrinter@printerhost.

Explanation: You specified an incorrect format for the LPRSET command.

System Action: The LPRSET command is not processed.

User or Operator Response: Reissue the LPRSET command with the correct syntax (nameofPrinter@printerhost). For more information about the LPRSET command, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: None.

Source Data Set: LPR, LPRSET

Procedure Name: ProcessOperands

EZB1020I Your LPR printer is currently set to *printer* at *host*

Explanation: This message is displayed when the LPRSET (QUERY) command is issued. The LPR printer name and host are identified.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPRSET

Procedure Name: ProcessQueryOption

EZB1021I Append of host name to LASTING.GLOBALV status *rc*

Explanation: This message is displayed when the LPRSET command is issued while tracing is on. The remote host name is added to the LASTING.GLOBALV data set. This message is followed by EZB1022I and EZB1023I.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPRSET

Procedure Name: ProcessArguments

EZB1022I Append of printer name to LASTING.GLOBALV status *rc*

Explanation: This message is displayed when the LPRSET command is issued while tracing is on. The printer name is added to the LASTING.GLOBALV data set. This message appears with EZB1021I and EZB1023I.

System Action: Processing continues.

User or Operator Response: None.

EZB1023I • EZB1026E

System Programmer Response: None.

Source Data Set: LPRSET

Procedure Name: ProcessArguments

EZB1023I Printer set to printer at host

Explanation: This message is displayed when the LPRSET command is issued while tracing is on. The LPR printer name and host are identified. This message is preceded by EZB1021I and EZB1022I.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPRSET

Procedure Name: ProcessArguments

EZB1024E System error error setting printer.

Explanation: This message occurs due to a failure while attempting to set the LPR printer. For more information about the LPRSET command, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Action: The LPRSET command is not processed.

User or Operator Response: Check the remote printer and resubmit the job.

System Programmer Response: None.

Source Data Set: LPRSET

Procedure Name: ProcessArguments

EZB1025E Cannot use BINARY with a DBCS Translation Mode.

Explanation: The BINARY parameter may not be specified with any other of the following LPR command line parameters:

- JIS78KJ
- JIS83KJ
- SJISKANJI
- EUCKANJI
- IBMKANJI
- HANGEUL
- KSC5601
- TCHINESE

System Action: The LPR command is not processed.

User or Operator Response: Consult your system programmer.

System Programmer Response: Specify either the BINARY parameter or a DBCS translation mode parameter. See the *z/OS Communications Server: IP User's Guide and Commands* for more information about using DBCS conversion in LPR.

Source Data Set: LPR

Procedure Name: ProcessDbcsOption

EZB1026E Cannot use multiple DBCS Translation Modes.

Explanation: More than one of the following LPR command line parameters was specified:

- JIS78KJ
- JIS83KJ
- SJISKANJI
- EUCKANJI

- IBMKANJI
- HANGEUL
- KSC5601
- TCHINESE

System Action: The LPR command is not processed.

User or Operator Response: Consult your system programmer.

System Programmer Response: Specify only one of the above DBCS translation mode parameters. See the *z/OS Communications Server: IP User's Guide and Commands* for more information about using DBCS conversion in LPR.

Source Data Set: LPR

Procedure Name: ProcessDbcsOption

EZB1027E Unable to Load DBCS_translate_table_dataset

Explanation: LPR was attempting to load a DBCS translation table corresponding to the DBCS conversion mode parameter specified on the LPR command line. All data sets in the search order hierarchy for the required translate table dataset, either do not exist, or do not contain data in the required format for DBCS binary translate tables.

System Action: The LPR command is not processed.

User or Operator Response: Consult your system programmer.

System Programmer Response: Configure a valid DBCS binary translate table data set in the search order hierarchy for the required DBCS translation table. Refer to the *z/OS Communications Server: IP User's Guide and Commands* for more information about the loading and customizing of DBCS translation tables.

Source Data Set: LPR

Procedure Name: LoadDbcsTables

EZB1043E Failed to Send PostScript_id. Return code = rc. Error Number= errno.

Explanation: This message is issued as a result of the LPD server failing to send the PostScript id contained within the PostScript data file.

errno is the UNIX System Services Return Code. Return Codes are listed and described in the *z/OS UNIX System Services Messages and Codes*.

System Action: LPR halts.

User or Operator Response: Notify the system programmer.

System Programmer Response: See the *z/OS Communications Server: IP and SNA Codes* for information about TCP/IP return codes and the *z/OS UNIX System Services Messages and Codes* for information about the *errno*.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB1044E Specify a user name after USER option.

Explanation: The LPR command was entered with no user name specified for the USER option.

System Action: LPR halts.

User or Operator Response: Reenter the LPR command, specifying a user name after the USER option. For more information, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: ProcessOptions

EZB1045E Send ACK failed. Return Code = *rc*. Error Number = *errno*.

Explanation: The client machine failed while attempting to send acknowledgments to the server machine. The current port number and the user IP address are displayed with this message.

errno is the UNIX System Services Return Code. Return Codes are listed and described in the *z/OS UNIX System Services Messages and Codes*.

System Action: LPR halts.

User or Operator Response: See the *z/OS Communications Server: IP and SNA Codes* for information about TCP/IP return codes and the *z/OS UNIX System Services Messages and Codes* for information about the *errno*.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB1046E Shutdown failed. Return Code = *rc*. Error Number = *errno*.

Explanation: While attempting to close the TCP connections on both the client and server sides, an error was detected. The current port number and the user IP address are displayed with this message.

errno is the UNIX System Services Return Code. Return Codes are listed and described in the *z/OS UNIX System Services Messages and Codes*.

System Action: LPR halts.

User or Operator Response: See the *z/OS Communications Server: IP and SNA Codes* for information about TCP/IP return codes and the *z/OS UNIX System Services Messages and Codes* for information about the *errno*.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB1047E Send Control File failed. Return Code = *rc*. Error Number = *errno*.

Explanation: An error was detected while attempting to send the file format information. The current port number and the user IP address are displayed with this message.

errno is the UNIX System Services Return Code. Return Codes are listed and described in the *z/OS UNIX System Services Messages and Codes*.

System Action: LPR halts.

User or Operator Response: See the *z/OS Communications Server: IP and SNA Codes* for information about TCP/IP return codes and the *z/OS UNIX System Services Messages and Codes* for information about the *errno*.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB1048E Send Data File failed. Return Code = *rc*. Error Number = *errno*.

Explanation: An error was detected while attempting to send the data file containing the file data. The current port number and the user IP address are displayed with this message.

errno is the UNIX System Services Return Code. Return Codes are listed and described in the *z/OS UNIX System Services Messages and Codes*.

System Action: LPR halts.

User or Operator Response: See the *z/OS Communications Server: IP and SNA Codes* for information about TCP/IP return codes and the *z/OS UNIX System Services Messages and Codes* for information about the *errno*.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB1049E Send printer command did not receive ACK. ACK message = *message*.

Explanation: No acknowledgment was received from the server while attempting to send a printer command. An acknowledgment message was passed. The current port number and the user IP address are displayed with this message.

System Action: LPR halts.

User or Operator Response: Use the ACK message to determine and correct the problem.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB1050E Failed to Send Data. Return Code = *rc*. Error Number = *errno* port number = *port* remote ip address = *ipaddr*

Explanation: An error was detected while attempting to send data to the server machine.

rc is the return code from the subroutine that was called.

errno is the UNIX System Services Return Code. Return Codes are listed and described in the *z/OS UNIX System Services Messages and Codes*.

port is the number of the connecting port.

ipaddr is the remote IP address.

System Action: LPR halts.

User or Operator Response: If *rc* is -1 and *errno* is 0, then there was a read error while trying to access the DBCS translation tables. Check your translation tables and their location. Correct any problems with the DBCS translation tables and retry the LPR command.

If *rc* is -2 and *errno* is 0, then there are DBCS characters that are not valid in the print file that you are trying to send. Turn on the trace option in LPR to see more messages from the translation routines. Correct the print file and retry the LPR command.

If *rc* is -1 and *errno* is nonzero, then look up the *errno* in the *z/OS UNIX System Services Messages and Codes* for help in diagnosing the exact problem.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB1051E Failed to Open connection to Port Number = *port*. return code = *rc* error number = *errno* port number = *port* remote ip address = *ipaddr*

Explanation: The attempt to open a connection at the stated port was unsuccessful. The requested port, return code, error number, current port, and the IP address are displayed with this message.

errno is the UNIX System Services Return Code. Return Codes are listed and described in the *z/OS UNIX System Services Messages and Codes*.

System Action: LPR halts.

User or Operator Response: Try to access the stated port number again. If the problem persists, notify the system programmer.

System Programmer Response: See the *z/OS Communications Server: IP and SNA Codes* for information about TCP/IP return codes and the *z/OS UNIX System Services Messages and Codes* for information about the *errno*.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB1052E StartTcpip: Return Code = rc and Error Number = errno.

Explanation: An error was detected while attempting to initialize TCPIP.

errno is the UNIX System Services Return Code. Return Codes are listed and described in the *z/OS UNIX System Services Messages and Codes*.

System Action: LPR halts.

User or Operator Response: See the *z/OS Communications Server: IP and SNA Codes* for information about TCP/IP return codes and the *z/OS UNIX System Services Messages and Codes* for information about the *errno*.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB1053E Could not set option. Return Code = rc and Error Number = errno.

Explanation: While attempting to set the socket KeepAlive option, an error was detected.

errno is the UNIX System Services Return Code. Return Codes are listed and described in the *z/OS UNIX System Services Messages and Codes*.

System Action: LPR halts.

User or Operator Response: See the *z/OS Communications Server: IP and SNA Codes* for information about TCP/IP return codes and the *z/OS UNIX System Services Messages and Codes* for information about the *errno*.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB1054E TcpClose: Return Code = rc and Error Number = errno.

Explanation: While attempting to close the TCP connection on both the client and server machines an error was detected. The current port number and the user IP address are displayed with this message.

errno is the UNIX System Services Return Code. Return Codes are listed and described in the *z/OS UNIX System Services Messages and Codes*.

System Action: LPR halts.

User or Operator Response: See the *z/OS Communications Server: IP and SNA Codes* for information about TCP/IP return codes and the *z/OS UNIX System Services Messages and Codes* for information about the *errno*.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB1055E Use a three-digit number after the JNUM option.

Explanation: The value specified for JNUM is not valid. It is either missing, greater than or less than three digits, or is not numeric. The JNUM value may be any three-digit number between 000 and 999. The print job is not processed.

System Action: LPR halts.

User or Operator Response: Correct the JNUM value specified on the LPR command.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPR

Procedure Name: ProcessOptions

EZB1056E **Invalid Internet address** *Foreignhost.*

Explanation: You provided an address containing numbers greater than 255. This is an invalid Internet address for LPR.

System Action: LPR halts.

User or Operator Response: Check the Internet address to which you are sending your output.

System Programmer Response: Assist the user as necessary.

Source Data Set: LPR

Procedure Name: MainLoop

EZB1057I **Loaded translation table from** *data set name*

Explanation: The fully qualified data set name used by LPR for the translation tables is displayed. These tables are used to translate EBCDIC to ASCII and ASCII to EBCDIC. This message may be issued more than once if more than one table is loaded. The last table loaded of a particular type will be the one used. (This will be the one cited in the last occurrence of this message).

System Action: Processing continues.

User or Operator Response: If the data set name displayed is not the one desired, verify that the expected data set exists. The search order used by LPR to find the translation data set is described in the *z/OS Communications Server: IP Configuration Reference*. If the data set name displayed is empty (" "), then no translate data set was found in the search order. Create the desired translate table, or correct the TRANSLATE option specified on the LPR command.

System Programmer Response: Verify that the system translate tables are correctly installed under the correct high level qualifier (e.g. *hlq.STANDARD.TCPXLBIN*). For information on determining the high level qualifier used, refer to the *z/OS Communications Server: IP Configuration Reference*.

Source Data Set: LPR

Procedure Name: ProcessOptions

EZB1098I **Return Code = rc. Error Number =** *errno.*

Explanation: See the user response. The text of this message is usually appended to the end of other messages.

errno is the UNIX System Services Return Code. Return Codes are listed and described in the *z/OS UNIX System Services Messages and Codes*.

System Action: Depending on the return code and error number in this message, LPR process may continue or may terminate.

User or Operator Response: See the *z/OS Communications Server: IP and SNA Codes* for information about TCP/IP return codes and the *z/OS UNIX System Services Messages and Codes* for information about the *errno*. If the return code is -1 and error number is 0, then negative acknowledgment was received from the server. Turn on the trace for the server to find out why the server rejected the LPR request.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB1099I **Port Number =** *port. Remote IP Addr =* *target address*

Explanation: See the user response. The text of this message is usually appended to the end of other messages. This message displays the port number and the IP address currently in use.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: LPR

Procedure Name: SendDataFile

EZB1100I Cannot load translate table with name - *name* using defaults

Explanation: The Line Printer Request (LPR) attempted to load an SBCS translation table corresponding to the TRANSLATETABLE or XLATETABLE parameter specified on the LPR command line. All data sets in the search order hierarchy for the required translate table data set, either do not exist, or do not contain data in the required format for SBCS binary translate tables.

name is an input string that becomes part of the translation table data set name (for example, *hlq.name.TCPXLBIN*).

System Action: LPR continues, however, default hardcoded tables are used for translations.

User or Operator Response: Notify your system programmer.

System Programmer Response: Configure a valid SBCS binary translate table data set in the search order hierarchy for the required SBCS translation table. See the *z/OS Communications Server: IP Configuration Reference* for more information about using translation tables, including search order hierarchy and customization.

Source Data Set: LPR

Procedure Name: ProcessOptions

EZB1200E Error parsing argument "*option*" (*specifier*) ; unknown kind

Explanation: XWindows has encountered an option or specifier that it does not recognize in a user-submitted command line. The command line is not processed.

System Action: XWindows continues.

User or Operator Response: Correct the syntax and resubmit the command.

System Programmer Response: None.

Source Data Set: PARSECMD

Procedure Name: _XReportParseError

EZB1201E Bad image *image_type*: error found in routine: *routine*

Explanation: Xwindows received an incorrect image type from the indicated routine. The image type and Xwindows error code are displayed in the message.

System Action: Xwindows halts.

User or Operator Response: Notify the system programmer of the error.

System Programmer Response: Use the error code given in the message and *z/OS Communications Server: IP and SNA Codes* to determine the cause of the error, and correct the indicated routine as necessary.

Source Data Set: XIMUTIL

Procedure Name: _XReportBadImage

EZB1202E Xlib: warning, client built for newer rev (*rev_number*) than server (*rev_number*)!

Explanation: The Xwindows client code was built for a newer version of Xwindows than the Xwindows server code. This can result in compatibility errors.

System Action: Xwindows halts.

User or Operator Response: Notify the system programmer of the error.

System Programmer Response: Correct either the client code or the server code to eliminate the revision disparity and rebuild the code.

Source Data Set: XOPENDIS

Procedure Name: _XRead

EZB1203E **Xlib: warning, client is protocol rev *revision*, server is rev *revision*!**

Explanation: The Xwindows server is using a different revision of the Xwindows code than the client. This can cause compatibility problems.

System Action: Xwindows halts.

User or Operator Response: Notify the system programmer of the error.

System Programmer Response: Correct the disparity in revision levels between the client and the server and restart Xwindows.

Source Data Set: XOPENDIS

Procedure Name: _XRead

EZB1204E **Xlib: connection to *server* refused by server**

Explanation: This message indicates that no authorization exists between the Xwindow server and the client program. The error causes the server to refuse the connection. The connection is not established.

System Action: No Xwindows session can be opened for this client.

User or Operator Response: Issue an Xhost command from the Xserver side.

System Programmer Response: None.

Source Data Set: XOPENDIS

Procedure Name: Main

EZB1205E **Xlib: sequence lost (0x*number_received* > 0x*number_expected*) in reply type 0x*reply_type*!**

Explanation: The Xwindows server received a reply packet with a sequence number greater than the sequence number expected, indicating that a reply packet has been lost.

System Action: Xwindows halts processing.

User or Operator Response: Notify the system programmer of the error.

System Programmer Response: Check the indicated Xclient to determine why the packet was lost and respond as indicated.

Source Data Set: XLIBINT

Procedure Name: _XSetLastRequestRead

EZB1206E **Xlib: unhandled wire event! event number = *number*, display = *display_number***

Explanation: Xwindows encountered an unknown event during conversion between the host format and the wire format.

System Action: Xwindows halts.

User or Operator Response: Notify the system programmer of the error.

System Programmer Response: Check the XEvent structure to make sure it is compatible with the current host. For more information, refer to the Xwindows documentation.

Source Data Set: XLIBINT

Procedure Name: _XUnknownWireEvent

EZB1207E **Xlib: unhandled native event! event number = *number*, display = *display_number***

Explanation: Xwindows encountered an unknown event while reformatting a wire event to the host structure.

System Action: Xwindows halts.

User or Operator Response: Notify the system programmer of the error.

EZB1208E • EZB1212E

System Programmer Response: Check the XEvent structure to make sure that it is compatible with the current host. For more information refer to the Xwindows documentation.

Source Data Set: XLIBINT

Procedure Name: _XUnknownNativeEvent

EZB1208E **XIO: fatal IO error** *error_description* **on X server** *address_of_server*

EZB1209E **after** *number* **requests (number known processed) with** *number* **events remaining.** *err_no* **(error) on X server** *server*

Explanation: The Xwindows server encountered an I/O error during processing. The type of error is indicated in the message. The error description portion of this message indicates the cause of the error.

System Action: Xwindows halts.

User or Operator Response: Notify the system programmer of the error.

System Programmer Response: Respond as indicated by the *error_description* portion of this message.

Source Data Set: XLIBINT

Procedure Name: _XDefaultIOError

EZB1210I **The connection was probably broken by a server shutdown or KillClient.**

Explanation: If this message follows messages EZB1208E and EZB1209E, it indicates that the error displayed in those messages was probably caused by the shutdown of the Xwindows server, or by the procedure KillClient, which closes a connection to an Xwindows client.

System Action: Xwindows halts.

User or Operator Response: Notify the system programmer of the error.

System Programmer Response: Check the Xwindows server to determine why it shut down or closed the client connection.

Source Data Set: XLIBINT

Procedure Name: _XDefaultIOError

EZB1211I **Request failed due to IUCV error.**

Explanation: This message follows messages EZB1208E and EZB1209E, and indicates that the error displayed in those messages was caused by a failure in the inter-user communication vehicle (IUCV). Xwindows uses IUCV to transmit requests and receive replies.

System Action: Xwindows halts.

User or Operator Response: Notify the system programmer of the error.

System Programmer Response: See additional messages to determine the cause of the IUCV error and respond as indicated.

Source Data Set: XLIBINT

Procedure Name: _XDefaultIOError

EZB1212E **Xlib: extension** *extension* **reason on display** *display*.

Explanation: Xwindows encountered an incorrect event handler when the function XSetExtensionErrorHandler or XMissingExtension is called.

System Action: Xwindows halts.

User or Operator Response: Notify the system programmer of the error.

System Programmer Response: Make sure that the event handler is set correctly before calling the associated

extension. For more information refer to the Xwindows documentation.

Source Data Set: EXTUTIL

Procedure Name: XExtDisplayInfo

EZB1220I *name: argument, value: 0xhex_value*

Explanation: This message displays the name and hexadecimal value of arguments for Xwindows processing.

System Action: Xwindows continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: ARGLIST

Procedure Name: PrintArgList

EZB1221E *error_prefix***Error:** *reason*

Explanation: Xwindows has encountered an error. The reason for the error is described in the *reason* portion of this message.

System Action: Xwindows halts.

User or Operator Response: Notify the system programmer of the error.

System Programmer Response: Respond as indicated by the *reason* portion of this message.

Source Data Set: ERROR

Procedure Name: _XtDefaultError

EZB1222E *warning_prefix* **Warning:** *reason*

Explanation: Xwindows has encountered an error. The reason for the error is described in the *reason* portion of this message.

System Action: Xwindows halts.

User or Operator Response: Notify the system programmer of the error.

System Programmer Response: Respond as indicated by the *reason* portion of this message.

Source Data Set: ERROR

Procedure Name: _XtDefaultWarning

EZB1230I *header w: children, m: request_mode, x: x_coordinate, y: y_coordinate, w: width, h: height, b: border_width*

Explanation: This message displays information about the geometry of the current Xwindows.

System Action: Xwindows continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: GEOUTILS

Procedure Name: PrintBox

Chapter 3. EZB2xxxx messages

EZB2000I *string string hexdata*

Explanation: The contents of a packet or control block are displayed.

System Action: The X25 server continues processing.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: SNAPIA5, SNAPAREA

EZB2010I *program {MVS} update level level*

Explanation: The version of X25IPI currently running has the indicated program name (usually X25IPI) and the indicated update level.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: Use the update level when reporting errors.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2011I **Command:** *command*

Explanation: This message is displayed when tracing is on. The X25 server is processing the command passed to it from the console input program.

System Action: The command is processed and execution continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: COMMAND

EZB2012E **Unrecognized command:** *command*

Explanation: The X25 server tried to process an unrecognized command that was passed to it from the console input program.

System Action: The X25 server continues processing.

User or Operator Response: Correct the command and resubmit. Refer to *z/OS Communications Server: IP Configuration Reference* for more information about TCPIP X25 commands.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: COMMAND

EZB2013T Unrecognized CIB verb= *cibverb*

Explanation: The X25IPI server received an unexpected command verb in a command input buffer from the command queue (QEDIT). The valid commands are STOP, START, and MODIFY.

System Action: X25IPI treats the command as STOP and exits.

User or Operator Response: Restart TCPIPX25.

System Programmer Response: Report the problem to the IBM Software Support Center.

Source Data Set: XNX25IPI

Procedure Name: COMMAND

EZB2020I MCH *lu state state session_state*

Explanation: X25IPI status for the indicated MCH is displayed. The following are the *Link_state* fields:

Field	Description
X'00'	Restart needed
X'10'	Ready
X'20'	Restart request sent
X'30'	Restart indication received

The following are the VTAM*session_state* fields:

Field	Description
X'20'	Enabled for LOGAPPL logon
X'30'	logon pending
X'40'	Opening
X'50'	Open (ready)
X'60'	Closing
X'70'	Unsuccessful

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: Use the status in problem determination.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2021I VC *vc LU lu DTE address state call_state session_state*

Explanation: X25IPI status for the indicated virtual circuit is displayed. The following are the *call_state* fields:

Field	Description
X'00'	Waiting for restart on MCH
X'10'	Available for connections
X'20'	Call request sent
X'30'	Call request received
X'40'	Ready for data transfer
X'41'	Reset request sent
X'42'	Reset request received
X'43'	Interrupt request sent
X'44'	Interrupt request received
X'50'	Call collision
X'60'	Clear request sent
X'70'	Clear request received

The following are the VTAM*session_state* fields:

Field	Description
X'00'	Disabled until restart on MCH

X'10' Available for connections
 X'20' Enabled for NPSI LU logon
 X'30' Logon pending
 X'40' Opening
 X'50' Open (ready)
 X'60' Closing
 X'70' Failed

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: Use the status in problem determination.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2022R IP *AS_*asname **state** *connection status*

Explanation: X25IPI status for the DLC path to the TCPIP address space is displayed. The following are the state codes:

X'80' DLC connection complete
 X'40' DLC connection pending
 X'00' no path

System Action: The X25 server continues processing.

User or Operator Response: Use the status in problem determination.

System Programmer Response: Use the status in problem determination.

Source Data Set: XNX25IPI

Procedure Name: CLIST

EZB2030I MCH *lu* **RC:** *name* **IP:** *name* **SN:** *name* **TX:** *name*

Explanation: In response to an EVENTS command, the internal subroutine names handling events associated with this MCH are displayed.

Subroutine

	Description
IP	VTAM input pending
RC	VTAM request complete
SN	VTAM session notification
TX	Timer expired

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: Use the data for problem determination.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2031I VC *vc* **RC:** *name* **IP:** *name* **SN:** *name* **TX:** *name*

Explanation: In response to an EVENTS command, the internal subroutine names handling events associated with this virtual circuit is displayed.

Subroutine

	Description
RC	VTAM request complete
IP	VTAM input pending
SN	VTAM session notification
TX	Timer expired

EZB2032I • EZB2040E

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: Use the data for problem determination.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2032I **MCH** *lu*

Explanation: The following X25IPI014R VC messages apply to virtual circuits on this MCH.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2033I **VC** *vc DTE_address S send_count R receive_count D drop_count Q queue_size*

Explanation: The traffic counts for this virtual circuit are displayed.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2034I **IP** *AS_userid S send_count R receive_count D drop_count Q queue_size*

Explanation: The traffic counts on the IUCV connection to the TCPIP address space are displayed.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2040E **MCH** *lu unknown*

Explanation: An attempt was made to send a call request for an unknown MCH. The request is ignored. The CERTCALL command is refused.

System Action: TCPIP continues.

User or Operator Response: Determine the correct MCH name, and reissue the CERTCALL command.

System Programmer Response: Assist the user as necessary.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2041I MCH *lu* restarting

Explanation: X25IPI is restarting this MCH in response to a RESTART command. A VTAM session is initiated for the NPSI MCH LU.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2042E MCH *lu* unavailable

Explanation: X25IPI could not restart this MCH because it is unavailable for logon. The MCH session is left available for initiation by LOGAPPL.

System Action: TCPIP continues.

User or Operator Response: Activate the MCH LU through VTAM.

System Programmer Response: Determine the reason why VTAM refused session initiation.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2043E MCH *lu* already started

Explanation: X25IPI could not restart this MCH because it is already active. The request is ignored.

System Action: TCPIP continues.

User or Operator Response: Notify the system programmer of the error.

System Programmer Response: Determine the correct MCH name and reissue the CERTCALL command.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2050I *object at address additional*

Explanation: This message reports the locations of internal storage as requested by debug flag 4. The *object* can be "Global storage", "MCH *lu* SDA", or "VC *vc* SDA". The *additional* variable, if present, reports the "stack at *address* limit *address*".

System Action: The X25 server continues processing.

User or Operator Response: None.

System Programmer Response: Use the storage addresses for problem determination.

Source Data Set: XNX25IPI

Procedure Name: STRTLINK

EZB2051I MCH *lu* SDA at *address*

Explanation: The contents of the session data area (SDA) for this connection are dumped.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: Use the data for problem determination.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2080I **Dispatch** *handler name (event handler address)* **SDA** *session data area address* **ECB** *event control block*

Explanation: This message is displayed when tracing is on. The X25 server scans for posted event control blocks and calls event handlers.

System Action: The X25 server continues processing.

User or Operator Response: None.

System Programmer Response: Use the dispatch trace in problem determination.

Source Data Set: XNX25IPI

Procedure Name: MAINLOOP

EZB2081I **Dispatch** *handler name (event handler address)* **SDA** *session data area address*

Explanation: This message is displayed when tracing is on. The X25 server scans for posted event control blocks and calls an MCH event handler.

System Action: The event handler is called.

User or Operator Response: None.

System Programmer Response: Use the dispatch trace in problem determination.

Source Data Set: XNX25IPI

Procedure Name: MCHEVENT

EZB2082I **Main wait:** *number of ECBs* **ECBs**

Explanation: This debug message displays the number of event control blocks (ECB) in the main wait list.

System Action: The X25 server continues processing.

User or Operator Response: None.

System Programmer Response: Use the event count in problem determination.

Source Data Set: XNX25IPI

Procedure Name: MAINLOOP

EZB2083I **Call** *caller's name (caller's address)* **from** *previous caller* **in** *previous caller's name (previous caller's address)*

Explanation: This debug message displays the name of the current subroutine and the calling subroutine.

System Action: The X25 server continues processing.

User or Operator Response: None.

System Programmer Response: Use the trace in problem determination.

Source Data Set: XNX25IPI

Procedure Name: FOLLOW

EZB2084E **Posted ECB at** *address* **had no handler**

Explanation: The X25IPI event control block at this address did not specify an event handler routine.

System Action: TCPIP continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: If the error recurs, obtain a dump of the X25IPI address space to make sure that the correct event handler is loaded.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2085E ECB address being added twice to wait list

Explanation: The X25IPI routine that adds elements to the ECB wait list detected an attempt to add an ECB twice.

System Action: TCPIP continues.

User or Operator Response: Notify the system programmer about the error

System Programmer Response: Determine the external event that caused this condition. Re-create it with an X25 internal trace running. Obtain a dump of the X25 region and submit the dump to the IBM Software Support Center.

Source Data Set: XNX25IPI

Procedure Name: None

EZB2090I Terminating

Explanation: X25IPI is terminating its execution and shutting down.

System Action: The VTAM ACB is closed.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2091I HALT notice accepted type *type*

Explanation: X25IPI received a VTAM HALT notification or a HALT console command and is shutting down execution.

System Action: VTAM LU sessions are closed, and the IUCV connection to the TCPIP address space is severed.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2092T Stack overflow at *address*

Explanation: X25IPI encountered a stack overflow at the indicated address. ABEND message X'091' is displayed.

System Action: X25IPI abends.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Determine the cause of overflow, and submit the ABEND dump to the IBM Software Support Center.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2093T Buffer released twice at *address* in routine (*address*)

Explanation: X25IPI encountered a consistency error in its buffer allocation pool. A buffer was released twice. This indicates a programming error. Return code X'92'. is displayed.

System Action: X25IPI abends.

User or Operator Response: Tell the system programmer about the error.

EZB2094S • EZB2100I

System Programmer Response: Determine the cause of the consistency error, and submit the ABEND dump to the IBM Software Support Center.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2094S NPSI SEND completion, pending packet = *packet number*

Explanation: A program flag indicating a deferred control packet to be sent has an unacceptable value. The pending condition is reset.

System Action: TCPIP continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2095T WTO text overflow

Explanation: XNX25IPI encountered an overflow in the Write To Operator (WTO) area, which is used to display informational and error messages. ABEND message X'099' is displayed.

System Action: The program abends.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Determine the cause of the overflow, and submit the ABEND dump to the IBM Software Support Center.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2099I Ended

Explanation: X25IPI has finished processing.

System Action: The X25IPI program exits.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2100I *configuration dataset record*

Explanation: This debug message displays a record that was read from the configuration data set.

System Action: The X25 server continues processing.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: X25IPI1

EZB2101E Unable to open configuration file, DDNAME=X25IPI

Explanation: X25IPI encountered an error opening its X25IPI configuration data set.

System Action: X25IPI does not start.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the DD for the X25IPI configuration data set.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2102E Unrecognized configuration entry: *keyword*

Explanation: X25IPI encountered an unrecognized entry in its X25IPI configuration data set. This configuration record is skipped; further configuration errors can result.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the entry in the X25IPI configuration data set.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2103E Missing configuration entry: *keyword*

Explanation: X25IPI could not find the indicated keyword in its configuration data set. You should have at least one entry defining a link in the file.

System Action: X25IPI does not start.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Add a Link definition to the X25IPI configuration data set for each indicated keyword to be used for TCPIP traffic.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2104E Buffer storage not available (*bytes* bytes required)

Explanation: X25IPI could not allocate sufficient storage during initialization.

System Action: X25IPI does not start.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Increase the X25IPI region or address space storage size to the indicated number of bytes.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2106E DLC Init function X25IPI failed R15=*value*

Explanation: X25IPI encountered an error issuing a DLC Init call.

System Action: X25IPI does not start.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Check that the X25IPI procedure is in the PROCLIB.

Source Data Set: XNX25IPI

EZB2107T • EZB2112E

Procedure Name: None.

EZB2107T **Programming error: not enough buffers allocated**

Explanation: The calculation of the number of buffers to allocate was incorrect.

System Action: X25IPI does not start.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2108I **Unable to obtain TCPIPJobname from TCPIP.DATA, default TCPIP job name used**

Explanation: X25IPI encountered an error obtaining the TCP/IP stack name from the TCPIP.DATA file. The default job name of TCPIP is being used.

System Action: TCPIP continues.

User or Operator Response: If TCP connections cannot be made, notify the system programmer.

System Programmer Response: If the job name of the TCPIP stack to be used by X25 is not TCPIP use the TCPIP.DATA TCPIPJOBNAME statement to specify the correct name. See the IP Configuration Guide for how TCPIP.DATA statements are searched for.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2110E **Unrecognized trace level: *level***

Explanation: The trace level, specified in either the X25IPI configuration data set or the TRACE command, is not correct. The trace level is set to "off".

System Action: TCPIP continues.

User or Operator Response: Reissue the TRACE command with valid trace level.

System Programmer Response: Correct the Trace entry in the X25IPI configuration data set.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2111I **VTAM ACB *IPI_APPN* opened successfully**

Explanation: The virtual transmission access method (VTAM) address space successfully opened an activity control block (ACB) for the indicated application.

System Action: The application begins processing. TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: xnx25ipi

Procedure Name: main

EZB2112E **Repeated VTAM record ignored: *record***

Explanation: X25IPI encountered more than one VTAM entry in the X25IPI configuration data set. The repeated VTAM record is skipped.

System Action: Initialization continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Remove the repeated VTAM record.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2113E VTAM application name missing

Explanation: The VTAM application ID is missing on the VTAM entry in the X25IPI configuration data set.

System Action: X25IPI does not start.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Specify the VTAM application ID and password on the VTAM entry.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2114E VTAM application password missing

Explanation: The VTAM application password is missing on the VTAM entry in the X25IPI configuration data set.

System Action: X25IPI does not start.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Specify the VTAM application password on the VTAM entry.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2115E VTAM ACB *application* open failed

Explanation: X25IPI encountered an error opening the VTAM ACB. This message is preceded by a EZB2401E or EZB2402E message reporting the VTAM error code.

System Action: X25IPI does not start.

User or Operator Response: Activate the X25IPI application in VTAM.

System Programmer Response: Use the VTAM error code from the message EZB2401E or EZB2402E to determine why the VTAM ACB OPEN macro was unsuccessful.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2121E DDN and non-DDN links cannot be mixed

Explanation: X25IPI encountered Link statements in its X25IPI configuration data set specifying both DDN and non-DDN links. The Link entry is discarded; further errors can result.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Separate DDN and non-DDN links into 2 X25IPI virtual machines.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2122E Link LU name missing

Explanation: The NPSI MCH LU name is missing on the Link entry in the X25IPI configuration data set. The Link entry is discarded; further errors can result.

System Action: Processing continues.

EZB2123E • EZB2126E

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Complete the Link entry. Refer to *z/OS Communications Server: IP Configuration Guide* for the format of the Link definition.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2123E Link DTE address missing

Explanation: X25IPI encountered a Link statement without a DTE address specified. This field is required for all non-DDN links. The Link entry is discarded; further errors can result.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Complete the Link entry. Refer to *z/OS Communications Server: IP Configuration Guide* for the format of the Link definition.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2124E Link window size missing or not numeric

Explanation: X25IPI encountered a Link statement in its X25IPI configuration data set that had a missing or nonnumeric default window size specification. The Link entry is discarded; further errors can result.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the window size. Refer to the *z/OS Communications Server: IP Configuration Reference* for the format of the Link definition.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2125E Link window size not in range 1..7 or 1..127

Explanation: X25IPI encountered a Link statement in its X25IPI configuration data set with an incorrect default window size specification. The window size should be in the range 1–7 or 1–127. The Link entry is discarded; further errors can result.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the window size.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2126E Link packet size missing or not numeric

Explanation: X25IPI encountered a Link statement in its X25IPI configuration data set that had a missing or nonnumeric default packet size. The Link entry is discarded; further errors can result.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the packet size. Refer to *z/OS Communications Server: IP Configuration Guide* for the format of the Link definition.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2127E Link packet size unacceptable

Explanation: X25IPI encountered a Link statement in its X25IPI configuration data set with an incorrect default packet size specification. The size should be 32, 64, 128, 256, 512, 1024, 2048, or 4096. The Link entry is discarded; further errors can result.

System Action: The link entry is discarded. Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the packet size.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2128E Link logical channel count missing or not numeric

Explanation: X25IPI encountered a Link statement in its X25IPI configuration data set that had a missing or nonnumeric logical channel count field. The Link entry is discarded; further errors can result.

System Action: The Link entry is discarded. Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the channel count. See *z/OS Communications Server: IP Configuration Guide* for the format of the Link definition.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2129E Link logical channel count not in range 1..1023

Explanation: X25IPI encountered a Link statement in its X25IPI configuration data set with an incorrect logical channel count field. The count should be in the range 1–1023. The Link entry is discarded; further errors can result.

System Action: The Link entry is discarded. Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the channel count.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2130E Link reserved channel count missing or not numeric

Explanation: X25IPI encountered a Link statement in its X25IPI configuration data set that had a missing or nonnumeric reserved channel count. The Link entry is discarded; further errors can result.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the reserved channel count. See *z/OS Communications Server: IP Configuration Guide* for the format of the Link definition.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2131E Link reserved channel count not in range 1..LCC

Explanation: X25IPI encountered a Link statement in its X25IPI configuration data set with an incorrect reserved channel count. The count should be between 1 and the number of logical channels that have been defined. The Link entry is discarded; further errors can result.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the reserved channel count. See *z/OS Communications Server: IP Configuration Guide* for the format of the Link definition.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2132E Link data network code missing

Explanation: X25IPI encountered a Link statement in its X25IPI configuration data set that omitted a data network identifier code (DNIC). DNICs should be specified for all networks (use DDN for DDN nets). The Link entry is discarded; further errors can result.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Complete the Link entry. See *z/OS Communications Server: IP Configuration Guide* for the format of the Link definition.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2133E Data network identifier code not decimal

Explanation: X25IPI encountered a Link statement in its X25IPI configuration data set that specified a nondecimal data network identifier code (DNIC). The DNIC should be decimal. The Link entry is discarded; further errors can result.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the DNIC number. See *z/OS Communications Server: IP Configuration Guide* for the format of the Link definition.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2134E Link DTE address not decimal

Explanation: X25IPI encountered a Link statement in its X25IPI configuration data set that contained a nondecimal DTE address. The Link entry is discarded; further errors can result.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the DTE address.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2135E Altlink record not preceded by a Link record

Explanation: The X25 server is processing an ALTLINK record that was not preceded by a LINK record in the configuration data set.

System Action: X25 ignores the ALTLINK configuration record.

User or Operator Response: Correct the configuration data set by either removing the ALTLINK statement or adding the correct LINK statement preceding the ALTLINK statement. For more information about the ALTLINK and LINK statements, see *z/OS Communications Server: IP Configuration Guide*.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: STRTALTL

EZB2140E Dest record not preceded by Link record

Explanation: X25IPI encountered a Dest record in its X25IPI configuration data set before any Link record. The Dest record should follow the corresponding Link record in the X25IPI configuration data set. The Dest entry is skipped.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Move the Dest record after the corresponding Link record.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2141E Dest IP address missing

Explanation: X25IPI encountered a Dest record in its X25IPI configuration data set that did not specify an IP address. The Dest entry is discarded.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Complete the Dest entry. See *z/OS Communications Server: IP Configuration Guide* for the format of the Dest record.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2142E Dest IP address must be decimal

Explanation: X25IPI encountered a Dest record in its X25IPI configuration data set specifying a nondecimal destination address. The destination address should be in dotted-decimal form. The Dest entry is discarded.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the IP address. See *z/OS Communications Server: IP Configuration Guide* for the format of the Dest record.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2143E Dest DTE address missing

Explanation: X25IPI encountered a Dest record in its X25IPI configuration data set, which omitted the destination X.25 DTE address specification. The destination X.25 DTE address specification is required on non-DDN networks. The Dest entry is discarded.

EZB2144E • EZB2150E

System Action:

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Complete the Dest record with the X.25 DTE address. See *z/OS Communications Server: IP Configuration Guide* for the format of the Dest record.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2144E Dest DTE address must be decimal

Explanation: X25IPI encountered a Dest record in its X25IPI configuration data set that had a nondecimal destination X.25 DTE address specified. The destination X.25 DTE address should be decimal. The Dest entry is discarded.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the DTE address.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2145E Dest call user data must be hexadecimal

Explanation: X25IPI encountered a Dest record in its X25IPI configuration data set specifying nonhexadecimal call user data (CUD) protocol ID. The CUD should be hexadecimal. The Dest entry is discarded.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the call user data.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2146E Dest facilities data must be hexadecimal

Explanation: The X25 server is processing the facilities field on a DEST statement. The field contained a non-hexadecimal character.

System Action: X25 discards this record and continues.

User or Operator Response: Correct the configuration data set by correcting the DEST facilities field. For more information about the configuration data set statements, see *z/OS Communications Server: IP Configuration Reference*.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: STRTDEST

EZB2150E Datagram size limit missing or not numeric

Explanation: X25IPI encountered a Buffers record in its X25IPI configuration data set that had a missing or nonnumeric buffer size specification. The Buffers record is ignored. The default values are applied.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Complete the Buffers record with the datagram size limit. See *z/OS Communications Server: IP Configuration Guide* for the format of the Dest record.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2151E Datagram size limit not in range 576...2048

Explanation: X25IPI encountered a Buffers record in its X25IPI configuration data set with an incorrect buffer size specification. The buffer size should be in the range 576–2048. The Buffers record is ignored. The default values are applied.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the datagram size limit.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2152E Extra buffer count missing or not numeric

Explanation: X25IPI encountered a Buffers record in its X25IPI configuration data set that had a missing or nonnumeric buffer count field. The extra buffer count is ignored.

System Action: The extra buffer count is ignored. Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the buffer count.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2153E VC send queue limit missing or not numeric

Explanation: X25IPI encountered a Buffers record in its X25IPI configuration data set that had a missing or nonnumeric send queue limit. The send queue limit defaults to 8.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the queue limit.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2155E Inactivity timeout missing or not numeric

Explanation: X25IPI encountered a Timers record in its X25IPI configuration data set that had a missing or nonnumeric inactivity time-out value. The Timers record is ignored. The default values are applied.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Complete the Timers record with the inactivity time-out. See *z/OS Communications Server: IP Configuration Reference* for the format of the Dest record.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2156E Minimum call timer missing or not numeric

Explanation: X25IPI encountered a Timer record in its X25IPI configuration data set that had a missing or nonnumeric minimum time field. This Timer record is ignored. The minimum call time defaults to 60 seconds.

System Action: Processing continues.

EZB2160E • EZB2163E

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the minimum call time.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2160E Options record not preceded by Link record

Explanation: X25IPI encountered an Options record in its X25IPI configuration data set before any Link record. The Options record should follow the associated Link record. The Options record is ignored.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Move the Options record after the corresponding Link record.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2161E Option name not recognized *name*

Explanation: X25IPI encountered an Options record in its X25IPI configuration data set with an unrecognized option name. The remainder of the Options record is skipped.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the option name. See *z/OS Communications Server: IP Configuration Guide* for the format of the Options record.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2162E Option packet size missing or not numeric

Explanation: X25IPI has encountered an Options statement in its X25IPI configuration data set (DD statement X25IPI) that has a missing or nonnumeric packet size specification. The remainder of the Options record is skipped.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the packet size.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2163E Option packet size unacceptable

Explanation: X25IPI encountered an Options record in its X25IPI configuration data set that specified an unacceptable packet size. The size should be 32, 64, 128, 256, 512, 1024, 2048, or 4096. The remainder of the Options record is skipped.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the packet size.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2164E Option window size missing or not numeric

Explanation: XNX25IPI encountered an Options statement in its X25IPI configuration data set that had a missing or nonnumeric window size. The remainder of the Options record is skipped.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the window size.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2165E Option window size not in range 1..7 or 1..127

Explanation: XNX25IPI encountered an Options statement in its X25IPI configuration data set that specified an incorrect window size. The window size should be in the range 1–7 or 1–127. The remainder of the Options record is skipped.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the window size.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2166E Option call user data missing or not hexadecimal

Explanation: The X25 server is processing an OPTION CALLDATA statement. The statement is missing the call user data, or the call user data specified was not hexadecimal.

System Action: X25 discards this record and continues.

User or Operator Response: Correct the configuration data set by correcting the CALLDATA option. For more information about the configuration data set statements, see *z/OS Communications Server: IP Configuration Guide*.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: STRTOPTN

EZB2167E Option facilities data must be hexadecimal

Explanation: The X25 server is processing an OPTION FACILITIES statement. The statement is missing the facilities data, or the facilities data specified was not hexadecimal.

System Action: X25 discards this record and continues.

User or Operator Response: Correct the configuration data set by correcting the FACILITIES option. For more information about the configuration data set statements, see *z/OS Communications Server: IP Configuration Reference*.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: STRTOPTN

EZB2180E FAST record not preceded by a Link record

Explanation: The X25 server is processing the FAST statement in the configuration data set. The FAST statement was not preceded by a LINK statement.

System Action: XNX25IPI ignores the FAST statement.

User or Operator Response: Correct the configuration data set by either removing the FAST statement or preceding

EZB2181E • EZB2184E

the FAST statement with a LINK statement. Rerun the TCPIPX25 catalog procedure. For more information about the configuration data set statements, see *z/OS Communications Server: IP Configuration Guide*.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: STRTFAST

EZB2181E FAST connect LU name prefix missing

Explanation: The X25 server is processing the FAST statement in the configuration data set. The FAST statement did not contain a VC LU prefix.

System Action: XNX25IPI ignores the FAST statement.

User or Operator Response: Correct the configuration data set by adding the prefix to the FAST statement. Rerun the TCPIPX25 catalog procedure. For more information about the configuration data set statements, see *z/OS Communications Server: IP Configuration Guide*.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: STRTFAST

EZB2182E FAST connect LU name too long

Explanation: The X25 server is processing the FAST statement in the configuration data set. The FAST statement contained a VC LU prefix that exceeded eight characters.

System Action: XNX25IPI ignores the FAST statement.

User or Operator Response: Correct the configuration data set by correcting the prefix in the FAST statement. Rerun the TCPIPX25 catalog procedure. For more information about the configuration data set statements, see *z/OS Communications Server: IP Configuration Guide*.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: STRTFAST

EZB2183E FAST connect LU name suffix unacceptable

Explanation: The X25 server is processing the FAST statement in the configuration data set. The FAST statement contained a suffix that had an incorrect character. The suffix should be decimal or hexadecimal, as appropriate to the specified FAST connect numbering scheme.

System Action: Remaining fast-connect LU names will not be generated.

User or Operator Response: Correct the configuration data set by correcting the suffix in the FAST statement. Rerun the TCPIPX25 catalog procedure. For more information about the configuration data set statements, see *z/OS Communications Server: IP Configuration Guide*.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: STRTFAST

EZB2184E FAST connect LU name suffix overflow

Explanation: The X25 server is processing the FAST statement in the configuration data set. The FAST statement contains a suffix that has too many characters. The LU name suffix generated for a fast-connect virtual circuit exceeds the available field width.

System Action: Remaining fast-connect LU names will not be generated.

User or Operator Response: Correct the configuration data set by correcting the suffix in the FAST statement. Rerun

the TCPIPX25 catalog procedure. For more information about the configuration data set statements, see *z/OS Communications Server: IP Configuration Guide*.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: STRTFAST

EZB2201I MCH *luname* OPNDST complete

Explanation: X25IPI has successfully established a VTAM session with the NPSI MCH LU.

System Action: The X25 server continues processing.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: MCHOPNDC

EZB2202I MCH *luname* restart packet sent

Explanation: The X25 server has sent a restart packet to a multichannel link (MCH).

System Action: The contents of the restart packet are displayed in subsequent messages.

User or Operator Response: None.

System Programmer Response: Use the data in problem determination.

Source Data Set: XNX25IPI

Procedure Name: MCHOPNDC

EZB2203I MCH *lu* restarting

Explanation: An X.25 restart exchange was started on the NPSI MCH.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2205E MCH *lu* open failed

Explanation: X25IPI encountered an error opening the indicated NPSI MCH. This message is preceded by the message EZB2407E or EZB2411E reporting a VTAM error code. The MCH LU is enabled for automatic recovery by LOGAPPL, or for manual restart.

System Action: Processing continues.

User or Operator Response: Activate the NPSI MCH in VTAM, and use the X25IPI RESTART command to reacquire the MCH.

System Programmer Response: Use the VTAM error code from the EZB2407E or EZB2411E message to determine why the MCH OPNDST was unsuccessful.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2206E MCH *lu* OPNDST did not complete

Explanation: A VTAM OPNDST request on a NPSI MCH LU was posted complete, but the NIB is not marked open. The contents of the session data area (SDA) for this MCH are dumped. The MCH LU is enabled for automatic recovery by LOGAPPL, or for manual restart.

System Action: Processing continues.

User or Operator Response: Activate the NPSI MCH in VTAM, and use the X25IPI RESTART command to reacquire the MCH.

System Programmer Response: Determine the state of the NPSI MCH LU using the VTAM DISPLAY command.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2210I MCH *luname* packet level ready

Explanation: The X25 server has made a multichannel link (MCH) available for virtual circuit connections.

System Action: XNX25IPI continues processing.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: MCHOPNDC

EZB2211I MCH *luname* packet received

Explanation: The X25 server has received a packet from a multichannel link (MCH).

System Action: The contents of the packet are displayed in subsequent messages.

User or Operator Response: None.

System Programmer Response: Use the data in problem determination.

Source Data Set: XNX25IPI

Procedure Name: MCHRSTIP

EZB2212I MCH *lu* discarded packet during restart

Explanation: X25IPI received an incorrect packet while restarting this MCH. The incorrect packet is dumped and discarded.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2213I MCH *lu* restart indication, cause=*value* diagnostic=*value*

Explanation: X25IPI received an X.25 restart indication for this MCH during the restart procedure with the indicated cause and diagnostic bytes. The X.25 network interface has been reinitialized. Virtual circuit connections on the MCH are closed, and the X.25 restart procedure is used to place the MCH back in operation. See *X.25 Network Control Program Packet Switching Interface Diagnosis, Customization, and Tuning* for a list of X.25 cause and diagnostic codes. See the provider of the X.25 network service for documentation that lists additional diagnostic codes.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2214I MCH *lu* restart confirmation

Explanation: X25IPI received a restart confirmation for this MCH. The MCH is marked ready for new connections.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2215I MCH *lu* restart complete

Explanation: X25IPI has completed terminating virtual circuits on an MCH undergoing restart. An X.25 restart confirmation is sent to complete the restart procedure.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2221I MCH *lu* restart indication, *cause=cause*, *diagnostic=diagnostic*

Explanation: X25IPI received a restart indication on this MCH with the indicated cause and diagnostic bytes. The X.25 network interface has been reinitialized. Virtual circuit connections on the MCH are closed, and the X.25 restart procedure is used to place the MCH back in operation. See *X.25 Network Control Program Packet Switching Interface Diagnosis, Customization, and Tuning* for a list of X.25 cause and diagnostic codes. See the provider of the X.25 network service for documentation that lists additional diagnostic codes.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2222I MCH *lu* restart confirm packet sent

Explanation: X25IPI transmitted a restart confirmation packet for this MCH. The restart confirmation packet is dumped.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2223I MCH *luname* request packet sent

Explanation: The X25 server has sent a request packet from a multichannel link (MCH).

System Action: The contents of the restart packet are displayed in subsequent messages.

User or Operator Response: None.

System Programmer Response: Use the data in problem determination.

Source Data Set: XNX25IPI

Procedure Name: MCSTSRSR

EZB2224I MCH *lu* restarting

Explanation: X25IPI is restarting the indicated MCH.

System Action: X25IPI waits to receive a restart confirmation on the MCH.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2230I MCH *luname* packet received

Explanation: The X25 server has received a packet from a multichannel link (MCH).

System Action: The contents of the restart packet are displayed in subsequent messages.

User or Operator Response: None.

System Programmer Response: Use the data in problem determination.

Source Data Set: XNX25IPI

Procedure Name: MCHRDYIP

EZB2231I MCH *lu* orphan packet received

Explanation: X25IPI received a packet for which it could find no associated connection. An X.25 clear request is sent on the virtual circuit.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2232I MCH *lu* diagnostic packet

Explanation: X25IPI received an X.25 diagnostic packet on this MCH. The diagnostic packet is dumped and discarded.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: Use the diagnostic packet to obtain additional information about X.25 network errors.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2233I MCH *lu* clear request sent

Explanation: X25IPI sent a clear request on the indicated MCH to recover from an error situation.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2234E MCH *lu* no free path available for incoming call MCH *lu* check number of logical channels on Link record

Explanation: X25IPI received an incoming call, but has no session areas available to handle it. This indicates that there are more virtual circuits subscribed than were specified on the LINK record in the X25IPI configuration data set. The incoming call is cleared.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Increase the virtual circuit count on the Link record to match the number of virtual circuits defined in the NPSI configuration.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2235W MCH *lu* orphan packet discarded

Explanation: X25IPI received a packet that it could not associate with any connection. The packet is dumped and discarded.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Determine the type of X.25 packet from the dump. A clear confirmation (X'17') can be discarded harmlessly.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2236E MCH *lu* unrecognized packet received

Explanation: X25IPI did not recognize the type code in a packet received from NPSI. The unrecognized packet is dumped and discarded.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Determine the packet type from the dump; contact the IBM Software Support Center.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2250I MCH *lu* terminating

Explanation: X25IPI is terminating this MCH session in response to an error condition or a HALT command.

System Action: Connections on virtual circuits associated with the MCH are terminated, and the MCH session is closed.

EZB2251I • EZB2281I

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2251I MCH *lu* closed

Explanation: X25IPI received a CLSDST completion indication for this MCH, indicating that this MCH has closed.

System Action: The MCH LU is enabled for automatic recovery by LOGAPPL, or for manual restart.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2252I MCH *luname* logon

Explanation: The multichannel link (MCH) was inactive and a logon notification was received. XNX25IPI opens a VTAM session.

System Action: XNX25IPI continues processing.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: MCHRDYIP

EZB2280E MCH *lu* session loss code *code*

Explanation: X25IPI received the indicated MCH status change code while the MCH was not operational. See message EZB2285E for the loss codes.

System Action: The MCH LU is enabled for automatic recovery by LOGAPPL, or for manual restart.

User or Operator Response: Notify the system programmer of the error.

System Programmer Response: Manually restart X25IPI, if necessary.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2281I VC *vc* packet discarded on failed session

Explanation: X25IPI discarded a packet received after a session was closed.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2282E MCH lu unexpected request completion

Explanation: X25IPI received a VTAM request completion notification for this MCH. This indicates a program error, because MCH sends are done synchronously. The notification is ignored.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2283I MCH lu DATE error report command=command error=error

Explanation: X25IPI received a NPSI Dedicated Access to X.25 Transport Extension (DATE) error report for the indicated command and error. See *X.25 Network Control Program Packet Switching Interface Host Programming* for the error codes.

System Action: X25IPI attempts to recover from the error.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2284E MCH lu DATE error report command=command error=error

Explanation: X25IPI encountered a DATE error because of an incorrect logical channel number. X25IPI discards the error indication. See *X.25 Network Control Program Packet Switching Interface Host Programming* for the error codes.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2285E MCH lu session loss code code

Explanation: X25IPI received a session status notification with the indicated session loss code for this MCH.

Code	Description
Less than 50	VTAM LOSTERM exit codes
50	VTAM SCIP exit UNBIND
64000	VTAM NS exit CLEANUP
64001	VTAM NS exit session initiation failure
64002	VTAM NS exit session initiation negative response

The MCH LU is enabled for automatic recovery by LOGAPPL, or for manual restart.

System Action: Processing continues.

User or Operator Response: Reactivate the NPSI MCH in VTAM. Use the X25IPI RESTART command to reacquire the MCH LU.

System Programmer Response: Use the VTAM error code to determine the reason why the session was unsuccessful.

EZB2301I • EZB2305W

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2301I VC ID incoming call from *address* user data *value*

Explanation: X25IPI has received an incoming call on this virtual circuit from the indicated address with the indicated user data (protocol ID). The call is accepted or refused.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: Use the address to determine the source of the connection.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2302I VC *vc* call accept packet sent

Explanation: X25IPI accepted an incoming call on this virtual circuit.

System Action: The contents of the X.25 call accept packet are displayed in subsequent messages.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: VCINCOM

EZB2304W VC *vc* incoming call cleared: caller not known

Explanation: X25IPI received a call on this virtual circuit from an address not present in the Dest entries for the associated MCH. The incoming call is cleared.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Determine the identity of the remote system initiating the call request, and add the address to the Dest list if the connection is authorized. The X.25 address of the calling system was noted in the preceding EZB2301I message for the VC ID.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2305W VC *vc* incoming call cleared: draining

Explanation: X25IPI cleared an incoming call because a VTAM HALT request has been issued to end communication. Incoming calls are refused once the VTAM HALT command is issued.

System Action: Processing continues.

User or Operator Response: Shutdown and restart X25IPI when VTAM is restarted.

System Programmer Response: Assist the user as necessary.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2306E VC *vc* incoming call cleared: *reason*

Explanation: X25IPI cleared an incoming call on this virtual circuit because of an error in the format of the X.25 call request packet:

- Called address is not decimal.
- Calling address is not decimal.
- CUD field is too long.
- CUD field not acceptable.
- Duplicate address.
- Facilities not acceptable.
- Reverse charging has not been enabled on the associated link.
- Reverse charging refused.
- Reverse charging was specified in the call.
- Reverse charging has not been enabled on the associated link.
- The address duplicated that of another connection.

The incoming call is cleared.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the remote system that is generating the incorrect calls. The X.25 address of the calling system was noted in the preceding EZB2301I message for the VC ID.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2307I VC *vc* clear request packet sent

Explanation: X25IPI refused an incoming call on this virtual circuit.

System Action: The contents of the X.25 clear request packet are displayed in subsequent messages.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: VCINCOM

EZB2308I VC *vc* finished sending call confirm

Explanation: A VTAM OPNDST call was accepted and the NPSI SEND completed.

System Action: XNX25IPI continues processing.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: VCACPTSC

EZB2310I VC *vc* outgoing call to *address*

Explanation: X25IPI is placing an outgoing call to this indicated address on this virtual circuit. Queued datagrams are sent on the connection after the call is accepted by the remote system.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: Use the address to determine the destination of the connection.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2311I VC *vc* call request packet sent

Explanation: X25IPI placed an outgoing call on this virtual circuit.

System Action: The contents of the X.25 call request packet are displayed in subsequent messages.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: VCCLLFCL

EZB2312I VC *vc* call request sent

Explanation: X25IPI placed an outgoing call request on this virtual circuit. The X.25 NCP Packet Switching Interface (NPSI) SEND is complete.

System Action: XNX25IPI continues processing.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: VCCALLSC

EZB2313W VC *vc* call timer expired

Explanation: The remote system has not responded to a call request within 200 seconds. The call is ended, and queued datagrams are discarded.

System Action: Processing continues.

User or Operator Response: Check the status of the remote system and the X.25 network. The X.25 address of the calling system was noted in the preceding EZB2310I message for the VC ID.

System Programmer Response: Assist the user as necessary.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2314I VC *vc* call accepted by *address* user data *value*

Explanation: An outgoing call on this virtual circuit was accepted by the remote system at the indicated address with the indicated user data (protocol specifier). Queued datagrams are sent.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: Use the address to determine the destination of the connection.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2315I VC *vc* retrying call with packet size *size*

Explanation: The packet size for this virtual circuit was reduced by the network or responder. NPSI DATE cannot handle this negotiation, thus the call is cleared and placed again with the smaller packet size.

System Action: Processing continues.

User or Operator Response: Tell the system programmer if this message recurs frequently.

System Programmer Response: Add or change an OPTIONS PACKETSZ entry on the associated Link specifying the smallest packet size used by other systems on the X.25 network. Consider using the NPSI GATE facility rather than DATE. GATE can handle the reduced packet size without the call to be repeated.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2316E VC *vc* outgoing call cleared: *reason*

Explanation: X25IPI cleared an outgoing call on this virtual circuit because of an error in one of the following formats:

- The X.25 call accept packet.
- Accepting user data is too long.
- Called address is not decimal.
- Calling address is not decimal.
- Facilities not acceptable.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the remote system that is generating the incorrect calls. The X.25 address of the called system was noted in the preceding EZB2310I message for the VC ID.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2317I VC *vc* call to address refused, cause=*cause* diagnostic=*diagnostic*

Explanation: An outgoing call on this virtual circuit to the indicated address was refused by the X.25 network, with the indicated cause and diagnostic bytes. Datagrams queued for the remote system are discarded. A new call is attempted when the TCP acknowledgment timer expires, or when a new connection is requested to the destination. TCP connections to the destinations handled by the remote system are unsuccessful if calls are not accepted within the initial connection time-out. See *X.25 Network Control Program Packet Switching Interface Diagnosis, Customization, and Tuning* for X.25 cause and diagnostic codes. See the X.25 network service provider for documentation about additional diagnostic codes.

System Action: Processing continues.

User or Operator Response: Check that the remote system and the X.25 network are operational.

System Programmer Response: Assist the user as necessary.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2320I VC *vc* NPSI logon LU *lu*

Explanation: NPSI generated a VTAM session logon from the indicated LU for the virtual circuit. The NPSI LU logon is accepted.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: Use the LU name for NPSI problem determination.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2321E VC *vc* session loss code

Explanation: X25IPI received the indicated virtual circuit (VC) status change code while the virtual circuit (VC) was not operational.

System Action: The virtual circuit (VC) LU is enabled for automatic recovery by LOGAPPL or for manual restart.

User or Operator Response: Notify the system programmer of the error.

System Programmer Response: Restart XNX25IPI if necessary.

Source Data Set: XNX25IPI

Procedure Name: VCLOGON

EZB2322I VC *vc* OPNDST complete

Explanation: A VTAM OPNDST request on a NPSI virtual circuit (VC) LU was posted complete.

System Action: X25IPI continues processing.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: VCOPENC

EZB2323E VC *vc* OPNDST did not complete

Explanation: A VTAM OPNDST request on a NPSI VC LU was posted complete, but the NIB is not marked open. The contents of the session data area (SDA) for this VC are dumped. The virtual circuit call is cleared.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Determine the state of the NPSI VC LU using the VTAM DISPLAY command.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2324I VC *vc* LU *lu* ready

Explanation: The virtual circuit LU session is ready for data transfer.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2325I VC *vc* facilities: *facilities*

Explanation: A list of facilities for this virtual circuit follows. The following are the facilities codes that can be noted:

Codes	Description
-------	-------------

pktpacket	
-----------	--

	The noted packet size is used
--	-------------------------------

precedence	
------------	--

	The noted DDN precedence is applied
--	-------------------------------------

priority	
----------	--

	Priority handling and charging is applied
--	---

revchg	
--------	--

	Reverse charging is applied
--	-----------------------------

standard

DDN standard service is used

wdwindow

The noted window size is used

The noted facilities are applied to the connection.

System Action: Processing continues.**User or Operator Response:** None.**System Programmer Response:** Use the information to determine the X.25 network facilities being used on the connection.**Source Data Set:** XNX25IPI**Procedure Name:** None.**EZB2326E VC *vc* facilities field unacceptable at offset *offset*****Explanation:** X25IPI encountered an incorrect X.25 call facilities field from a remote system for this virtual circuit. The X.25 call facilities field is dumped, and the call is cleared.**System Action:** Processing continues.**User or Operator Response:** Tell the system programmer about the error.**System Programmer Response:** Correct the remote system that is generating the incorrect facilities. The X.25 address of the remote system was noted in the preceding EZB2301I or EZB2310I message for the VC ID.**Source Data Set:** XNX25IPI**Procedure Name:** None.**EZB2330I VC *vc* call complete****Explanation:** A VTAM OPNDST request on a NPSI virtual circuit (VC) LU was posted complete.**System Action:** X25IPI continues processing.**User or Operator Response:** None.**System Programmer Response:** None.**Source Data Set:** XNX25IPI**Procedure Name:** MCHRDYIP**EZB2331I VC *vc* data sent****Explanation:** X25IPI has sent a datagram to the X.25 NCP Packet Switching Interface (NPSI).**System Action:** The contents of the X.25 data packet sequence are displayed in subsequent messages.**User or Operator Response:** None.**System Programmer Response:** None.**Source Data Set:** XNX25IPI**Procedure Name:** VCLOGON**EZB2332I VC *vc* data received****Explanation:** X25IPI has received a data packet that contained an IP datagram.**System Action:** The contents of the X.25 data packet sequence are displayed in subsequent messages.**User or Operator Response:** None.**System Programmer Response:** None.

EZB2333I • EZB2336I

Source Data Set: XNX25IPI

Procedure Name: VCRI PDT

EZB2333I VC *vc* packet received

Explanation: X25IPI has received a General Access To X.25 Transport Extension (GATE) control packet on this virtual circuit (VC).

System Action: The contents of the control packet are displayed in subsequent messages.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: VCRI PDT

EZB2334E VC *vc* oversize data packet received, length=*length*

Explanation: A datagram was received on this virtual circuit that had a length exceeding the buffer size specified in the X25IPI configuration data set. Either the local or remote system is misconfigured. The packet is dumped, and the connection is cleared.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Check the buffer size on the Buffers record in the X25IPI configuration data set. The buffer size should be large enough to hold the maximum IP datagram permitted by the usage agreements of the X.25 network. The X.25 address of the remote system was noted in the preceding EZB2301I or EZB2310I message for the VC ID.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2335E VC *vc* unrecognized packet received

Explanation: X25IPI did not recognize the type code in a packet received from NPSI GATE. The unrecognized packet is dumped and discarded.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Determine the packet type from the dump; contact the IBM Software Support Center.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2336I VC *vc* inactivity timer expired

Explanation: The inactivity time for this virtual circuit passed with no data transferred. The virtual circuit connection is closed.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2340I VC *vc* call reset, cause=*cause* diagnostic=*diagnostic*

Explanation: A call reset was received for this virtual circuit with the indicated cause and diagnostic bytes. The reset is confirmed, and data transfer continues. See *X.25 Network Control Program Packet Switching Interface Diagnosis, Customization, and Tuning* for a list of X.25 cause and diagnostic codes. See the provider of the X.25 network service for documentation that lists additional diagnostic codes.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2341I VC *vc* reset collision

Explanation: A reset collision occurred on this virtual circuit. Data transfer resumes.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2342I VC *vc* reset confirmed

Explanation: A reset on this virtual circuit was confirmed by the remote system. Data transfer resumes on the virtual circuit.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2343W VC *vc* qualified data packet discarded

Explanation: A qualified data packet was received on an IP connection. Qualified data packets are not specified for use on IP connections. The qualified data packet is dumped and discarded.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the remote system that is sending qualified data packets. The X.25 address of the remote system was noted in the preceding EZB2301I or EZB2310I message for the VC ID.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2344W VC *vc* interrupt indication received

Explanation: An interrupt indication was received for this virtual circuit. Interrupt packets are not specified for use on IP connections. The interrupt packet is dumped, and an interrupt response is sent.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Correct the remote system that is sending interrupt packets. The X.25 address of the remote system was noted in the preceding EZB2301I or EZB2310I message for the VC ID.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2345E VC *vc* interrupt confirmed

Explanation: An interrupt was confirmed for this virtual circuit. Interrupt packets are not specified for use on IP connections. The virtual circuit is reset, and data transfer continues.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2346I VC *vc* reset confirmation packet sent

Explanation: A reset packet was received and X25IPI has sent a reset confirmation packet on this virtual circuit (VC).

System Action: The contents of the X.25 reset confirmation packet are displayed in subsequent messages.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: VCRIPDT

EZB2347I VC *vc* interrupt confirm packet sent

Explanation: An interrupt packet was received and X25IPI has sent an interrupt confirmation packet on this virtual circuit (VC).

System Action: The contents of the X.25 interrupt confirmation packet are displayed in subsequent messages.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: VCCONF1

EZB2348I VC *vc* reset request packet sent

Explanation: X25IPI has sent a reset packet on this virtual circuit (VC) to reset the connection.

System Action: The contents of the X.25 reset request packet are displayed in subsequent messages.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: VCCONF1

EZB2350I VC *vc* call cleared, cause=*cause* diagnostic=*diagnostic*

Explanation: A call was cleared on this virtual circuit with the indicated cause and diagnostic bytes. A clear confirmation is sent, and the virtual circuit connection is closed. See *X.25 Network Control Program Packet Switching Interface Diagnosis, Customization, and Tuning* for a list of X.25 cause and diagnostic codes. See the provider of the X.25 network service for documentation that lists additional diagnostic codes.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: Use the cause and diagnostic codes to determine the reason the call was cleared.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2351I VC *vc* connection terminated for address: sent count received count dropped count

Explanation: The number of datagrams sent, received, and dropped on the virtual circuit to the remote system with the indicated X.25 address is shown when the connection is ended.

System Action: The VC LU session is closed.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2352I VC *vc* closed

Explanation: The call on the virtual circuit was ended. The virtual circuit is reused for new calls.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2353I VC *vc* clear request packet sent

Explanation: X25IPI has sent a clear packet on this virtual circuit (VC) to clear the connection. XNX25IPI is in the process of closing the connection.

System Action: The contents of the X.25 clear request packet are displayed in subsequent messages.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: VCCLROUT

EZB2354I VC *vc* clear confirm packet sent

Explanation: X25IPI has sent a clear confirmation packet on this virtual circuit (VC) for the clear request. XNX25IPI is in the process of closing the connection.

System Action: The contents of the X.25 clear request packet are displayed in subsequent messages.

User or Operator Response: None.

System Programmer Response: None.

EZB2355I • EZB2358I

Source Data Set: XNX25IPI

Procedure Name: VCCLROUT

EZB2355I VC *vc* close pending

Explanation: A close of this virtual circuit is partially completed.

System Action: X25IPI waits for the remaining close events to occur.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2356W VC *vc* unable to clear GATE call in state P2

Explanation: X25IPI needed to clear a NPSI GATE call while the call was pending. The NPSI GATE programming interface does not allow a call to be cleared in this state.

System Action: X25IPI waits for action by the X.25 network.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Determine the reason the X.25 network or remote system did not respond to the call request.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2357E VC *vc* clearing limit exceeded

Explanation: No clear response was received from the remote system after four clear requests. The virtual circuit is marked as cleared.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Determine the reason the X.25 network or remote system did not respond to the clear request.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2358I VC *vc* clear confirmed

Explanation: The remote system responded to a clear request on this virtual circuit. Virtual circuit termination continues; message EZB2352I should follow.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2360I VC *vc* closed LU

Explanation: X25IPI has completed a VTAM CLSDST for a virtual circuit (VC) LU session.

System Action: The X.25 server continues processing.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: VCCLOSRC

EZB2361W VC *vc* clearing timer expired

Explanation: The remote system did not respond to a clear request within 180 seconds. The clear request is retried 4 times.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Determine the reason the remote system did not respond to the clear request.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2362I VC *vc* clear request packet sent

Explanation: X25IPI has sent a clear packet on this virtual circuit (VC) to clear the connection. XNX25IPI is in the process of closing the connection.

System Action: The contents of the X.25 clear request packet are displayed in subsequent messages.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: VCCLRTMR

EZB2363E VC *vc* clearing limit exceeded

Explanation: No clear response was received from the remote system after four clear requests. The virtual circuit is marked as cleared.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Determine the reason the X.25 network or remote system did not respond to the clear request.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2364I VC *vc* clear confirmation sent

Explanation: A clear request from the remote system was confirmed on this virtual circuit. Virtual circuit termination continues; message EZB2352I should follow.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

EZB2365I • EZB2370E

Procedure Name: None.

EZB2365I VC *vc* clear sent

Explanation: A clear request was sent on this virtual circuit. Virtual circuit termination continues; message EZB2352I should follow.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2366I VC *vc* finished lu *luname* activity

Explanation: X25IPI reports the completion of a pending VTAM request during virtual circuit (VC) closing.

System Action: The X.25 server continues processing.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: VCCLWTRC

EZB2367I VC *vc* reuse delay ended

Explanation: The timer expired for NPSI cleanup for this virtual circuit. A new call can now be made on this virtual circuit.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2368I VC *vc* retry busy call

Explanation: X25IPI retries a call that previously received a busy indication from NPSI.

System Action: Processing continues.

User or Operator Response: If the error recurs, tell the system programmer about the error.

System Programmer Response: If the error recurs, check the number of virtual circuits specified on the Link record in the X25IPI configuration data set against the number defined in the NPSI configuration. Use the VTAM DISPLAY command to determine the state of the NPSI switched VC LUs.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2370E VC *vc* inactive: VTAM request completion REQ=*request*

Explanation: A VTAM request has completed for this virtual circuit after the connection has ended.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2371E VC *vc* inactive: session loss code *code*

Explanation: X25IPI received a session status notification with the indicated session loss code for this virtual circuit after the connection was ended. The notification is ignored. See message EZB2383I for the loss codes.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2372I VC *vc* inactive: session cleanup

Explanation: Session cleanup is in progress for this inactive virtual circuit. The VC LU session is closed.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2373E VC *vc* inactive: VTAM logon refused

Explanation: A late logon from a NPSI VC LU was refused after a call was cleared. The logon is rejected.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2374I VC *vc* packet discarded on dead connection

Explanation: A packet was received on this virtual circuit after the connection was ended. The packet is discarded.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2381E VC *vc* DATE error report command=*command* error=*error*

Explanation: X25IPI received a NPSI DATE error report for the indicated command and error. See *X.25 Network Control Program Packet Switching Interface Host Programming* for the error codes.

System Action: X25IPI attempts to recover from the error.

User or Operator Response: None.

System Programmer Response: None.

EZB2382E • EZB2385E

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2382E VC *vc* **GATE error report** *command=command error=error*

Explanation: X25IPI received a NPSI GATE error report for the indicated command and error. See *X.25 Network Control Program Packet Switching Interface Host Programming* for the error codes.

System Action: X25IPI attempts to recover from the error.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2383E VC *vc* **session loss code** *value* **on LU** *lu*

Explanation: A session loss status notification was received for this virtual circuit with the indicated loss code.

Code **Description**

Less than 50

VTAM LOSTERM exit codes

50 VTAM SCIP exit UNBIND

64000 VTAM NS exit CLEANUP

64001 VTAM NS exit session initiation failure

64002 VTAM NS exit session initiation negative response

The virtual circuit connection is closed.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: Use the VTAM error code to determine why the session was unsuccessful.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2384I VC *vc* **discarded packet** *packet* **in state** *state*

Explanation: The indicated packet was discarded on this virtual circuit to accomplish error recovery. Virtual circuit error recovery is completed.

System Action: Processing continues.

User or Operator Response: See message EZB2021I VC for the call state codes.

System Programmer Response: Assist the user as necessary.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2385E VC *vc* **received packet** *packet* **invalid in state** *state*

Explanation: The indicated received packet was incorrect for the current virtual circuit state. The virtual circuit is reset or cleared.

System Action: Processing continues.

User or Operator Response: See message EZB2021I VC for the call state codes. Tell the system programmer about the error.

System Programmer Response: Determine the reason the X.25 network or remote system sent the incorrect packet.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2386E VC *vc* discarded packet *packet* in state *state*

Explanation: The indicated packet for VC was discarded because the state was not valid for the current virtual circuit state. The virtual circuit *vc* is cleared.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: Determine the reason the remote X.25 network sent the incorrect *state* value for the circuit.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2401E VTAM GENCB failed, R15=*value* R0=*value*

Explanation: A VTAM GENCB call was unsuccessful with the indicated R15 and R0 values.

System Action: X25IPI does not start.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Reassemble X25IPI with the most recent VTAM macro library.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2402E VTAM ACB OPEN failed, R15=*value* ACBERFLG=*value*

Explanation: A VTAM OPEN request failed with the indicated R15 and ACBERFLG values.

System Action: X25IPI does not start.

User or Operator Response: Activate the X25IPI application in VTAM.

System Programmer Response: Use the VTAM error codes to determine why the VTAM ACB OPEN macro was unsuccessful. Refer to *z/OS Communications Server: SNA Programming* where codes can be found.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2403E VTAM SETLOGON failed, RTNCD=*rc* FDB2=*value*

Explanation: A VTAM SETLOGON request failed with the indicated return code and FDB2 values.

System Action: Processing continues, but virtual circuit (VC) LU logons will fail.

User or Operator Response: None.

System Programmer Response: Use the VTAM error code to determine the reason for the failure of the VTAM SETLOGON request.

Source Data Set: XNX25IPI

Procedure Name: VTAMENAB

EZB2404E VTAM SETLOGON QUIESCE failed, RTNCD=*rc* FDB2=*value*

Explanation: A VTAM SETLOGON QUIESCE request failed with the indicated return code and FDB2 values.

System Action: X25IPI termination continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Use the VTAM error codes to determine why the VTAM SETLOGON macro. Refer to the *z/OS Communications Server: SNA Programming* where codes can be found.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2405E VTAM ACB CLOSE failed, R15=*value* ACBERFLG=*value*

Explanation: A VTAM CLOSE request failed with the indicated return values.

System Action: X25IPI termination continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Use the VTAM error codes to determine why the VTAM CLOSE macro was unsuccessful. Refer to the *z/OS Communications Server: SNA Programming* where codes can be found.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2406E VTAM GENCB failed, R15=*value* R0=*value*

Explanation: A VTAM GENCB call failed with the indicated R15 and R0 values.

System Action: The X.25 server continues processing.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Reassemble X25IPI with the most recent VTAM macro library.

Source Data Set: XNX25IPI

Procedure Name: VTAMSIN

EZB2407E VTAM VTAM request lu failed, RTNCD=*rc* FDB2=*value*

Explanation: The indicated VTAM request was unsuccessful with the indicated return code and FDB2 values. The virtual circuit can become unusable.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Use the VTAM error codes to determine why the VTAM request was unsuccessful. Refer to the *z/OS Communications Server: SNA Programming* where codes can be found. Determine the state of the NPSI LU using the VTAM DISPLAY command. If all virtual circuits become unusable, HALT and restart the X25IPI application.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2410I VTAM request complete for luname REQ=*value*

Explanation: X25IPI completed a VTAM request. See message EZB2411E for the request codes.

System Action: The X.25 server continues processing.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: VTAMCHK

EZB2411E VTAM request failed for lu REQ=request RNTCD=rc FDB2=value sense=value (caller)

Explanation: The indicated VTAM request was unsuccessful with the indicated return code, FDB2, and sense values. The caller field specifies the routine that issued the request. The following are the request codes:

Request Code

	Description
X'17'	OPNDST
X'1F'	CLSDST
X'22'	SEND
X'23'	RECEIVE

System Action: If the error occurred on a NPSI MCH session, the MCH is shut down. If the error occurred on a VC session, the connection is closed.

User or Operator Response: If the error occurred on a NPSI MCH session, activate the NPSI MCH in VTAM and use the X25IPI RESTART command to reacquire the MCH.

System Programmer Response: Use the VTAM error codes to determine why the VTAM ACB OPEN macro was unsuccessful. Refer to the *z/OS Communications Server: SNA Programming* where codes can be found.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2420I Logon exit refusing session

Explanation: XNX25IPI is refusing a logon session because:

- For an MCH with LOGAPPL coded, the MCH is not defined by a LINK entry, or the MCH is not in a state where a logon is expected
- The logon was not initiated by X.25 NPSI
- A virtual circuit is not in a state where a logon is expected, possibly because a call request was unsuccessful.

The NPSI LU session logon is refused.

System Action: Processing continues.

User or Operator Response: Notify the system programmer if the message is issued frequently, and connections are failing.

System Programmer Response: Check the source of the logons; determine why the call request was unsuccessful.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2421E Network services exit: Unrecognized request type

Explanation: XNX25IPI received an unrecognized Network Services RU from VTAM. The following are the supported RU:

- Cleanup Session (X'810629')
- Notify (X'810620')
- NS Procedure Error (X'010604').

VTAM is notified that the NS RU was not handled.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Obtain a VTAM buffer trace to determine the Network Services RU type.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2422I Network services exit: Cleanup session notification posted

Explanation: A VTAM Cleanup Session RU was received. The VTAM session is ended. A failure message for the MCH or virtual circuit (VC) session follows.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2423W Network services exit: Session initiation failure notification posted

Explanation: A VTAM Session initiation failure RU was received. The VTAM session is ended. A failure message for the MCH or VC session can follow.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2424T Session control exit: Unrecognized or unexpected request type

Explanation: An unexpected VTAM Session Control RU was received. ABEND code X'941' is displayed with this message.

System Action: X25IPI abends.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Determine the source of VTAM session control request. Submit the ABEND dump to the IBM Software Support Center.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2425I Session control exit: UNBIND notification posted

Explanation: A VTAM Session Control RU of type UNBIND was received. The VTAM session is ended. A failure message for the MCH or VC session can follow.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2431E VTAM terminating, reason unknown

Explanation: A VTAM Shutdown notification was received. The cause is unknown. This rules out a standard shutdown, HALT QUICK, INACT, or HALT CANCEL reason codes.

System Action: X25IPI ends.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Determine the source of the Shutdown notification.

Source Data Set: NX25IPI

Procedure Name: None.

EZB2432I VTAM HALT issued, drain flag set

Explanation: A VTAM HALT request was issued.

System Action: X25IPI ends after current connections are closed. New connections are refused.

User or Operator Response: Restart X25IPI, if appropriate, after VTAM resumes.

System Programmer Response: Assist the user as necessary.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2433I VTAM HALT QUICK or VARY INACT, *application* issued

Explanation: A VTAM HALT QUICK or VARY INACT request was issued.

System Action: X25IPI ends.

User or Operator Response: Restart X25IPI, if appropriate, after VTAM resumes.

System Programmer Response: Assist the user as necessary.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2434I VTAM HALT CANCEL issued

Explanation: A VTAM HALT CANCEL request was issued.

System Action: X25IPI ends.

User or Operator Response: Restart X25IPI, if appropriate, after VTAM resumes.

System Programmer Response: Assist the user as necessary.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2453I IP AS_userid path refused for userid *userid*, draining

Explanation: A DLC connection from the TCPIP address space was refused because a VTAM HALT command had been issued.

System Action: Processing continues.

User or Operator Response: HALT and restart the X25IPI application when VTAM is restarted.

System Programmer Response: Assist the user as necessary.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2460E Destination address *address* not configured

Explanation: An IP datagram was received from TCPIP for transmission to the indicated address, which is not listed in the Dest entries in the X25IPI configuration data set. The IP datagram is discarded.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Add a Dest record to the X25IPI configuration data set for the IP address or network. Check the GATEWAY entries in *hlq*.PROFILE.TCPIP for a misrouted network number.

EZB2462E • EZB2493I

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2462E IP *AS_userid* rejected message: too long IP *AS_userid* rejected message: too short

Explanation: A DLC message was received from the TCPIP address space which was too long or too short. The message is rejected.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2480I IP *AS_id* disconnected: sent *count* received *count* dropped *count*

Explanation: The number of datagrams sent, received, and dropped on the DLC connection to the TCPIP address space are shown when the connection is terminated.

System Action: X25IPI ends.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2491I DLC AS *asname* path severed

Explanation: The DLC connection to TCPIP has been severed, most likely because TCPIP is being terminated or the X25NPSI device has been STOPped.

System Action: X25IPI exits.

User or Operator Response: Restart X25IPI, if appropriate, after TCPIP is restarted.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: IPRDYPN

EZB2492I DLC AS *path* accepted for userid *userid*

Explanation: A DLC connection from the TCPIP address space was accepted. IP datagrams are transferred.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2493I DLC received data

Explanation: X25IPI displays the contents of an IP datagram received from TCPIP in subsequent messages.

System Action: The X.25 server continues processing.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: IPRDYPN

EZB2494I **DLC send datagram length** *length next offset offset*

Explanation: X25IPI displays the length of the datagram to be sent to TCPIP.

System Action: The X.25 server continues processing.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: IPSENDDG

EZB2495I **DLC send total length** *length*

Explanation: X25IPI displays the total length of IP datagrams to be sent to TCPIP.

System Action: The X.25 server continues processing.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: XNX25IPI

Procedure Name: IPSENDDG

EZB2496E **XNX25IUT failed: CSM storage problem**

Explanation: XNX25IUT interface returned a return code of 4. This indicates an error with CSM storage.

System Action: X25IPI closes connection with TCPIP.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Check for correct installation of the MVS TCPIP product.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2497E **DLC *function* failed: STAFD= *reason***

Explanation: XNX25IUT interface returned a return code of 8. The indicated DLC function, issued by XNX25IPI, failed for the reason indicated in the STAFD reason field.

- 1 INIT
- 2 SEND
- 3 RECV
- 4 CLEAR
- 5 TERM

System Action: The DLC connection is severed.

User or Operator Response: Tell the system programmer.

System Programmer Response: Perform the action described in *z/OS Communications Server: IP and SNA Codes* for the indicated status code.

Source Data Set: XNX25IPI

Procedure Name: None.

EZB2498E

EZB2498E XNX25IUT failed: Unexpected return code

Explanation: XNX25IUT interface returned a return code other than 0, 4, or 8.

System Action: Processing continues.

User or Operator Response: Tell the system programmer about the error.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: XNX25IPI

Procedure Name: None.

Chapter 4. EZB3xxxx messages

EZB3000I *** Can't find initialize address for server *server: return_code*

Explanation: The local host was unable to find the address of the Domain Name Server. Without the Domain Name Server, the local host is unable to determine the addresses of other hosts on the network.

System Action: TCPIP continues.

User or Operator Response: Notify the system programmer of the problem.

System Programmer Response: Add the address of the Domain Name Server (NSINTERADDR) to the *hlq.TCPIP.DATA* data set. For more information about the *hlq.TCPIP.DATA* data set, refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: NSLOOKUP

Procedure Name: main

EZB3001I Usage:

EZB3002I nslookup [-opt ...] # interactive mode using default server

EZB3003I nslookup [-opt ...] - server #interactive mode using default server

EZB3004I nslookup [-opt ...] host #just look up 'host' using default server

EZB3005I nslookup [-opt ...] host #just look up 'host' using 'server'

Explanation: These messages give the format and usage for the NSLOOKUP command. For more information about the NSLOOKUP command, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NSLOOKUP

Procedure Name: Usage

EZB3006I SetDefaultServer: invalid name: *name*

Explanation: The name specified for the default Domain Name Server does not correspond to a local host. The Domain Name Server is not initialized.

System Action: TCPIP continues.

User or Operator Response: Notify the system programmer of the problem.

System Programmer Response: Correct the default Domain Name Server name (NSINTERADDR) in the *hlq.TCPIP.DATA* data set. For more information about configuring the Domain Name Server, refer to *z/OS Communications Server: IP Configuration Reference*.

Source Data Set: NSLOOKUP

Procedure Name: SetDefaultServer

EZB3007I *** Can't find address for server *server: reason*

Explanation: TCPIP was unable to change to another Domain Name Server because it could not find an address for the new Domain Name Server. The reason it could not find the address is displayed in the message.

System Action: TCPIP continues.

User or Operator Response: Notify the system programmer of the problem.

System Programmer Response: Respond as indicated by the *reason* portion of this message.

Source Data Set: NSLOOKUP

Procedure Name: SetDefaultServer

EZB3008I *** No query_type (*query_abbrev.*) records available for host

Explanation: No information of the type requested is available for the indicated host.

System Action: TCPIP continues.

User or Operator Response: Send a PING to the indicated host to determine if it is reachable through the network. Check the *hlq.SEZAINST* data set to make sure that recursion has been requested. Recursion will allow the Domain Name Server for the local zone to communicate with other Domain Name Servers to determine the requested address. If the error persists, notify the system programmer. For more information about recursion, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: Make sure that the Domain Name Server is started and online. If necessary, update the address tables for the Domain Name Server to include the indicated host.

Source Data Set: NSLOOKUP

Procedure Name: DoLookup

EZB3009I *** Request to server timed-out

Explanation: A request to the indicated server reached the end of its time to live. The request is not answered.

System Action: TCPIP continues.

User or Operator Response: Resubmit the request. If the error persists, notify the system programmer.

System Programmer Response: Check the indicated server to determine why it is not answering requests.

Source Data Set: NSLOOKUP

Procedure Name: DoLookup

EZB3010I *** Server can't find host: *reason*

Explanation: The Domain Name Server is unable to find an address for the indicated host. The reason is displayed in the message.

System Action: TCPIP continues.

User or Operator Response: Notify the system programmer of the problem.

System Programmer Response: Respond as indicated by the *reason* portion of the message.

Source Data Set: NSLOOKUP

Procedure Name: DoLookup

EZB3011E *** Can't open file for writing

Explanation: The Domain Name Server is unable to open a file to write the answer to a request. The request is not answered.

System Action: TCPIP continues.

User or Operator Response: Notify the system programmer of the error.

System Programmer Response: Make sure that the Domain Name Server has the proper write authority.

Source Data Set: NSLOOKUP

Procedure Name: LookupHost, LookupHostWithServer

EZB3012I > *request*

Explanation: This message is written to the output file. It echoes the request to the Domain Name Server that produced the output file.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NSLOOKUP

Procedure Name: LookupHost, LookupHostWithServer

EZB3013W *** Can't find address for server *server: reason*

Explanation: The default Domain Name Server was unable to get information about the requested Domain Name Server. The reason for the error is displayed in the message. The request is not answered.

System Action: TCPIP continues.

User or Operator Response: Notify the system programmer of the error.

System Programmer Response: Respond as indicated by the *reason* portion of this message.

Source Data Set: NSLOOKUP

Procedure Name: LookupHostWithServer

EZB3014E *** Invalid set command

Explanation: The SET subcommand, which is used to set or change the options for the NSLOOKUP command, was submitted with no associated option. The SET subcommand is not accepted.

System Action: TCPIP continues.

User or Operator Response: If the NSLOOKUP command was submitted in interactive mode, resubmit the command specifying a valid option after the SET subcommand. To accept the current options to the NSLOOKUP command, omit the SET subcommand. If the NSLOOKUP command was submitted from the *user_id*.NSLOOKUP.ENV data set, edit the data set, including a valid option after the SET subcommand to change the NSLOOKUP options, or deleting the SET subcommand to accept the current option settings. For more information on the SET subcommand, the options to the NSLOOKUP command, and the *user_id*.NSLOOKUP.ENV data set, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: Assist the user as necessary.

Source Data Set: NSLOOKUP

Procedure Name: SetOption

EZB3015I d2 mode disabled; still in debug mode

Explanation: The NO D2 option has been accepted, disabling the high-level tracing for the Domain Name Server. The server is still in debug mode.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NSLOOKUP

Procedure Name: SetOption

EZB3016E invalid port value: *port*

Explanation: The PORT option to the SET subcommand, which specifies the port to use when contacting the Domain Name Server, was submitted with an incorrect value for the port. The given value was either out of the range of valid ports, or the value of a port that is already assigned. The PORT option is not accepted.

System Action: TCPIP continues.

User or Operator Response: Resubmit the NSLOOK command, specifying a valid port in the PORT option to the SET subcommand. The Domain Name Server is a well-known service, and is allocated port 53. For more information about the NSLOOKUP command, its subcommands and options, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: Assist the user as necessary.

Source Data Set: NSLOOKUP

Procedure Name: SetOption

EZB3017E invalid retry value: *value*

Explanation: The RETRY option to the SET subcommand of the NSLOOKUP command was submitted with an incorrect value. The NSLOOKUP command is not accepted. The RETRY option specifies the number of times a request is resent. If the RETRY option is set to 0, no requests are sent, resulting in the error message "no response from server".

System Action: TCPIP continues.

User or Operator Response: Resubmit the NSLOOKUP command with a valid value for the RETRY option of the SET subcommand. For more information about the NSLOOKUP command and its subcommands and options, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: Assist the user as necessary.

Source Data Set: NSLOOKUP

Procedure Name: SetOption

EZB3018E invalid timeout value: *value*

Explanation: The TIMEOUT option to the SET subcommand of the NSLOOKUP command was submitted with an incorrect value. The NSLOOKUP command is not accepted. The TIMEOUT option specifies the number of seconds to wait before canceling a request. If the TIMEOUT option is set to 0, requests are canceled immediately and are never answered.

System Action: TCPIP continues.

User or Operator Response: Resubmit the NSLOOKUP command with a valid value for the RETRY option of the SET subcommand. For more information about the NSLOOKUP command and its subcommands and options, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: Assist the user as necessary.

Source Data Set: NSLOOKUP

Procedure Name: SetOption

EZB3019E * Invalid option: *option***

Explanation: The SET subcommand was submitted with an option that the Domain Name Server did not recognize. The command is not accepted.

System Action: TCPIP continues.

User or Operator Response: Check the spelling and syntax and resubmit the command. For more information about valid options for the SET subcommand, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: Assist the user as necessary.

Source Data Set: NSLOOKUP

Procedure Name: SetOption

EZB3020I **Set options:**

EZB3021I *nodebug nodefname nosearch norecurse*

EZB3025I *nod2 novc noignoretc port=port*

EZB3029I **querytype=type class=class timeout=number retry=number**

EZB3033I **root=server**

EZB3034I **domain=domain**

EZB3035I *nobrackets*

EZB3194I *diff/time/nostamp*

EZB3036I **srchlist=domain1/domain2/domain3**

Explanation: These messages display the state information used by the resolver library and other options set by the user. The prefix *no* to an option name indicates that the option is not selected. For more information about these options, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NSLOOKUP

Procedure Name: ShowOptions

EZB3038E ***** Can't initialize resolver.**

Explanation: The Domain Name Server was unable to initialize the resolver library routines, which are used by clients to request resolution by the Domain Name Server. This indicates that there was an error in specifying the configuration file for the resolver, the RESOLVER address space was not started, or the Domain Name Server was unable to allocate the configuration data set. See the *z/OS Communications Server: IP Configuration Reference* for an explanation of the search order for the configuration data set.

System Action: Initialization halts. TCPIP continues.

User or Operator Response: Check the syntax of the configuration data set, start the RESOVLER address space, and resubmit the command. If the error persists, specify the configuration options using the NSLOOKUP command and its parameters. For more information about the configuration data set, refer to *z/OS Communications Server: IP Configuration Reference*. For more information about the NSLOOKUP command and its parameters, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: Make more storage available to the Domain Name Server if necessary.

Source Data Set: NSLOOKUP

Procedure Name: Main

EZB3039E * Can't find server address for 'server':**

Explanation: While initializing the resolver library used by the clients to request domain name resolution, the Domain Name Server was unable to find an IP address for the indicated server.

System Action: Initialization halts. TCPIP continues.

User or Operator Response: Correct the configuration data set to include an IP address for the default Domain Name Server. For more information about the DNS configuration data set, refer to *z/OS Communications Server: IP Configuration Reference*.

System Programmer Response: Assist the user as necessary.

Source Data Set: NSLOOKUP

Procedure Name: ReadRC

EZB3040E * Can't find server name for address *address*: *reason***

Explanation: While initializing the resolver library, which is used by the client to request domain name resolution, the Domain Name Server was unable to find a server name for the indicated address.

System Action: Initialization halts. TCPIP continues.

User or Operator Response: Correct the TCPIP.DATA data set to include a NSINTERADDR statement for the IP address of the Domain Name Server, or enter the NSLOOKUP command, using the *server_name* and *server_address* parameters to specify the default Domain Name Server. For more information about the Domain Name Server configuration and the steps involved, refer to *z/OS Communications Server: IP Configuration Reference*. For more information about the NSLOOKUP command and its parameters, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: Assist the user as necessary.

Source Data Set: NSLOOKUP

Procedure Name: Usage

EZB3041E * Default servers are not available**

Explanation: The Domain Name Servers specified as the default servers in the resolver library used by the client to request address resolution by the Domain Name Server are not available.

System Action: Initialization halts. TCPIP continues.

User or Operator Response: Use the NSLOOKUP command with the *server_name* and *server_address* parameters to specify a different default server. If the error persists, notify the system programmer.

System Programmer Response: Make sure that the indicated Domain Name Servers are started and online.

Source Data Set: NSLOOKUP

Procedure Name: Usage

EZB3042I > *command*

Explanation: This message is the command prompt for interactive NSLOOKUP sessions. For more information about interactive sessions using NSLOOKUP, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NSLOOKUP

Procedure Name: Usage

EZB3071W (name truncated?)

Explanation: This message indicates a compression error in the resource records generated by a name server query.

System Action: TCPIP continues.

User or Operator Response: Notify the system programmer.

System Programmer Response: Check the indicated Domain Name Server to determine the cause of the compression error. For more information on resource records and the Domain Name Server, refer to *z/OS Communications Server: IP Configuration Reference*.

Source Data Set: BD@DEBUG

Procedure Name: Print_rr

EZB3078I address, class = number, len = length

Explanation: This message is sent to the trace file if the D2 option was specified for the NSLOOKUP command. It displays the class and length of an address type resource record generated by a name server query.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: BD@DEBUG

Procedure Name: Print_rr

EZB3089I origin = name

EZB3090I mail addr = name

EZB3091I serial = serial_number (name)

EZB3092I refresh = refresh_time (time)

EZB3093I retry = retry_time (time)

EZB3094I expire = time_to_live (time)

EZB3095I minimum ttl = min_time_to_live (time)

Explanation: These messages are sent to the trace file if the DEBUG option was specified. They give information about the options selected for the Domain Name Server. For more information on the Domain Name Server, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: BD@DEBUG

Procedure Name: Print_rr

EZB3097E **errors = *name***

Explanation: This message indicates the host to which errors will be sent if they are encountered by the Domain Name Server.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: BD@DEBUG

Procedure Name: Print_rr

EZB3105I **NULL (*dlen length*)**

Explanation: This message is sent to the trace file if the DEBUG option was specified. It indicates that a resource record generated by a name server request has a type of null and the indicated length.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: BD@DEBUG

Procedure Name: Print_rr

EZB3106E **??? unknown type *type* ???**

Explanation: This message is sent to the trace file if the DEBUG option was specified. It indicates that a resource record generated by a name server request has a type that the Domain Name Server does not recognize. The type is displayed in the message.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: Check the indicated Domain Name Server to determine why it is generating incorrect resource records.

Source Data Set: BD@DEBUG

Procedure Name: Print_rr

EZB3108E ***** Error: record size incorrect (*actual_record_size* != *stated_record_size*)**

Explanation: This message is sent to the trace file if the DEBUG option was specified. It indicates that the Domain Name Server received a record that is not equal to the record size set in the *hlq*.PROFILE.TCPIP data set.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: Change the record sizes in the *hlq*.PROFILE.TCPIP data set so that the stated record size matches the actual record size.

Source Data Set: BD@DEBUG

Procedure Name: Print_rr

EZB3110I **Non-authoritative answer:**

Explanation: The name for which the client requested resolution is outside the current Domain Name Server's zone of authority, and recursive resolution was not requested. The Domain Name Server returns the name of the name server most likely to be able to resolve the request. The client issues a request to this name server.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: BD@GET

Procedure Name: FreeHostInfoPtr

EZB3112I **Authoritative answers can be found from:** *server(s)*

Explanation: The addresses requested by the client are outside the current Domain name server's zone of authority, and no recursive resolution was requested. The current Domain Name Server returns the names of the servers most likely to have information about the names being requested.

System Action: TCPIP continues.

User or Operator Response: Query the indicated name servers for more information.

System Programmer Response: Assist the user as necessary.

Source Data Set: BD@GET

Procedure Name: FreeHostInfoPointer

EZB3113I **Aliased to** *host*

Explanation: The address requested by the client is not in the zone of authority for the current Domain Name Server. The name server returns the name, or alias, of the name server most likely to have the address in its domain.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: BD@GET

Procedure Name: GetHostByName

EZB3116E ***** Is: invalid request** *request*

Explanation: The Domain Name Server received a request it did not recognize. The Domain Name Server is unable to process this request. The request is displayed in this message.

System Action: TCPIP continues.

User or Operator Response: Check the syntax and resubmit the request.

System Programmer Response: Assist the user as necessary.

Source Data Set: BD@LIST

Procedure Name: ListHostsByType

EZB3117I ***** Can't list domain** *domain: reason*

Explanation: This message is sent to the trace file. The procedure ListHostsByType, which lists the hosts known to the Domain Name Server, was unsuccessful for the indicated reason.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: Respond as indicated by the *reason* portion of this message.

Source Data Set: BD@LIST

Procedure Name: ListHostsByType

EZB3119I [host]**Explanation:** This message displays the name of the local host.**System Action:** TCPIP continues.**User or Operator Response:** None.**System Programmer Response:** None.**Source Data Set:** BD@LIST**Procedure Name:** ListHostsByType

EZB3120E *** Can't open *file* for writing**Explanation:** The function OpenFile, which parses the command string for a file name and opens the file for writing, was unsuccessful, indicating an incorrect command string or an error opening the file. The data set is not opened.**System Action:** TCPIP continues.**User or Operator Response:** Check the syntax and resubmit the command. If the error persists, notify the system programmer.**System Programmer Response:** Make sure that the user has the proper authority to write the indicated data set.**Source Data Set:** BD@LIST**Procedure Name:** ListHosts

EZB3121I >command**Explanation:** This message is written to a file opened by the function OpenFile. It echoes the last command submitted by the user.**System Action:** TCPIP continues.**User or Operator Response:** None.**System Programmer Response:** None.**Source Data Set:** BD@LIST**Procedure Name:** ListHosts

EZB3122I Alias**Explanation:** This message precedes the alias of a host. Hosts can be addressed by their alias, their network address, or their canonical name. For more information about aliases, see the *z/OS Communications Server: IP User's Guide and Commands*.**System Action:** TCPIP continues.**User or Operator Response:** None.**System Programmer Response:** None.**Source Data Set:** BD@LIST**Procedure Name:** ListHosts

EZB3124I Received *number* records.**Explanation:** The Domain Name Server received the indicated number of records in response to a query.**System Action:** TCPIP continues.**User or Operator Response:** None.**System Programmer Response:** None.

Source Data Set: BD@LIST

Procedure Name: ListHosts

EZB3125E * ls: error receiving zone transfer:**

EZB3126I result: result, answers = number, authority = number, additional = number

Explanation: The client was unable to read the response from an LS command, used to list the name servers in other domains known to the local Domain Name Server. The data from the LS command is not transferred.

System Action: TCPIP continues.

User or Operator Response: Notify the system programmer of the error.

System Programmer Response: Check the Domain Name Server to make sure that it is properly configured to be compatible with the other Domain Name Servers on the system. For more information on configuring the Domain Name Server, refer to *z/OS Communications Server: IP Configuration Reference*.

Source Data Set: BD@LIST

Procedure Name: ListHosts

EZB3127I domain_name

Explanation: This message displays the domain name for which an answer returned from the name server is valid.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: BD@LIST

Procedure Name: PrintListInfo

EZB3131I (address_protocol_identifiers

Explanation: This message indicates the Internet address protocol being used by NSLOOKUP. For more information about Internet address protocols, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: BD@LIST

Procedure Name: PrintListInfo

EZB3132I (dlen = length?)

Explanation: This message indicates the length of data in a response received by NSLOOKUP.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: BD@LIST

Procedure Name: PrintListInfo

EZB3133I ***

Explanation: This message precedes other informational messages.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: BD@LIST

Procedure Name: PrintListInfo

EZB3134I *server, type class*

Explanation: This message displays the name of the server being queried, the type of query, and the class of address being requested.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: BD@DEBUG

Procedure Name: Print_cdname_sub

EZB3143I *text data*

Explanation: This is variable data which is displayed in response to the NSLOOKUP command. It is defined by the installation in the name server data base.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: BD@DEBUG

Procedure Name: Print_rr

EZB3144I *number*

Explanation: This message displays the user or group ID from a name server query.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: BD@LIST

Procedure Name: PrintListInfo

EZB3145I *number*

Explanation: This message displays the protocol value taken from a WKS resource record. For more information about resource records, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: BD@LIST

Procedure Name: PrintListInfo

EZB3146 *protocol*

Explanation: This message displays the mnemonic for the protocol value taken from a well known services (WKS) resource record. For more information about resource records, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: BD@LIST

Procedure Name: PrintListInfo

EZB3150E *** Can't open *data_set* for reading

Explanation: The Domain Name Server was unable to read from the indicated data set. The Domain Name Server query is not answered.

System Action: TCPIP continues.

User or Operator Response: Make sure the indicated data set is in storage accessible to the Domain Name Server.

System Programmer Response: Assist the user as necessary.

Source Data Set: BD@LIST

Procedure Name: ViewList

EZB3151I *host*

Explanation: This message displays the output of the LS function.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: BD@LIST

Procedure Name: ViewList

EZB3154I **Finger: no current host defined.**

Explanation: No host has been defined to the current finger server, which displays information about users of a remote host. The FINGER query is not answered.

System Action: TCPIP continues.

User or Operator Response: Resubmit the FINGER command, using the format FINGER@host to define the remote host to be used to complete the request.

System Programmer Response: Assist the user as necessary.

Source Data Set: BD@LIST

Procedure Name: Finger

EZB3155I **Finger: unknown service**

Explanation: The finger server, which gives information about the users of a remote host, received a request for a service that it does not recognize. The request is not processed.

System Action: TCPIP continues.

User or Operator Response: Check the syntax and resubmit the command. For more information about the finger server, see the *z/OS Communications Server: IP User's Guide and Commands*.

EZB3156I • EZB3175I

System Programmer Response: Assist the user as necessary.

Source Data Set: BD@LIST

Procedure Name: Finger

EZB3156I > *data_set_name*

Explanation: This message displays the name of the data set to which information returned by the finger server will be sent.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: BD@LIST

Procedure Name: Finger

EZB3169E *** Can't allocate memory

Explanation: The Domain Name Server was unable to allocate storage to hold lookup tables because no data set name was specified.

System Action: TCPIP continues.

User or Operator Response: Check the syntax and resubmit the command. If the error persists, notify the system programmer.

System Programmer Response: Make sure that sufficient storage is available for the tables needed by the Domain Name Server. If the error persists, check for allocation errors on the server.

Source Data Set: BD@SUBR

Procedure Name: Malloc

EZB3170I **Server:** *server*

EZB3171I **Addresses:** *addresses*

EZB3172I **Address:** *address*

Explanation: These messages give the name and addresses of the server being queried.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: BD@SUBR

Procedure Name: PrintHostInfo

EZB3175I **Aliases:** *alias_names*

Explanation: This message is displayed with message EZB3170I and gives the alias names corresponding to the indicated addresses.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: BD@SUBR

Procedure Name: PrintHostInfo

EZB3176I **Served by:**

EZB3177I *servers*

EZB3178I *, internet_address*

Explanation: These messages are displayed with message EZB3170I. They list the name servers for the indicated addresses.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: BD@SUBR

Procedure Name: PrintHostInfo

EZB3181W **unknown query class:** *class*

Explanation: The Domain Name Server received a query of a class it does not recognize. Repeated queries of this class will tie up the name server, preventing it from replying to valid queries. The query is not processed.

System Action: TCPIP continues.

User or Operator Response: Notify the system programmer of the problem.

System Programmer Response: Check the client to determine why it is sending incorrect queries. Take the client offline until the problem is solved.

Source Data Set: BD@SUBR

Procedure Name: StringToClass

EZB3182W **unknown query type:** *type*

Explanation: The Domain Name Server received a query of a type it does not recognize. The query cannot be processed, and too many queries of this type will tie up the name server, preventing it from replying to valid queries. The query is not processed.

System Action: TCPIP continues.

User or Operator Response: Notify the system programmer of the problem.

System Programmer Response: Check the indicated client to determine why it is sending queries of an incorrect type. Take the client offline until the problem is corrected.

Source Data Set: BD@SUBR

Procedure Name: StringToType

EZB3194I *typestamp*

Explanation: This message indicates the type of time stamping selected for the NSLOOKUP command. The possible values are:

Value Indicates

time The time is displayed before each output line.

diff The time is displayed before each output line only when the time changes.

no No time stamping is selected.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NSLOOKUP

Procedure Name: ShowOptions

EZB3195I *[host]*

Explanation: This message displays the name of the local host.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: BD@LIST

Procedure Name: ListHostsByType

EZB3196W **no DOMAIN is entered, using default**

Explanation: The Domain Name Server received a command that does not specify a domain. The name server uses the default domain.

System Action: TCPIP continues.

User or Operator Response: Reenter the command, specifying the correct domain if necessary.

System Programmer Response: Assist the user as necessary.

Source Data Set: NSLOOKUP

Procedure Name: SetOptions

EZB3197W **no ROOT is entered, using default**

Explanation: The Domain Name Server received a command that did not specify the root name server to be used. The Domain Name Server uses the default server.

System Action: TCPIP continues.

User or Operator Response: Resubmit the command, specifying the root server if necessary.

System Programmer Response: Assist the user as necessary.

Source Data Set: NSLOOKUP

Procedure Name: SetOptions

EZB3198W **no SRCHLIST is entered, using default**

Explanation: The Domain Name Server received a command that did not specify the search list to be used. The Domain Name Server uses the default search list.

System Action: TCPIP continues.

User or Operator Response: Resubmit the command, specifying the search list if necessary.

System Programmer Response: Assist the user as necessary.

Source Data Set: NSLOOKUP

Procedure Name: SetOptions

EZB3201I **text = *text***

Explanation: This message displays the contents of a text type query.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: BD@DEBUG

Procedure Name: Print_rr

EZB3202E *** Syntax error in *option option*.

Explanation: The indicated option of the NSLOOKUP command was submitted with incorrect syntax. The option is ignored.

System Action: TCPIP continues.

User or Operator Response: Resubmit the NSLOOKUP command with the proper syntax for all options. For more information about the NSLOOKUP command and the proper syntax for its options, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: Assist the user as necessary.

Source Data Set: NSLOOKUP

Procedure Name: comml_option

EZB3203E *** The '=' sign is missing in *option option*

Explanation: The indicated option of the NSLOOKUP command was submitted without a '='. The option cannot be processed.

System Action: TCPIP continues.

User or Operator Response: Resubmit the NSLOOKUP command with the proper syntax for all options. For more information about the NSLOOKUP command and its options, see the *z/OS Communications Server: IP User's Guide and Commands*.

System Programmer Response: Assist the user as necessary.

Source Data Set: NSLOOKUP

Procedure Name: comml_option

EZB3205I *local host*

Explanation: This is the name of the local host or the name of a server that knows about the local domain.

System Action: NSLOOKUP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: BD@LIST

Procedure Name: ListSubr

EZB3206I nslookup [-option ...] [host-to-find | - [server]]

EZB3207I Commands: (identifiers are shown in uppercase, <> means optional)

EZB3208I NAME - print info about the host/domain NAME using

EZB3209I default server

EZB3210I NAME1 NAME2 - as above, but use NAME2 as server

EZB3211I	help or ? - print info on common commands;
EZB3212I	use TSO's help nslookup for more details
EZB3213I	set OPTION - set an option
EZB3214I	all - print options, current server and host
EZB3215I	<no>debug - print debugging information
EZB3216I	<no>d2 - print exhaustive debugging information
EZB3217I	<no>defname - append domain name to each query
EZB3218I	<no>recurse - ask for recursive answer to each query
EZB3219I	<no>vc - always use a virtual circuit
EZB3220I	domain=NAME - set default domain name to NAME
EZB3221I	srchlist=N1</N2/.../N6> - set domain to N1 and search
EZB3222I	list to N1,N2, etc.
EZB3223I	root=NAME - set root server to NAME
EZB3224I	retry=X - set number of retries to X
EZB3225I	timeout=X - set initial time-out interval to X seconds
EZB3226I	querytype=X - set query type, e.g.,
EZB3227I	A,ANY,CNAME,HINFO,MC,NS,PTR,SOA,WKS
EZB3228I	type=X - synonym for querytype
EZB3229I	class=X - set query class to one of IN (Internet),
EZB3230I	CHAOS, HESIOD or ANY
EZB3231I	server NAME - set default server to NAME, using current
EZB3232I	default server

EZB3233I lserver NAME - set default server to NAME, using initial server

EZB3234I finger <USER> - finger the optional NAME at the current

EZB3235I default host

EZB3236I root - set current default server to the root

EZB3237I ls <opt> DOMAIN [> DATASET] - list addresses in DOMAIN

EZB3238I (optional: output to DATASET)

EZB3239I -a - list canonical names and aliases

EZB3240I -h - list HINFO (CPU type and operating system)

EZB3241I -s - list well-known services

EZB3242I -d - list all records

EZB3243I -t TYPE - list records of the given type

EZB3244I (e.g., A, CNAME, MX, etc.)

EZB3245I view DATASET - sort an 'ls' output file and view it with more

EZB3246I exit - exit the program

Explanation: This is the summary of available commands and options that you can review by typing help or ? at the > prompt (see message EZB3042I), once you have entered the NSLOOKUP interactive session. For details about any of these commands or options, see *z/OS Communications Server: IP User's Guide and Commands*.

System Action: The name server continues until you type exit at the > prompt (see message EZB3042I).

User or Operator Response: Use these help messages as a guide for entering subsequent NSLOOKUP commands.

System Programmer Response: None.

Source Data Set: NSLOOKUP

Procedure Name: PrintHelp()

EZB3250I invalid dig option *option*

Explanation: The option indicated was entered at the command line and is not a valid DIG option.

System Action: TCPIP continues.

User or Operator Response: Enter a valid DIG option. *z/OS Communications Server: IP User's Guide and Commands* contains the DIG command syntax and the valid options.

System Programmer Response: None.

Source Data Set: DIG

Procedure Name: PrintHelp

EZB3251I no dig -T value specified

Explanation: No wait time (-T value) was specified. Wait time is the time to wait between successive queries when operating in batch mode. The default wait time is 0, which indicates no wait time.

System Action: TCPIP continues.

User or Operator Response: If you wish to specify a wait time other than 0, issue the DIG command with the -T option. See *z/OS Communications Server: IP User's Guide and Commands* for the syntax of the DIG command. Otherwise, no action is necessary.

System Programmer Response: None.

Source Data Set: DIG

Procedure Name: PrintHelp

EZB3252I invalid dig -T value tvalue

Explanation: The -T value specified on the DIG command is not valid.

System Action: TCPIP continues.

User or Operator Response: Specify a -T value in seconds. 0 is the default.

System Programmer Response: None.

Source Data Set: DIG

Procedure Name: PrintHelp

EZB3253I ; invalid class specified

Explanation: The network class specified on the DIG command is not valid.

System Action: TCPIP continues.

User or Operator Response: Specify a valid network class on the DIG command. DIG recognizes only the IN, CHAOS, HESIOD, and ANY network classes.

System Programmer Response: None.

Source Data Set: DIG

Procedure Name: PrintHelp

EZB3254I ; invalid type specified

Explanation: The query type specified on the DIG command is not valid.

System Action: TCPIP continues.

User or Operator Response: Specify a valid query type, *qtype* on the DIG command. See *z/OS Communications Server: IP User's Guide and Commands* for information about the DIG command and valid query types.

System Programmer Response: None.

Source Data Set: DIG

Procedure Name: PrintHelp

EZB3255I Missing batch file name

Explanation: The -f option was specified on the DIG command with no data set name.

System Action: TCPIP continues.

User or Operator Response: Specify a batch data set name on the -f option of the DIG command and reissue the command. *z/OS Communications Server: IP User's Guide and Commands* contains information about the DIG command and -f option.

System Programmer Response: None.

Source Data Set: DIG

Procedure Name: PrintHelp

EZB3256I invalid dig -x option *option*

Explanation: The -x option specified on the DIG command is not valid.

System Action: TCPIP continues.

User or Operator Response: Specify a valid dotted decimal notation IP address. *z/OS Communications Server: IP User's Guide and Commands* describes the DIG command and the -x option.

System Programmer Response: None.

Source Data Set: DIG

Procedure Name: PrintHelp

EZB3257I no dig -p value specified

Explanation: The DIG command -p option was specified with no value. The port number given when contacting the name server must be specified with the -p option.

System Action: TCPIP continues.

User or Operator Response: Specify the port number given when contacting the name server on the DIG command with the -p option. The default for the Domain Name Server is 53. The -p option allows you to override this default.

System Programmer Response: None.

Source Data Set: DIG

Procedure Name: PrintHelp

EZB3258I invalid dig -p value *pvalue*

Explanation: The -p value specified on the DIG command is not valid.

System Action: TCPIP continues.

User or Operator Response: Specify a valid port number on the DIG -p option and reissue the command. Use the decimal port number given when contacting the name server.

System Programmer Response: None.

Source Data Set: DIG

Procedure Name: PrintHelp

EZB3259I ; invalid type/class specified

Explanation: The query type or query class specified on the DIG command is not valid.

System Action: TCPIP continues.

User or Operator Response: Specify a valid query class and query type on the DIG command. See *z/OS Communications Server: IP User's Guide and Commands* for information about valid classes and types.

System Programmer Response: None.

Source Data Set: DIG

Procedure Name: PrintHelp

EZB3260I ; pflag: *pfcode* res: *resvalue*

Explanation: This message displays the print flag values and the resolver options that are set for the request.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG

Procedure Name: PrintHelp

EZB3262I ; Bad server: *servername* - - using default server and timer opts

Explanation: The server specified is not a recognized server. DIG is using the default server and the associated timer options instead.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG

Procedure Name: PrintHelp

EZB3263I ;; FROM: *hostname* to SERVER: *servername*

Explanation: Messages are being sent from the host indicated to the server indicated.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG

Procedure Name: PrintHelp

EZB3264I ;; WHEN: *time*

Explanation: The day, date, and time of the DIG request is displayed.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG

Procedure Name: PrintHelp

EZB3265I ;; re_mkquery: buffer too small

Explanation: The storage buffer is too small to complete the query request.

System Action: TCPIP continues.

User or Operator Response: Define a larger buffer and reissue the DIG query request.

System Programmer Response: None.

Source Data Set: DIG

Procedure Name: PrintHelp

EZB3271I ; MSG SIZE sent *sendsize* rcvd: *recvsize*

Explanation: The size of the message when sent from the host and as received by the server is displayed.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG

Procedure Name: PrintHelp

EZB3280I ; Matching SOA found

Explanation: The authority record being searched for was located.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@LIST

Procedure Name: do_zone

EZB3281I ; ListHosts: error receiving zone transfer:

Explanation: An error occurred during a zone transfer between a primary and a secondary name server.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center to report this error.

Source Data Set: DIG@LIST

Procedure Name: do_zone

EZB3282I ; result: *text*, answers = *value*, authority = *value*

Explanation: This message displays with message EZB3281I and indicates the result of the zone transfer attempt, information from the answer section of the response, and the address of the authoritative name server for the response.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@LIST

Procedure Name: do_zone

EZB3283X additional= *number*

Explanation: This message appears with message EZB3282I. Additional resources records that have not been requested, but might be useful, are displayed.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@LIST

Procedure Name: do_zone

EZB3284I ; ***Error during listing of *domain*

Explanation: An error occurred during the listing of information about the domain name indicated in the message.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center to report this error.

Source Data Set: DIG@LIST

Procedure Name: do_zone

EZB3286I ; *** Invalid option: *option*

Explanation: A DIG command option that was specified is not valid.

System Action: TCPIP continues.

User or Operator Response: Check the command you entered to be sure you specified the command with the correct syntax. *z/OS Communications Server: IP User's Guide and Commands* describes the DIG command and its syntax.

System Programmer Response: None.

Source Data Set: DIG@OPT

Procedure Name: SetOption

EZB3287I invalid timeout value: *value*

Explanation: The time-out value specified on the DIG command is not a valid value. The time-out value specifies the number of seconds to wait before timing out of a request.

System Action: TCPIP continues.

User or Operator Response: Specify a decimal digit value for the time-out value using the DIG command.

System Programmer Response: None.

Source Data Set: DIG@OPT

Procedure Name: SetOption

EZB3288I invalid retry value: *limit*

Explanation: The retry limit specified on the DIG command is not valid. The retry value specifies the number of times a request is sent.

System Action: TCPIP continues.

User or Operator Response: Specify a decimal digit value for the retry limit using the DIG command.

System Programmer Response: None.

Source Data Set: DIG@OPT

Procedure Name: SetOption

EZB3289I ; ***Bad char in numeric string -- ignored

Explanation: DIG found a nonnumeric character in a numeric string. DIG is ignoring the character.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@OPT

Procedure Name: SetOption

EZB3314X *service_name*

Explanation: This message displays the service name. This message is issued for debugging purposes.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@DEBU

Procedure Name: p_rr

EZB3316I **cms;; ->>HEADER<<-**

EZB3317X **opcode:** *code*

EZB3318X **, status** *status*

EZB3319X **, id:** *idname*

Explanation: These messages display in response to a DIG command. They indicate the operation type, for example, QUERY; the status of the request, for example, noerror, which says that the request was made correctly; and the ID of the queried Domain Name Server. If the DIG command is issued with no other parameters, this information is returned about the default server.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@DEBU

Procedure Name: fp_query

EZB3320I **;; flags:**

EZB3321X **qr**

EZB3322X **aa**

EZB3323X **tc**

EZB3324X **rd**

EZB3325X **ra**

EZB3326X **pr**

EZB3327X **res_opts:** *options*

Explanation: Some combination of the above flags are displayed to indicate which options are set on. The flags and their meanings are as follows:

Flag	Meaning
------	---------

qr	Prints the outgoing query.
----	----------------------------

aa	Accepts only authoritative responses to queries.
----	--

EZB3328I • EZB3335X

tc Truncated.
rd Recursion desired.
ra Recursion available.
pr Uses only the primary name server for the zone.

res_opts
resolver options

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@DEBU

Procedure Name: fp_query

EZB3328I **:: Ques:** *value,*

EZB3329I **Ans:** *value,*

EZB3330I **Auth:** *value,*

EZB3331I **Addit:** *value,*

Explanation: This message displays the values of the question, answer, authoritative, and additional sections of the response. The question section contains the original query; the answer section contains the set of all resource records from the name server database that satisfy the query; the authoritative section contains resource records that specify the address of an authoritative name server for the query; and the additional section contains resource records that have not been explicitly requested, but could be useful.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@DEBU

Procedure Name: fp_query

EZB3332I **:: QUESTIONS:**

EZB3333X **::**

EZB3334X **, type =** *type*

EZB3335X **, class =** *class*

Explanation: These messages prints the question records as well as the type of query requested and the network class requested.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@DEBU

Procedure Name: p_rr

EZB3336X *timetolive*

Explanation: This is the time-to-live (TTL) value for the resource record. TTL is the number of seconds that a record is valid in a cache.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@DEBU

Procedure Name: p_rr

EZB3337I *class type*

Explanation: This is the network class requested in the query and the type of query to be performed.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@DEBU

Procedure Name: p_rr

EZB3338X *type*

Explanation: This is the type of query to be performed.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@DEBU

Procedure Name: p_rr

EZB3339I *;; proto: number*

EZB3340X *, port: number*

Explanation: These messages indicate the network protocol and the port number for the domain name identified by the address record.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@DEBU

Procedure Name: p_rr

EZB3341X *string*

EZB3342X *string*

Explanation: These messages indicate the central processing unit type and the operating system of a node. This information is part of the host information record.

System Action: TCPIP continues.

User or Operator Response: None.

EZB3343X • EZB3349X

System Programmer Response:

Source Data Set: DIG@DEBU

Procedure Name: p_rr

EZB3343X *(serialvalue ;serial*

EZB3344X *refreshvalue ;refresh*

EZB3345X *retryvalue ;retry*

EZB3346X *expirevalue ;expire*

EZB3347X *minimvalue) ;minim*

Explanation: These messages are part of the authoritative section of the response to a DIG query that requested authority records. The following list defines each of the values:

- Serial is the serial number of the zone database.
- Refresh is the refresh interval, or the length of time, in seconds, you must allow between the refreshing of a database from a remote name server.
- Retry is the retry interval that indicates the length of time, in seconds, you must allow before retrying a failed refresh.
- Expire is the expiration time-to-live that indicates the maximum time for records to be valid in the zone database.
- Minim is the minimum time-to-live that indicates the minimum time for records to be valid in the zone database.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@DEBU

Procedure Name: p_rr

EZB3348X *term*

Explanation: This message identifies a host that can act as a mail exchange for the domain specified in the domain name field. A mail exchange runs a mail agent that delivers or forwards mail for the domain name specified in the first field of the resource record.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@DEBU

Procedure Name: p_rr

EZB3349X *term*

Explanation: This message displays the group ID associated with the resource record.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@DEBU

Procedure Name: p_rr

EZB3350X *string string (*

Explanation: This message indicates the IP address and protocol names for the resource record.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@DEBU

Procedure Name: p_rr

EZB3351X *string number (*

Explanation: This message indicates the protocol name and number for the resource record.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@DEBU

Procedure Name: p_rr

EZB3352I *number*

Explanation: This message indicates the number of protocol numbers for services stored in the well-known services record.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@DEBU

Procedure Name: p_rr

EZB3353X **First *number* bytes of hex data:**

Explanation: This message displays the number of bytes that are displayed following this message. This message is issued when debugging is requested.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@DEBU

Procedure Name: p_rr

EZB3354I *data*

Explanation: This is the data in a resource record that appears in an unspecified format (binary format). Message EZB3353X tells the number of bytes of the data that are displayed.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@DEBU

Procedure Name: p_rr

EZB3355X ; *defmetolive*

Explanation: This is the default time-to-live (TTL) value. TTL is the number of seconds that a record is valid in a cache.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@DEBU

Procedure Name: p_rr

EZB3356I ;; packet size error

Explanation: The query packet is not the standard size.

System Action: TCPIP continues.

User or Operator Response: Resubmit the query request.

System Programmer Response: None.

Source Data Set: DIG@DEBU

Procedure Name: p_rr

EZB3359I ;; ANSWERS:

Explanation: This message precedes the display of records in the answers section of a query response.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@DEBU

Procedure Name: fp_query

EZB3360I ;; AUTHORITY RECORDS:

Explanation: This message precedes the display of records in the authoritative section. These resource records specify the address of an authoritative name server for the query.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@DEBU

Procedure Name: fp_query

EZB3361I ;; ADDITIONAL RECORDS:

Explanation: This message precedes the display of records in the additional section of a query response.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@DEBU

Procedure Name: fp_query

EZB3362I ; <<>>DiG *version* <<>>

Explanation: This messages displays the version of the Domain Information Groper (DIG) currently in use on the system.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG

Procedure Name: main

EZB3363X *defwell-known-service*

Explanation: The service displayed is the default well-known-service name.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG@DEBU

Procedure Name: p_rr

EZB3364I *dig.help*

Explanation: The help command has been issued and the DIG.HELP data set is being opened.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG

Procedure Name: PrintHelp

EZB3365I *file open*

Explanation: The help data set is now open.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG

Procedure Name: main

EZB3366I - August 30, 1990

EZB3367I - *dig @server domain query-type query-class*

EZB3368I *+query-option -dig-option comment*

EZB3369I	server - a domain name or a dot-notation Internet address
EZB3370I	default: nsinteraddr parm address defined in
EZB3371I	dataset TCPIP.DATA
EZB3372I	DiG Defaults: Name server address in DIG.ENV dataset
EZB3373I	domain - request domain information for domain name
EZB3374I	query-type - a, any, cname, hinfo, mx, ns, ptr, soa, wks
EZB3375I	query-class - IN (Internet), CHAOS(obsolete), HESIOD
EZB3376I	<i>+query option</i>
EZB3377I	<no>debug (deb) turn on/off debugging mode <deb>
EZB3378I	<no>d2 turn on/off extra debugging mode <nod2>
EZB3379I	<no>recurse (rec) use/don't use recursive lookup <rec>
EZB3380I	retry=# (ret) set number of retries to # <4>
EZB3381I	time=# (ti) set timeout length to # seconds <4>
EZB3382I	<no>ko keep open option (implies vc) <noko>
EZB3383I	<no>vc use/don't use virtual circuit <novc>
EZB3384I	<no>defname use/don't use default domain name <novc>
EZB3385I	<no>search (sea) use/don't use domain search list <sea>
EZB3386I	domain=NAME (do) set default domain name to NAME
EZB3387I	<no>ignore (i) ignore/don't ignore trunc. errors <noi>
EZB3388I	<no>primary (pr) use/don't use primary server <nopr>
EZB3389I	<no>aonly (aa) authoritative query only flag <noaa>
EZB3390I	<no>sort (sor) sort resource records <nosor>

EZB3391I	<no>cmd echo parsed arguments <cmd>
<hr/>	
EZB3392I	<no>stats (st) print query statistics (RTT,etc) <st>
<hr/>	
EZB3393I	<no>Header (H) print basic header <H>
<hr/>	
EZB3394I	<no>header (he) print header flags <he>
<hr/>	
EZB3395I	<no>ttlid (tt) print TTLs <tt>
<hr/>	
EZB3396I	<no>cl print class info <nocl>
<hr/>	
EZB3397I	<no>qr print outgoing query <noqr>
<hr/>	
EZB3398I	<no>reply (rep) print reply <rep>
<hr/>	
EZB3399I	<no>ques (qu) print question section <qu>
<hr/>	
EZB3400I	<no>answer (an) print answer section <an>
<hr/>	
EZB3401I	<no>author (au) print authoritative section <au>
<hr/>	
EZB3402I	<no>addit (ad) print additional section <ad>
<hr/>	
EZB3403I	pfdef set to default print flags
<hr/>	
EZB3404I	pfmin set to minimal default print flags
<hr/>	
EZB3405I	pfset=# set print flags to #
<hr/>	
EZB3406I	(# can be hex/octal/decimal)
<hr/>	
EZB3407I	pfand=# bitwise and print flags with #
<hr/>	
EZB3408I	pfor=# bitwise or print flags with #
<hr/>	
EZB3409I	- dig option
<hr/>	
EZB3410I	-x dot-notation-address
<hr/>	
EZB3411I	-f file for dig batch mode
<hr/>	
EZB3412I	-T Time in seconds between start of successive queries

EZB3413I • EZB3433I

EZB3413I	-p Port number
EZB3414I	-P <i>ping-string</i>
EZB3415I	-t query-type
EZB3416I	-c query-class
EZB3417I	-envsav This flag specifies that the dig environment
EZB3418I	(defaults, print options, etc.), after all of
EZB3419I	the arguments are parsed, should be saved to a
EZB3420I	file to become the default environment. DiG.env
EZB3421I	is created in the current working directory.
EZB3422I	-envset This flag only affects batch query runs. When
EZB3423I	-envset is specified on a line in a dig batch file
EZB3424I	the dig environment after the arguments are parsed,
EZB3425I	becomes the default environment for the duration of
EZB3426I	the batch file, or until the next line which
EZB3427I	specifies -envset.
EZB3428I	-<no>stick This flag only affects batch query runs. It
EZB3429I	specifies that the dig environment (as read
EZB3430I	initially or set by -envset switch) is to be
EZB3431I	restored before each query (line) in a dig
EZB3432I	batch file.

EZB3433I %comment - included argument that is not parsed

Explanation: Messages EZB3364I - EZB3433I appear in response to the HELP DIG command. See *z/OS Communications Server: IP User's Guide and Commands* for more information about the DIG command.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: DIG

Procedure Name: PrintHelp()

EZB3434I Can not read DIG environmental data set. Defaults used.

Explanation: This message is displayed if a DIG command was invoked with the -envsav option from a Communications Server release earlier than CS for OS/390 V1R2. This option is not compatible with CS for OS/390 V1R2 or later releases.

System Action: No query options saved from the DIG command prior to CS for OS/390 V1R2 are restored to a CS for OS/390 V1R2 or later release of the DIG command.

User or Operator Response: To create an environmental data set in a CS for OS/390 V1R2 or later format, use the -envsav option with any DIG command that has the query options desired. See the *z/OS Communications Server: IP User's Guide and Commands* for information about the TSO DIG command.

System Programmer Response: none

Source Data Set: DIG

Procedure Name: main

EZB3483E Couldn't open output file

Explanation: NSDBLOAD encountered an error opening its output file. The program ends with exit code 99.

System Action: The name server exits.

User or Operator Response: Check that the name server has read/write access to the file.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: main

EZB3484E Error(s) occurred reading master datafile.

Explanation: NSDBLOADz encountered an error while processing the master data file.

System Action: The name server ends.

User or Operator Response: Check the data syntax of the master data file.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: NSDBLOAD

Procedure Name: main

EZB3487I Error occurred deleting table info.

Explanation: NSDBLOADz encountered an error deleting SQL table information.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: main

EZB3489I *sql statement* sqlcode sqlcode

Explanation: NSDBLOAD was unable to process the indicated SQL command.

System Action: NSDBLOAD continues.

User or Operator Response: Use the *sqlcode* that is displayed in this message and *DB2 Messages and Codes* to determine the cause of the error and respond as indicated.

EZB3490I • EZB3496W

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name:

EZB3490I *time: Delete all data from name table*

Explanation: NSDBLOADz is deleting all data from the indicated table.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: deleteinfo

EZB3491I *time: Deleted all data in sql table name*

Explanation: NSDBLOADz has deleted all data in the indicated SQL table.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: deleteinfo

EZB3493E **Couldn't read output resource record file**

Explanation: NSDBLOADz encountered an error opening its output file. The program ends with exit code 99.

System Action: The NSDBLOAD program ends.

User or Operator Response: Check for the name server's access to the file.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: insertsql

EZB3495W **error invalid class class**

Explanation: The NSDBLOAD program encountered an incorrect class specification in an input record. The record is ignored.

System Action: TCPIP continues.

User or Operator Response: Check for the correct CLASS of the defined resource record.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: insertsql

EZB3496W **error invalid type type**

Explanation: The NSDBLOAD program encountered an incorrect type specification in an input record.

System Action: The record is ignored.

User or Operator Response: Check for the correct TYPE of the defined resource record.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: insertsql

EZB3497I *time:* **Finished adding data to table**

Explanation: The NSDBLOAD program has finished adding data to its SQL table.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: insertsql

EZB3498I *time:* **Update statistics on the name table**

Explanation: The NSDBLOAD program updates the statistics on this table.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: insertsql

EZB3499I *time:* **Data base update completed.**

Explanation: The NSDBLOAD program has completed the database update.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: readmaster

EZB3501E **Couldn't read input master file**

Explanation: The NSDBLOAD program encountered an error opening the resource record file previously generated. The program ends with exit code 99.

System Action: The name server ends.

User or Operator Response: Check the read/write access on the input file for the current user ID.

System Programmer Response: Check for the existence of a resource record file.

Source Data Set: NSDBLOAD

Procedure Name: insertsql

EZB3503E **Origin definition error**

Explanation: The NSDBLOAD program encountered a \$ORIGIN statement in the input file without an accompanying definition.

System Action: TCPIP continues.

User or Operator Response: Correct the \$ORIGIN input statement to NSDBLOAD.

System Programmer Response: None.

EZB3504I • EZB3510W

Source Data Set: NSDBLOAD

Procedure Name: readmaster

EZB3504I *time: New origin origin*

Explanation: The NSDBLOAD program encountered a \$ORIGIN statement with the specified origin in its input file.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: readmaster

EZB3505I *time: @ - current origin origin*

Explanation: The NSDBLOAD program is using the specified origin.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: readmaster

EZB3506E **Origin has not been defined**

Explanation: A relative domain name defined in a master file requires a domain origin suffix.

System Action: TCPIP continues.

User or Operator Response: Correct the master file to include a \$ORIGIN *domain* statement.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: readmaster

EZB3509W **Format error 1: line**

Explanation: The NSDBLOAD program encountered a syntax error in the input file. The line causing the error is displayed and ignored.

System Action: TCPIP continues.

User or Operator Response: Check the TYPE field in the defined resource record.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: readMaster

EZB3510W **Format error 2: line**

Explanation: The NSDBLOAD program encountered a syntax error in the input file. The line causing the error is displayed and ignored.

System Action: TCPIP continues.

User or Operator Response: The required TYPE field is missing from the resource record.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: readMaster

EZB3511W Format error 3: line

Explanation: The NSDBLOAD program encountered a syntax error in the input file. The line causing the error is displayed and ignored.

System Action: TCPIP continues.

User or Operator Response: The required TYPE field is missing from the resource record.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: readMaster

EZB3512W Obsolete root definition replace with ‘.’

Explanation: The NSDBLOAD program encountered an obsolete root definition (..) in the input file. The NSDBLOAD program ignores this error.

System Action: TCPIP continues.

User or Operator Response: Change the root definition from ‘..’ to ‘.’.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: readmaster

EZB3513W Invalid wild card definition

Explanation: The NSDBLOAD program encountered an incorrect wildcard definition in the input file.

System Action: TCPIP continues.

User or Operator Response: Correct the input file. See RFC 1035 for the correct wild card entry. See Appendix B, “Related protocol specifications (RFCs),” on page 271 for information about accessing RFCs.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: readmaster

EZB3514W Mailbox in SOA data field is specified incorrectly.

Explanation: The NSDBLOAD program encountered an incorrectly specified RNAME field within an SOA RR RDATA field while reading the master data file.

System Action: TCPIP continues.

User or Operator Response: See RFC 1034 for the proper specification of the RNAME field. See Appendix B, “Related protocol specifications (RFCs),” on page 271 for information about accessing RFCs.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: readmaster

EZB3515I Drop index *sql table*

Explanation: NSDBLOAD will preform an SQL drop index command against the table displayed in the message.

System Action: NSDBLOAD performs the SQL drop index command and continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: dropindex

EZB3516I *SQLcommand*

Explanation: The name server will perform a create index SQL command against the SQL table displayed in the message.

System Action: The NSDBLOAD performs the create index command and continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: createndex

EZB3517I * WARNING: OWNER of SQL table is not NAMESEV. ****

EZB3518I * Ensure view exists on *owner.nstable*. ****

Explanation: The current name server SQL table is not owned by NAMESRV.

System Action: NSDBLOAD continues.

User or Operator Response: Make sure that the owner specified in the message has view authority for the table.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: whichtable

EZB3519I Select *tcurrent*

Explanation: NSDBLOAD is selecting a table to receive SQL data.

System Action: NSDBLOAD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: whichtable

EZB3520I PREPARE SQL error: *sqlcode*

Explanation: NSDBLOAD did a PREPARE of the SQL DB2 table, and SQL returned the code that is displayed in the message.

System Action: NSDBLOAD continues.

User or Operator Response: Use the *sqlcode* that is displayed in this message and *DB2 Messages and Codes* to determine the cause of the error and respond as indicated.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: whichtable

EZB3521I Declare SQL error: *sqlcode*

Explanation: NSDBLOAD did a DECLARE of the SQL DB2 table and received the return code that is displayed in the message.

System Action: NSDBLOAD continues.

User or Operator Response: Use the *sqlcode* that is displayed in this message and *DB2 Messages and Codes* to determine the cause of the error and respond as indicated.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: whichtable

EZB3522I Open SQL error: *sqlcode*

Explanation: NSDBLOAD did an OPEN of the SQL DB2 table and received the return code that is displayed in the message.

System Action: NSDBLOAD continues.

User or Operator Response: Use the *sqlcode* that is displayed in this message and *SGL/DS Messages and Codes* to determine the cause of the error and respond as indicated.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: whichtable

EZB3523I Table *ttable* not defined in *nstable*.

EZB3524I using default table *sqltable*

Explanation: NSDBLOAD performed an SQL FETCH command. The table displayed in the message was not defined in the *nstable*.

System Action: NSDBLOAD uses the default table and continues.

User or Operator Response: Verify that the table specified in the NSDBLOAD command was valid and it has been defined in the DNSTABLE data set.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: whichtable

EZB3525I Using sql table *table*

Explanation: The table displayed is being used by NSDBLOAD.

System Action: NSDBLOAD continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: whichtable

EZB3534I

EZB3535I NSDBLOAD reads a master dataset, as defined in RFC1034,

EZB3536I generates resource records, and inserts them into an SQL

EZB3537I table. Multiple name servers can execute simultaneously

EZB3538I on the same system assuming they are using separate ports

EZB3539I or are executing on different TCPIP service machines.

EZB3540I When inserting data into an SQL table owned by NAMESRV the

EZB3541I above SQL names are permitted. When inserting into an SQL

EZB3542I table NOT owned by namesrv, the SQL tablename must be fully

EZB3543I qualified. If a view is defined on nstable, NSDBLOAD will

EZB3544I determine from the status field which table is current (0|1)

EZB3545I Format: nsdbload [db2-subsystem sqltable input-dataset

EZB3546I output-dataset]

EZB3547I where: db2name – SQL subsystem name

EZB3548I sqltable – the table owner defaults to NAMESRV.

EZB3549I Any of the following forms are permitted:

EZB3550I cache, namesrv.cache, or namesrv.cache0.

EZB3551I input-dataset – master data dataset

EZB3552I output-dataset – resource record dataset

Explanation: NSDBLOAD displays this help text if you specified a question mark as the only command line parameter.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: makelower

EZB3560I *time:* Do you want to create resource records from a master file (y/n)?

Explanation: The NSDBLOAD command is prompting you to determine if the resource records should be created from a master file or the program should end now.

System Action: TCPIP continues.

User or Operator Response: Enter Y or N.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: main

EZB3561I *time:* Do you want to delete all resource records from db(y/n)?

Explanation: The NSDBLOAD command is prompting you to determine if all resource records should be deleted from the database.

System Action: TCPIP continues.

User or Operator Response: Enter Y or N.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: main

EZB3562I *time:* Do you want to insert the RR in the sqltable now(y/n)?

Explanation: The NSDBLOAD command is prompting you to determine if the resource records defined in the input parameters should be inserted into the SQL tables now.

System Action: TCPIP continues.

User or Operator Response: Enter Y or N.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: main

EZB3563I *time:* Do you want translate all data to lower case?(y/n)

Explanation: The NSDBLOAD command is prompting you to determine if all input data should be converted to lowercase.

System Action: TCPIP continues.

User or Operator Response: Enter Y or N.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: insertsql

EZB3564I *time:* Do you want translate all data to upper case?(y/n)

Explanation: NSDBLOAD is prompting you to determine if all input data should be translated to uppercase.

System Action: TCPIP continues.

User or Operator Response: Enter Y or N.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: insertsql

EZB3565I *time: Drop index table*

Explanation: NSDBLOAD is issuing an SQL DROP INDEX command for this table.

System Action: TCPIP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name: dropindex

EZB3566I **RDATA Error:** *resource record*

Explanation: The resource record displayed is missing the RDATA field.

System Action: NLDBLOAD increments an error counter and continues.

User or Operator Response: Correct the resource record of the name server SQL table.

System Programmer Response: None.

Source Data Set: NSDBLOAD

Procedure Name:

EZB3825T *name/udp: unknown service*

Explanation: A UDP port number for the service *name* was not assigned in the tcpip.v3r1.ETC.SERVICES data set.

System Action: NCPROUTE exits.

User or Operator Response: None.

System Programmer Response: Verify that the tcpip.v3r1.ETC.SERVICES data set has two entries in the form:
ncproute

port/udp
router *port/udp*

The entries must start in column 1 and be in lowercase. Verify that the NCPROUTE service port number is the port being used by the NCP clients. The port value defined in the UDPPORT= keyword on the IPOWNER statement in the NCP generation definition must match the NCPROUTE service port. The default is UDP port 580.

The reserved router service port number is 520 and is required for the NCPROUTE transport of RIP packets to NCP clients, which are responsible for broadcasting the packets to other RIP routers. The router service port number cannot be overridden due to NCP restriction.

Also, verify that port 580 has been reserved for NCPROUTE under the PORT statement in the tcpip.v3r1.PROFILE.TCPIP data set.

Source Data Set: NRMAIN

Procedure Name: main

EZB3826I **Port** *port* **assigned to** *name*

Explanation: NCPROUTE will listen for traffic from NCP clients on the specified port *port* assigned to service *name*.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: If communications cannot be established with a client, compare NCPROUTE's port number with the value specified in the client's NCP generation definition. The port value defined in the UDPPORT= keyword on the IPOWNER statement in the NCP generation definition must match.

Source Data Set: NRMAIN

Procedure Name: main

EZB3827T Terminating since clients require the socket

Explanation: NCPROUTE attempted to open a socket on a well known or user-defined port but the open was not successful, or the socket could not be bound to an Internet address and port number. Clients will not be able to communicate with NCPROUTE because a socket is not available.

System Action: NCPROUTE ends abnormally.

User or Operator Response: None.

System Programmer Response: Examine previous messages to determine the nature of the error as indicated by a detailed tcperror() library message. Correct the problem as indicated by error. Refer to the *z/OS C/C++ Run-Time Library Reference* for more information about socket() function errors.

Source Data Set: NRMAIN

Procedure Name: Main

EZB3828T Usage: NCPROUTE parameters

Explanation: Incorrect parameters were passed to NCPROUTE.

System Action: NCPROUTE ends abnormally.

User or Operator Response: None.

System Programmer Response: Verify that the parameters are correct. The parameters are case-sensitive and must be separated by spaces. See *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: NRMAIN

Procedure Name: main

EZB3829I Waiting for incoming packets

Explanation: NCPROUTE is waiting for datagrams from NCP clients. Each time that NCPROUTE finishes processing an event, such as an incoming datagram or a timer that expires, NCPROUTE issues this message and waits for the next event. These messages should occur at least once every 30 seconds, but will increase in frequency as the server performs more work.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Check to see if the assigned port number for NCPROUTE matches the NCP clients' generation definitions.

If six or more of these messages occur consecutively, NCPROUTE is not receiving any datagrams from NCP clients. Verify that a client has an established session with NCPROUTE, and if one exists, examine the status of the client's interfaces.

Source Data Set: NRMAIN

Procedure Name: main

EZB3830E The main select was interrupted:

Explanation: An error occurred while NCPROUTE was waiting for an event to occur. A more detailed tcperror() library message follows.

System Action: NCPROUTE continues, unless an IUCV error occurred.

User or Operator Response: None.

System Programmer Response: Verify that TCPIP is running.

EZB3831I • EZB3834I

Source Data Set: NRMAIN

Procedure Name: Main

EZB3831I Send delayed dynamic update

Explanation: A routing update, which had been delayed to prevent packet storms, has been transmitted. This occurs 2–5 seconds after a dynamic update has been issued.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRMAIN

Procedure Name: Main

EZB3832E While receiving a packet from a client:

Explanation: An error occurred while attempting to receive a packet from a client. A more detailed tcperror() library message follows.

System Action: NCPROUTE continues, and the incoming packet is discarded. If an IUCV error occurs, NCPROUTE will end.

User or Operator Response: None.

System Programmer Response: See the next generated error message and correct the error.

Source Data Set: NRMAIN

Procedure Name: process

EZB3833E While receiving a packet from the SNMP agent: *agent*

Explanation: An error occurred while attempting to read a packet from the SNMP agent. A more detailed tcperror() library message follows. If the connection with the Agent has been reset, the SNMP agent will be terminated and incoming SNMP requests will be ignored.

System Action: NCPROUTE continues. The SNMP packet is discarded. If an IUCV error occurs, NCPROUTE will end.

User or Operator Response: None.

System Programmer Response: If the connection with the SNMP agent has been reset, restart the SNMP daemon (OSNMPD).

Source Data Set: NRMAIN

Procedure Name: read_dpi

EZB3834I *****

Explanation: Two of these banners enclose a message that may need attention. when viewing the output. The severity of the enclosed message is indicated.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Read the enclosed message and, if necessary, resolve the indicated situation.

Source Data Set: *various*.

Procedure Name: *various*.

EZB3835T Invalid parameter: *parameter*

Explanation: An incorrect parameter was passed from the tcpip.v3r1.SEZAINST(NCPROUT) start proc JCL. The parameter could be passed from the command line parameters or from the default parameter list in the start proc JCL.

System Action: NCPROUTE exits.

User or Operator Response: None.

System Programmer Response: Correct the parameter from the command line parameters or in the default parameter list of the start proc JCL.

Source Data Set: NRMAIN

Procedure Name: main

EZB3836W The SNMP agent has terminated

Explanation: The socket used to communicate with the SNMP agent has been reset. This implies that the SNMP agent has ended. Without this socket, SNMP requests are not processed.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: If the SNMP agent (OSNMPD) is not running, restart the agent. NCPROUTE will attempt to reestablish communications with the agent after each new SNMP request is received, so no further action is required. If the agent is running, contact the IBM Software Support Center.

Source Data Set: NRMAIN

Procedure Name: read_dpi

EZB3837W SNMP requests will be ignored until it is restarted.

Explanation: Because communication with the SNMP agent is not possible, incoming SNMP requests are ignored until a connection has been established.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Start the SNMP agent (OSNMPD). NCPROUTE will attempt to establish communication with the agent for each SNMP packet received until a connection is established.

Source Data Set: NRMAIN

Procedure Name: read_dpi

EZB3838W An SNMP DPI packet arrived from a non-internet machine

Explanation: An SNMP DPI packet was received from a non-Internet network. SNMP DPI traffic is only accepted from machines running TCPIP. The incoming packet is discarded.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Locate the machine generating the SNMP DPI packet and have it stopped. Ignore these messages.

Source Data Set: NRMAIN

Procedure Name: read_dpi

EZB3839T A socket could not be created:

Explanation: NCPROUTE could not open a new socket. With the socket unavailable, the NCP clients will not be able to establish communications with NCPROUTE. A more detailed tcperror() library message follows.

System Action: NCPROUTE ends abnormally.

User or Operator Response: None.

System Programmer Response: Verify that TCPIP is active, that another program is not using NCPROUTE's port, that the well-known port has been reserved in the PORT statement of *hlq.PROFILE.TCPIP* data set, and that the user-defined port has been specified in *hlq.ETC.SERVICES* data set. Correct the problem as indicated by the error in the detailed tcperror() library message. Refer to the *z/OS C/C++ Run-Time Library Reference* for more information about socket() function errors.

Source Data Set: NRMAIN

Procedure Name: getsocket

EZB3840T Broadcasting cannot be enabled on the socket:

Explanation: NCPROUTE cannot enable the socket for broadcasting. NCPROUTE must be able to broadcast over interfaces which support broadcasting in order to communicate with the NCP clients. A more detailed tcperror() library message follows.

System Action: NCPROUTE ends abnormally.

User or Operator Response: None.

System Programmer Response: Contact your IBM Software Support Center.

Source Data Set: NRMAIN

Procedure Name: getsocket

EZB3841T The socket bind failed

Explanation: NCPROUTE was unable to associate an Internet address and port number to the newly created socket. Another application could be using the port. A more detailed tcperror() library message follows.

System Action: NCPROUTE ends abnormally.

User or Operator Response: None.

System Programmer Response: Use the NETSTAT ALLCONN command to verify that there is no other application using the NCPROUTE port number. Look for a port in the "Local Socket" column which matches NCPROUTE's port in the *hlq.ETC.SERVICES* data set. You should reserve this port for NCPROUTE's exclusive use by adding an entry to the PORT statement in the *hlq.PROFILE.TCPIP* data set.

Source Data Set: NRMAIN

Procedure Name: Getsocket

EZB3842W Hello from existing client *client*

Explanation: An NCP client with a current session has entered a reset state and has started to send Hello packets in an attempt to establish a session with NCPROUTE. This client might have been shut down and restarted. The current session ends and a new one is started.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRPDUS

Procedure Name: Pdu_in

EZB3843W Status from a session-less client *client*

Explanation: A client has transmitted an interface status change PDU without first establishing a session with NCPROUTE. This is a protocol violation. The PDU is discarded. Also, the client is added to a “Client Protocol Violation List” and no further error messages will be issued for this client until it successfully establishes a session. At that point, NCPROUTE will accept the client, and the client’s name will be removed from the list.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Restart the client NCP. If the problem persists, contact the IBM Software Support Center.

Source Data Set: NRPDUS

Procedure Name: pdu_in

EZB3844W This datagram is being ignored

Explanation: A datagram was received and some outstanding error prevents NCPROUTE from operating on the packet. The most likely cause is that a protocol error occurred between the client and NCPROUTE. The datagram is discarded.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Identify the cause of the error from a previous error message and correct the problem. The client must be reset so that a new session is established.

Source Data Set: NRPDUS

Procedure Name: pdu_in

EZB3845W Transport from session-less client *client*

Explanation: A client NCP has transmitted a Transport PDU without first establishing a session with NCPROUTE. This is a protocol violation. The PDU is discarded. Also, the client is added to a “Client Protocol Violation List” and no further error messages will be issued for this client until it successfully establishes a session. At that point, NCPROUTE will accept the client and the client’s name will be removed from the list.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Restart the client NCP. If the problem persists, contact the IBM Software Support Center.

Source Data Set: NRPDUS

Procedure Name: pdu_in

EZB3846W InactList from session-less client *client*

Explanation: A client NCP has transmitted an inactive interface PDU without first establishing a session with NCPROUTE. This is a protocol violation. The PDU is discarded. Also, the client is added to a “Client Protocol Violation List” and no further error messages will be issued for this client until it successfully establishes a session. At that point, NCPROUTE will accept the client and the client’s name will be removed from the list.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Restart the client NCP. If the problem persists, contact the IBM Software Support Center.

Source Data Set: NRPDUS

Procedure Name: pdu_in

EZB3847W Protocol Violation: client *client* sent

Explanation: A foreign machine sent a packet to the NCPRROUTE port that does not contain a valid type field in the packet header. This is probably not an NCP, but another machine on the network. The PDU is discarded. Also, the client is added to a “Client Protocol Violation List” and no further error messages will be issued for this client until it successfully establishes a session. At that point, NCPRROUTE will accept the client and the client’s name will be removed from the list.

System Action: NCPRROUTE continues.

User or Operator Response: None.

System Programmer Response: Identify the source of the packet from the displayed Internet address and correct the problem. If the machine in question is an NCP, verify that it is loaded with the correct software and reset it. If the machine is an NCP and resetting it does not correct the problem, contact the IBM Software Support Center.

Source Data Set: NRPDUS

Procedure Name: pdu_in

EZB3848W An excessive number of clients (#) have been issued protocol violations. Further warning messages will be suppressed.

Explanation: A large number of clients have committed protocol violations and have not subsequently established sessions with NCPRROUTE. No further “Warning” messages are issued. NCPRROUTE assumes that the problems have been noticed by this point and that steps are being taken to correct them.

System Action: NCPRROUTE continues.

User or Operator Response: None.

System Programmer Response: Verify that the most recent versions of NCPRROUTE and the NCP drivers are installed. Verify that previous protocol errors on individual client NCPs were investigated and corrective actions were taken. Contact the IBM Software Support Center if the protocol violations continue.

Source Data Set: NRPDUS

Procedure Name: bad_client

EZB3850E Status change for unknown interface *interface*

Explanation: NCPRROUTE received a status change request PDU from the NCP client for an interface that was unknown. Either NCPRROUTE did not completely read in the NCP client’s Routing Information Table (RIT), or the RIT was not built correctly during NCP generation. Another possible cause is that the NCP client dynamically added the interface to its tables. Dynamically-added interfaces are not currently supported by NCPRROUTE. The Status PDU is discarded.

System Action: NCPRROUTE continues.

User or Operator Response: None.

System Programmer Response: Check the interface configuration messages created in the console log at install time to verify that the interfaces defined in the NCP client’s generation match the interfaces processed by NCPRROUTE. If a mismatch is found, verify that the RIT is built correctly. If the problem cannot be corrected, contact the IBM Software Support Center.

Source Data Set: NRPDUS

Procedure Name: recv_status

EZB3851I Accepting bad client *client*

Explanation: A client that committed a protocol violation earlier has successfully established a session and any new protocol violations are reported.

System Action: The client is removed the “Client Protocol Violation List”. With the client removed from this list, NCPRROUTE resumes reporting any new protocol violation errors.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRPDUS

Procedure Name: forgive

EZB3855I **NCP_Add out to client Route to address1 via interface interface to address2 Metric: metric, Type type, Subnetmask mask**

Explanation: An "Add" PDU is being sent to a client, which causes a route to be added to its IP route tables.

address1 is the destination Internet address of the route.

The *interface* is the name given to the interface during NCP generation.

address2 is the Internet address of an intermediate router or zero if the destination is directly connected.

The *metric* is the relative cost of using this route as opposed to another route.

The *type* is either Host, Subnet, or Network and indicates the route type being added.

System Action: NCPRROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRPDUS

Procedure Name: nr_do_add

EZB3856I **Pending delete for interface interface**

Explanation: NCPRROUTE received a status change request from the NCP client to delete the specified interface from its interface tables. The NCP client had deleted the interface from its configuration as a result of dynamic reconfiguration. NCPRROUTE puts the interface in pending delete state so that routing outages can be reported to other interfaces and the incorrect addition of routes in routing responses can be prevented. NCPRROUTE will continue to remove the specified interface from its tables until either the routes are timed out or a status change request to add a new interface with the same Internet address is received.

System Action: NCPRROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRPDUS

Procedure Name: recv_status

EZB3857I **Adding new interface interface**

Explanation: NCPRROUTE received a status change request from the NCP client to add a new interface to its interface tables. The NCP client had added the interface to its configuration as a result of dynamic reconfiguration. NCPRROUTE will manage routes for the new interface.

System Action: NCPRROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRPDUS

Procedure Name: recv_status

EZB3858E Unknown status change request

Explanation: NCPROUTE received an invalid status change request from the NCP client. The Status PDU is discarded.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact your IBM Software Support Center.

Source Data Set: NRPDUS

Procedure Name: recv_status

EZB3859I Deleting interface *interface*

Explanation: NCPROUTE is deleting the specified interface from its interface tables. The interface was previously in pending delete state so that routing outages could be reported to other interfaces and the incorrect addition of routes in routing responses could be prevented. The deletion occurs when either the routes attached to the interface have timed out or a status change request to add a new interface with the same Internet address is received.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRPDUS

Procedure Name: recv_status

EZB3860E The transmission of an 'Add' to client *client* failed

Explanation: TCPIP detected that the "Add" PDU could not be delivered successfully. The add request, for adding a route to the NCP client's routing table, is not performed. A more detailed tcperror() library message follows.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Examine the error message that follows and proceed according to the recommendations.

Source Data Set: NRPDUS

Procedure Name: nr_do_add

EZB3862I NCP_Add out to *client* Route to *address1* via interface *interface* to *address2* Metric: *metric*, Type *type*, Subnetmask *mask*

Explanation: A "Delete" PDU is being sent to the named client. The *address* is the destination of the route to be deleted, and the *type* is either Host, Subnet or Network and is used by the client to locate the correct route table from which to delete the route.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRPDUS

Procedure Name: nr_do_delete

EZB3863S RIP2 authentication key exceeds maximum allowed or contains unsupported characters

Explanation: The RIP Version 2 authentication key, specified on the RIP2_AUTHENTICATION_KEY entry in the NCPROUTE profile data set, or on the options statement entry in the NCP client's gateways data set for an interface, is invalid. The authentication key may have contained unsupported characters or have exceeded the maximum 16 characters allowed. The authentication key is ignored and no authentication check is performed.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the NCPROUTE profile or the NCP client's GATEWAYS data set.

Source Data Set: NRMAIN, NRSTART

Procedure Name: read_profile, ParseOptions

EZB3864E The transmission of a 'Delete' to client *client* failed

Explanation: TCPIP detected that the "Delete" PDU could not be delivered to an NCP client. The delete request, for deleting a route from the NCP client's routing table, is not performed. A more detailed tcperror() library message follows.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Examine the error message that follows and proceed according to the recommendations.

Source Data Set: NRPDUS

Procedure Name: nr_do_delete

EZB3865T Out of memory during transport

Explanation: NCPROUTE has used all available storage while attempting to send a "Transport" PDU to a client. In this condition, NCPROUTE can no longer communicate with its clients.

System Action: NCPROUTE exits.

User or Operator Response: None.

System Programmer Response: Increase NCPROUTE's region size and restart. Take into consideration that storage requirements are based on the number of clients being served and the number of routes being managed. If the problem still cannot be resolved, contact the IBM Software Support Center.

Source Data Set: NRPDUS

Procedure Name: nr_do_transport

EZB3866E The transmission of a 'Transport' to client *client* failed

Explanation: TCPIP detected that it would be unable to deliver the "Transport" PDU to a client. The "Transport" PDU is discarded. A more detailed tcperror() library message follows.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Examine the error message that follows and proceed according to the recommendations.

Source Data Set: NRPDUS

Procedure Name: nr_do_transport

EZB3867I Acknowledge to *client*: Hello Received

Explanation: A client has attempted to establish a session with the server and is being updated on the status of the request. The "Hello" PDU was received and is sent immediately to quiet the client.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRPDUS

Procedure Name: recv_hello

EZB3868I Acknowledge to *client*: RIT Loaded OK

Explanation: A client has attempted to establish a session with the server, and is being updated on the status of the request. The Routing Information Table (RIT) was loaded and appears usable.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRPDUS

Procedure Name: recv_hello

EZB3869E Acknowledge to *client*: RIT Load Failed

Explanation: A client has attempted to establish a session with the server and is being updated on the status of the request. An abend occurred during the load of the Routing Information Table (RIT). The session with the server is not established. A previous message explains the cause of the abend.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Examine the abnormal end code in a previous message and correct the error. If the problem cannot be corrected, contact the IBM Software Support Center.

Source Data Set: NRPDUS

Procedure Name: recv_hello

EZB3870E Acknowledge to *client*: RIT ID Bad

Explanation: A client has attempted to establish a session with the server and is being updated on the status of the request. An error was detected in the correlation string of the Routing Information Table (RIT). The string does not match the one in the received "Hello" PDU. Most likely the NCP client has been reconfigured after a new NCP load and, as a result, the RIT ID was updated. The session with the server is not established.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Verify that the correct RIT was installed and that another data set was not inadvertently loaded earlier. The RIT should be referenced in the operating system's link list or in the DD:STEPLIB statement of the NCPROUTE catalogued procedure. If the problem cannot be corrected, contact the IBM Software Support Center.

Source Data Set: NRPDUS

Procedure Name: recv_hello

EZB3871E Acknowledge to *client*: RIT Bad

Explanation: A client has attempted to establish a session with the server and is being updated on the status of the request. An error was detected in the Routing Information Table (RIT) data. The session with the server is not established.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Verify that the correct RIT was installed and that another data set was not inadvertently loaded earlier. The RIT should be referenced in the operating system's link list or in the DD:STEPLIB statement of the NCPROUTE catalogued procedure. If the problem cannot be corrected, contact the IBM Software Support Center.

Source Data Set: NRPDUS

Procedure Name: recv_hello

EZB3872E Acknowledge to *client*: RIT Not Found

Explanation: A client has attempted to establish a session with the server and is being updated on the status of the request. The Routing Information Table (RIT) could not be found matching the NCP name in the received "Hello" PDU. The session with the server is not established.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Verify that the RIT built during NCP generation has been installed into a partitioned data set and referenced by the DD:STEPLIB statement of NCPROUTE catalogued procedure. If the problem cannot be corrected, contact the IBM Software Support Center.

Source Data Set: NRPDUS

Procedure Name: recv_hello

EZB3873E Acknowledge to *client*: Unsupported Ack Type

Explanation: A client has attempted to establish a session with the server, and is being updated on the status of the request. An unknown Acknowledgement type was received. The session with the server is not established.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: NRPDUS

Procedure Name: recv_hello

EZB3875W Variable subnetting not supported by client *client*

Explanation: Variable subnetting is not supported by the *client* and the RIP supply/receive control setting may have been set in the NCPROUTE configuration such that RIP Version 2 packets are to be sent or received over interface(s). NCPROUTE will override the control settings to RIP1 for compatibility with the NCP client configuration.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Verify the NCP software level for variable subnetting support in the NCP client. If RIP Version 2 packets are not to be used, correct the NCPROUTE profile data set or the NCP client's gateways data set to use RIP Version 1 packets.

Source Data Set: various

Procedure Name: various

EZB3876I Hello from new client *client*

Explanation: An NCP client is attempting to establish a session with NCPROUTE. This is done by sending a "Hello" PDU containing the NCP name and correlation string used to verify that the correct Routing Information Table (RIT) is loaded.

Note: The NCP client will issue many "Hello" PDUs up to the maximum specified in the NCP gen until it successfully establishes a session with the server. Once the maximum is reached, the NCP client will repeat the cycle again after a 9-minute delay timer has expired. The delay timer is used to prevent NCP alert flooding. The NCP client's issuance of "Hello" PDUs can happen even after the server has established a session with the NCP client. For example, the NCP client could have been restarted. An attempt will be made to establish a new session.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRPDUS

Procedure Name: recv_hello

EZB3877I RIT dataset name: *ritdsname*

Explanation: The specified Routing Information Table (RIT) data set name, as supplied in the Hello PDU, is loaded and processed.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRPDUS

Procedure Name: recv_hello

EZB3878I RIT ID: *table_id*

Explanation: The specified ID string is used as a correlation string to verify that the correct Routing Information Table (RIT) has been loaded.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None, however, if this ID appears unusual, determine if the correlation string was overridden during NCP load.

Source Data Set: NRPDUS

Procedure Name: recv_hello

EZB3879E timestamp

Explanation: An error occurred and a full time stamp is written showing the date and time when the error occurred. While each message is timestamped, date information is omitted, and this message is issued to help those running for long periods of time.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRPDUS

Procedure Name: Recv_hello

EZB3880E Error opening RIT *client1* for client *client2*

Explanation: The Routing Information Table (RIT), which was created during NCP generation, could not be opened. *client1* is the name of the member on which the load was attempted, *client2* is the Internet address of the client that is attempting to establish a session with NCPROUTE. NCPROUTE does not establish the session.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Examine the abend code and reason code in the following EZB3881E message for the specific cause of the error. In the event of an 804 abend, verify that the RIT exists and is accessible to NCPROUTE.

Source Data Set: NRPDUS

Procedure Name: recv_hello

EZB3881E Abend code: *abend*, Reason: *reason*

Explanation: An abend occurred during an attempt to load the Routing Information Table (RIT) into memory with the LOAD macro. The abend and reason code returned are displayed. The session with the server is not established.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: See *z/OS MVS System Codes* for an explanation of the abend and reason codes, and further instructions.

Source Data Set: NRPDUS

Procedure Name: recv_hello

EZB3882E A session will not be established

Explanation: An error occurred during an attempt to establish a session with a client, and the session cannot be established at this time.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the errors described in previous messages.

Source Data Set: NRPDUS

Procedure Name: recv_hello

EZB3883E Client: *client*

Explanation: An error occurred during a transaction with the client. The client is the Internet address of the client. This message is used to identify the failing client, and is followed by additional messages.

System Action: NCPROUTE issues additional messages.

User or Operator Response: None.

System Programmer Response: Examine the messages that follow for a description of the error.

Source Data Set: NRPDUS

Procedure Name: recv_hello

EZB3884E The RIT ID field does not match the one in the Hello. RIT *idfield*, Hello: *correlfield*

Explanation: The ID field in the Routing Information Table (RIT) does not match the correlation field passed from the client NCP in the hello datagram. The mismatched id and correlation fields are displayed. This indicates that the client is using a generation that is different from the one that built the RIT. A session is not established.

EZB3885I • EZB3888E

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Generally, the RIT used by NCPROUTE should be the RIT that was generated during the generation of the current NCP load. In this case, an older RIT with the same name is probably being loaded. Locate and remove the old RIT. In some cases, multiple machines might want to own a single NCP and use the same NCP generation, but would require a unique RIT to run properly. See the NCP documentation for instructions about how to set the correlation string in your generations and during NCP load.

Source Data Set: NRPDUS

Procedure Name: recv_hello

EZB3885I **Input parameters:** *parms*

Explanation: This message lists the string of input parameters passed to NCPROUTE from tcpip.v3r1.SEZAINST(NCPROUT) start proc JCL. The parameters could either be passed from the command line parameters or from the defined parameter list in the start proc JCL. If no parameters are specified, '*None*' will appear in the string.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRMAIN

Procedure Name: main

EZB3886E **An error occurred while loading the RIT for client *client*.**

Explanation: An internal error occurred. As a result of the error, the information in the Routing Information Table (RIT) is not available but an error condition was not indicated on return from the load. The new session ends.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: NRPDUS

Procedure Name: recv_hello

EZB3887E **An internal error occurred, terminating session.**

Explanation: Because a session was established, but is unusable, the new session ends and this message is displayed.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: NRPDUS

Procedure Name: recv_hello

EZB3888E **An error occurred during interface initialization.**

Explanation: An error occurred during processing of the Routing Information Table (RIT) interface list. Previous messages should describe the nature of the error. The new session ends.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Examine previous messages and follow recommendations for the error that occurred.

Source Data Set: NRPDUS

Procedure Name: recv_hello

EZB3889E The session will be terminated

Explanation: An error occurred that makes one of NCPROUTE's sessions unusable. The session ends, and the client NCP goes into a reset state within three minutes. At that point, the NCP client attempts to reestablish a session.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Examine previous messages to determine what error occurred and correct the problem. If the error takes a while to correct, renaming the Routing Information Table (RIT) causes future attempts at establishing a session to be unsuccessful.

Source Data Set: NRPDUS

Procedure Name: recv_hello

EZB3890I Recv: status from *client*

Explanation: A client's interface has changed state and the client has reported this by an interface status change PDU.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRPDUS

Procedure Name: recv_status

EZB3891I Interface *ip_addr* is now *status* - *interface_name*

Explanation: The client's interface name having an Internet address of *ip_addr* has changed state and is currently set to either "up" or "down".

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: If the status change is not expected, examine the client NCP to determine the reason for the change. See the NCP documentation for more information.

Source Data Set: NRPDUS

Procedure Name: recv_status

EZB3894I Transport from *client*: *count* bytes of RIP data.

Explanation: Client *client* has received a packet addressed to UDP port 520 and has forwarded this packet to NCPROUTE in a Transport PDU.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRPDUS

Procedure Name: recv_transport

EZB3895I Transport from *client*: *count* bytes of SNMP data.

Explanation: The *client* has received a packet addressed to UDP port 161 and has forwarded this to NCPROUTE in a transport PDU.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRPDUS

Procedure Name: recv_transport

EZB3896E Transport for unsupported port *port number* bytes received and discarded

Explanation: The client NCP has sent NCPROUTE a Transport PDU containing a packet that was addressed to a port that NCPROUTE does not support. The transport is discarded.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: NRPDUS

Procedure Name: recv_transport

EZB3897I NCPROUTE Server started

Explanation: NCPROUTE has completed initialization and is waiting for incoming packets from NCP client(s) to initiate sessions.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRMAIN

Procedure Name: main

EZB3898I Recv: Inactive Interface List from *client number* interface(s) found:

Explanation: After a session is established, the client sends NCPROUTE a list containing the number of interfaces that are currently down. All other interfaces (listed in the Routing Information Table (RIT)) are assumed to be active. Following this message, the IP addresses and names of the inactive interfaces are displayed.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRPDUS

Procedure Name: recv_inactlist

EZB3899I *ipaddr* - interface

Explanation: The IP address and name of the inactive interface are displayed.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRPDUS

Procedure Name: recv_inactlist

EZB3900W Unable to open NCPROUTE profile *profile*

Explanation: A profile data set was not specified or could not be opened. This message indicates which data set the open was attempted on.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Verify that the profile data set is defined in //DD:NCPRPROF of the NCPROUTE start proc JCL and is accessible by NCPROUTE. If the data set is sequential, ensure that the FREE=CLOSE parameter is specified. Do not specify this parameter if the data set is partitioned.

Source Data Set: NRMAIN

Procedure Name: read_profile

EZB3901W Ignoring route *destination*, supernetting not supported

Explanation: The route to *destination* received in the RIP packet happens to be a supernet type of route and is ignored since the supernetting feature is not supported by the NCP client. A supernet route is one where its subnet mask is less than the route's network class mask.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Verify the NCP software level for supernetting support in the NCP client.

Source Data Set: NRINPUT

Procedure Name: rip_input

EZB3902W Address is experimental, or has non-zero port

Explanation: An incorrect Internet address was encountered, which is either in an experimental address class, or is using an unusual port number (Routing Information Protocol (RIP) packets only). The address is being validated to make sure that a network user does not pretend to be a router to change the routing table of nearby routers (such as the client NCP). The address is not considered as a valid destination address for a route, and the route entry is discarded.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Identify the machine or user that generated the packet in question and correct the problem.

Source Data Set: NRAF

Procedure Name: inet_checkhost

EZB3903W Invalid internet address

Explanation: An Internet address in an incoming route is determined to not be a member of any defined Internet address class. The route is discarded.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the router that originated this packet.

Source Data Set: NRAF

Procedure Name: inet_checkhost

EZB3904W A must-be-zero field is non-zero

Explanation: An Internet address in an incoming route contains a nonzero value in a field that must be 0. The incoming route is discarded.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the router that originated this packet.

Source Data Set: NRAF

Procedure Name: inet_checkhost

EZB3905S function *function* client *client* unknown

Explanation: The function was unable to obtain a list of interfaces for the client with Internet address *client*. If *client* is an actual client of NCPROUTE, the interface list has been lost, otherwise the function is using an incorrect address for its client. NCPROUTE indicates that the requested interface does not exist.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: NRIF

Procedure Name: ifwithaddr

EZB3906S NULL interface list for client *client*

Explanation: A list of interfaces for client *client* exists, but is marked empty. Dynamic routing requires at least one interface. The requested interface is indicated as nonexistent.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Verify that the client NCP generation had at least one interface that did not have the RIPMGD=NO keyword coded. If none are found, correct the configuration and regenerate the client. Verify that EZB3956I messages are issued which correspond with the Routing Information Protocol (RIP)-managed interfaces in the generation. If none are found, attempt to reload the Routing Information Table (RIT). Contact the IBM Software Support Center if the problem persists.

Source Data Set: NRIF

Procedure Name: ifwithaddr

EZB3908I Modify command is set for all clients

Explanation: NCPROUTE is configured to process MODIFY commands for all clients rather than for a specific client.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRMAIN

Procedure Name: do_modify

EZB3910I **Modify command is set for client** *client*

Explanation: NCPROUTE is configured to process MODIFY commands for a targeted client rather than for all clients.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRMAIN

Procedure Name: do_modify

EZB3911S **Function** *function* **address family out of range. The address is** *address*.

Explanation: One of NCPROUTE's route entries has an unsupported address family. NCPROUTE cannot determine the network based on the address, therefore it cannot determine an interface that serves the logical network. The requested interface is indicated as nonexistent.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Noninternet addresses are not supported in this version of NCPROUTE. Either ignore the message, or have the router that originated this route stop sending noninternet routes to the client NCP. Look back through the output to find the last ADD or CHANGE for this destination to obtain the Internet address of the router.

Source Data Set: NRIF

Procedure Name: ifwithnet

EZB3912I **ifwithnet: compare with** *interface*

Explanation: The route's network address is being compared with one of NCPROUTE's interface entries for a network number match.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRIF

Procedure Name: if_ifwithnet

EZB3913I **ifwithnet: remote interface ignored**

Explanation: One of NCPROUTE's interface entries is for a remote interface, which is ignored during the search for a network number match.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRIF

Procedure Name: if_ifwithnet

EZB3914S **ifwithnet: interface has bad address family**

Explanation: One of NCPROUTE's interface entries has an incorrect address family. NCPROUTE cannot determine the network based on the address. Therefore, it cannot find an interface that serves the logical network.

System Action: NCPROUTE continues.

EZB3915I • EZB3918I

User or Operator Response: None.

System Programmer Response: Either ignore the message or correct the NCP client's interface that contains the incorrect address family. If the interface cannot be corrected, contact your IBM Software Support Center.

Source Data Set: NRIF

Procedure Name: if_ifwithnet

EZB3915I netmatch *ipaddr1* and *ipaddr2*

Explanation: A network number match was found for the route's network address *ipaddr1* with one of NCPROUTE's interface entries having a network address of *ipaddr2*.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRIF

Procedure Name: if_ifwithnet

EZB3916I Blocking route for *destination*

Explanation: From a packet received over a particular interface, the route for *destination* is being blocked from being added to NCPROUTE's routing table. This is the result of a RIP I/O filter option specified in the client's *hlq.ETC.GATEWAYS* data set. See *z/OS Communications Server: IP Configuration Reference* for more information.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRINPUT

Procedure Name: rip_input

EZB3917I NCPROUTE's internal *type* table for client *client*:

Explanation: NCPROUTE's internal IP routing or interface table is displayed for diagnosis.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTABLES, NRIF

Procedure Name: dsp_rrtables, dsp_iftables

EZB3918I Modify parameters processed; see SYSPRINT or SYSERR output for results

Explanation: The modify command with specified parameters has been processed. For results, see the SYSPRINT or SYSERR output.

System Action: NCPRoute continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRMAIN

Procedure Name: do_modify

EZB3921I Tracing debug packets *action timestamp*

Explanation: Debug packets tracing is enabled. The packets are displayed in data format.

System Action: NCPRoute continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTRACE

Procedure Name: modifydebuglevel

EZB3922S inet_makeaddr: no session with *client*

Explanation: The function inet_makeaddr was unable to obtain a list of interfaces for the client with Internet address *client*. If *client* is an actual client of NCPROUTE, the interface list has been lost; otherwise, the function is using an incorrect address for its client. A NULL Internet address is created.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: NRINET

Procedure Name: inet_makeaddr

EZB3923S inet_netof: no session with *client*

Explanation: The function inet_netof was unable to obtain a list of interfaces for the client with Internet address *client*. If *client* is an actual client of NCPROUTE, the interface list has been lost; otherwise, the function is using an incorrect address for its client. Because a subnetmask cannot be located, a network number is returned.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: NRINET

Procedure Name: inet_netof

EZB3924I inet_netof: ignoring REMOTE interface

Explanation: One of NCPROUTE's interface entries is for a remote interface, which is ignored during the calculation for a (sub)network number from the interface's Internet address.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRINET

Procedure Name: inet_netof

EZB3925S Inet_lnaof: no session with *client*

Explanation: The function inet_lnaof was unable to obtain a list of interfaces for the client with Internet address *client*. If *client* is an actual client of NCPROUTE, the interface list has been lost; otherwise, the function is using an incorrect address for its client. Because subnetmask information is not available, the host part of the Internet address is returned.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center. This might contain a subnet number if the interface is subnetted.

Source Data Set: NRINET

Procedure Name: inet_lnaof

EZB3926S inet_rtflags: no session with *client*

Explanation: The function `inet_rtflags` was unable to obtain a list of interfaces for the client with Internet address *client*. If *client* is an actual client of NCPROUTE, the interface list has been lost; otherwise, the function is using an incorrect address for its client. A determination is made to indicate either a network address or a host address assuming no subnetting has been done.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: NRINET

Procedure Name: inet_rtflags

EZB3927E Packet from unsupported address family (*family*), cmd (*command*)

Explanation: A packet was received from a non-Internet network. RIP traffic is only accepted from machines running IP. The incoming packet is discarded.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Locate the machine generating the RIP packet and have it stopped. These messages can also be ignored.

Source Data Set: NRINPUT

Procedure Name: rip_input

EZB3928E RIP version 0 packet received from *ipaddr*

Explanation: A Routing Information Protocol (RIP) Version 0 packet was received from the specified address. This RIP version is obsolete. The incoming packet is discarded.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the router that is sending the Version 0 packets.

Source Data Set: NRINPUT

Procedure Name: rip_input

EZB3929I Request: output routines removed

Explanation: NCPROUTE received a Routing Information Protocol (RIP) request packet for a specific set of routes rather than for a set of all routes. (A request for all routes is indicated by an address family of 0 and infinite metric of 16.) Since NCPROUTE does not support handling requests for a specific set of routes, the output routines in this case have been removed to prevent a response.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRINPUT

Procedure Name: rip_input

EZB3930W Trace command from unknown router *ipaddr*

Explanation: A trace packet was received from a router that is either not directly connected to any of the client's Routing Information Protocol (RIP) managed interfaces, or is directly connected to an interface that is not capable of supporting RIP traffic. RIP requires that an interface be capable of supporting link-level broadcast traffic, be a point-to-point interface (such as NCST sessions), or have an active gateway defined. The incoming packet is discarded.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Determine where the packet originated from and correct the problem. Bridges can allow routers on logically disconnected networks to talk with each other even though the routers have no logical connection to each other's networks.

Source Data Set: NRINPUT

Procedure Name: rip_input

EZB3931W Response from non-router *ipaddr*

Explanation: A Routing Information Protocol (RIP) response packet was received with an incorrect port number. All routers on a network must agree on the port number that is used to exchange routing information, and this port must be restricted so that other applications cannot generate routing updates. Either a router is configured using the wrong port number, or an application issues RIP routing updates. The RIP response is discarded.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Reconfigure the router to use a correct port number or locate the application that is generating the updates and correct the problem.

Source Data Set: NRINPUT

Procedure Name: rip_input

EZB3932W Invalid packet from passive interface *interface*

Explanation: A Routing Information Protocol (RIP) response packet was received from a passive interface. Passive interfaces receive routing updates from the client, but cannot produce routing updates. The packet is discarded.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Locate the router that is producing the bogus packet and correct the problem.

Source Data Set: NRINPUT

Procedure Name: rip_input

EZB3933W Packet from unknown router *ipaddress (reason)*

Explanation: A Routing Information Protocol (RIP) response was received from a router which is not directly connected via a broadcast network, a point-to-point (NCST) network, or an active gateway as defined in a GATEWAYS PDS member. The description of the reason for this warning is one of the following:

Reason	Explanation
--------	-------------

Interface in strange state	
-----------------------------------	--

	The network does not support broadcast or point-to-point transmission.
--	--

Iflookup failed	
------------------------	--

	Not directly connected.
--	-------------------------

Link is PASSIVE!	
-------------------------	--

	Cannot update.
--	----------------

EZB3937W • EZB3941I

The packet is discarded.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Locate the router that produced the packet and correct the problem.

Source Data Set: NRINPUT

Procedure Name: rip_input

EZB3937W New route is in unsupported address family. client = *client*, route from *ipaddr*, new family = *family*

Explanation: An incoming router from another router is in an address family that is not supported by NCPROUTE. Currently, only Internet addresses are supported. The client is the Internet address of the client NCP, the *ip addr* is the Internet address of the router that originated the route that is not valid, and family is the address family that is not supported by NCPROUTE. The route that is not valid is discarded.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Locate the router that produced the route and correct the problem.

Source Data Set: NRINPUT

Procedure Name: rip_input

EZB3939E Illegal address *hostaddr* in route from *ipaddr*

Explanation: An Internet address that is not valid was received in an update from router *ipaddr*. A previous message indicates the nature of the problem with the address. The route that is not valid is discarded.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Locate the router that originated the route and correct the problem.

Source Data Set: NRINPUT

Procedure Name: rip_input

EZB3940W Bad metric (*metric*) in route to destination from router *ipaddr*

Explanation: A route was received from *ipaddr* that contained a metric that was not in the range 1 - 16. The route is discarded.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Locate the router ip address and correct the problem.

Source Data Set: NRINPUT

Procedure Name: rip_input

EZB3941I Adjusting large metric (*metric*) to infinity

Explanation: From the routing updates, a route contained a metric exceeding the maximum supported metric of 16. The metric is changed to infinite metric of 16, to indicate that the destination route is network unreachable.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRINPUT

Procedure Name: rip_input

EZB3943I Send dynamic update

Explanation: Changes have occurred and an update has not been sent recently. Enough time has passed since the last update was sent so that it is safe to transmit again without risking an update storm. Also, no dynamic update was pending; otherwise, NCPROUTE would wait until that update occurred before sending the update.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRINPUT

Procedure Name: rip_input

EZB3944I Delay dynamic update

Explanation: Changes have occurred in the network topology, but an update was recently made to adjacent routers. When the last update was made, a random delay time, 2-5 seconds from that point, was determined. This update is scheduled to occur at that time.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRINPUT

Procedure Name: rip_input

EZB3945I Inhibit dynamic update for *seconds usec*

Explanation: A dynamic update has just been sent. Another dynamic update is prevented from occurring for the number of microseconds indicated in the message. A random time is chosen, 2-5 seconds from this time, and if another update is needed later, it will be delayed until this random time has passed.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRINPUT

Procedure Name: rip_input

EZB3946S toall (STUB) -- this shouldn't be used anymore

Explanation: The routine (toall) called by NCPROUTE is no longer supported.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: NROUTPUT

Procedure Name: toall

EZB3947S toall: client *client* unknown

Explanation: Information about the client is not available, but NCPROUTE is attempting to send output over all of the client's interfaces. This is an internal error. No output is sent to the client.

System Action: NCPROUTE continues.

EZB3948I • EZB3951I

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NROUTPUT

Procedure Name: toall_ifs

EZB3948I Interface *interface* not up

Explanation: The specified interface is detected to be inactive. No route will be added unless the interface is (re)activated.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSTART

Procedure Name: addrouteforif

EZB3949I Interface *interface* is passive

Explanation: The *interface* is in a passive state, meaning that RIP traffic is disabled for the interface. Routing updates will not be broadcast to the interface and incoming routing updates are ignored. This may be the result of a RIP I/O filter option specified in the client's *hlq.ETC.GATEWAYS* data set. See *z/OS Communications Server: IP Configuration Reference* for more information.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NROUTPUT

Procedure Name: toall

EZB3950I toall: requested to skip interface *interface*

Explanation: The interface *interface* is skipped because the interface has already received notification of a routing change. If a broadcast of the routing table has not been sent recently and a routing change has occurred, a dynamic routing update will be broadcasted to other interfaces to inform adjacent routers of the routing change.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NROUTPUT

Procedure Name: toall

EZB3951I client *client*: supply *destaddress* -> port *via* *interfacename*

Explanation: NCPROUTE is directing the client to transmit a Routing Information Protocol (RIP) datagram to the destination address. This can be either a broadcast address or a host address. If the port is zero, the Routed port will be used from *hlq.ETC.SERVICES*; otherwise, the datagram will be transmitted to the specified port. NCPROUTE will ask the client to use the specified interface name when sending the datagram.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NROUTPUT

Procedure Name: supply

EZB3952E **Unknown interface name (*interface*) and/or address (*ip_addr*)**

Explanation: In the line entry for the client's *hlq.ETC.GATEWAYS* data set, the options definition contains interface information that is not in NCPROUTE's interface tables. Most likely, the interface name is misspelled or the interface's Internet address is specified incorrectly. Although other options may be processed normally, the invalid option is ignored.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the client's *hlq.ETC.GATEWAYS* data set. See *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: NRSTART

Procedure Name: gateways

EZB3953S **NULL base**

Explanation: NCPROUTE was unable to locate a base hash table for the client. Every active client should have this table, so its absence indicates an internal error. No output occurs.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: NROUTPUT

Procedure Name: supply

EZB3954W **Invalid metric, changing to one**

Explanation: In the line entry for the NCP client's *hlq.ETC.GATEWAYS* data set, the metric has an incorrect value. The metric is changed to 1.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the NCP client's *hlq.ETC.GATEWAYS* data set.

Source Data Set: NRSTART

Procedure Name: gateways

EZB3955S **Function *function*: out of memory**

Explanation: NCPROUTE was unable to allocate memory because no more storage is available in the region. The following describe functional errors:

Function

Error Description

ifinit No storage available to add a client interface entry.

nradd No storage available to add a route table entry. Each route table entry requires 64 bytes.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Increase the region size in the startup procedure.

Source Data Set: various

Procedure Name: various

EZB3956I Processing interface *interface*

Explanation: The indicated interface was found in the Routing Information Table (RIT) for the new client and is being added to NCPROUTE's tables.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSTART

Procedure Name: ifinit

EZB3957E Modify command ignored, invalid parm(s): *parms*

Explanation: Invalid parameters were passed to NCPROUTE from a MODIFY command.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the parameters from the MODIFY command.

Source Data Set: NRMAIN

Procedure Name:

EZB3958E Modify command ignored, client 'c=' not specified

Explanation: The parameters specified in the MODIFY command requires the target client specification.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Specify the target client in the "c=" parameter of the MODIFY command.

Source Data Set: NRMAIN

Procedure Name: parse_parms

EZB3959I point-to-point interface, using *addrtype*

Explanation: The new interface is point-to-point, and the destination address type, *addrtype*, can either be *dstaddr* or *broadaddr*. The address type is determined by the interface definition in the NCP generation. The destination address may be coded such that the network or subnetwork directed broadcast address is used or that the unicast or host address representing the other end of the point-to-point link is used. This address is used for sending routing information over an interface to the destination router or host. The exception is when the interface is multicast-capable; the multicast address is used.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSTART

Procedure Name: addrouteforif

EZB3960I This interface is not point-to-point *dstaddr*

Explanation: The new interface is not point-to-point. A route is being added based on the interface definition, and the network or subnetwork route will be used as the route destination.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSTART

Procedure Name: addrouteforif

EZB3961I not an internal interface

Explanation: The new interface does not appear to be associated with a real device, this interface is most likely a pseudo-interface created because of an external route in the GATEWAYS member for this client. No route is created for this interface.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSTART

Procedure Name: addrouteforif

EZB3962I Adding *type* route for interface

Explanation: The route using the network, subnetwork or destination type is being added to the interface in NCPROUTE's routing table.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSTART

Procedure Name: addrouteforif

EZB3963I Re-installing interface *interface*

Explanation: The previously deleted interface is being re-installed since traffic has been detected over this interface.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSTART

Procedure Name: addrouteforif

EZB3965I Route *seq; dest via gateway metric metric, supnetmask mask*

Explanation: A route is being added based on an IPROUTE statement coded during NCP generation. The *seq* is the position in the route table, *dest* is the destination IP address, *gateway* is the nexthop for the route, and *metric* is the cost of using this route, and *mask* is the subnet mask for the route.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSTART

Procedure Name: bld_rtbl

EZB3967W Unable to open a GATEWAYS dataset for client *client*. Attempt to open *dataset*

Explanation: A gateways data set was not specified, or could not be opened for the specified client. This message indicates which data set the open was attempted on.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: If a gateways data set is being used, verify that the PDS member name is coded correctly in the *hlq*.PROFILE.TCPIP data set. Verify that the gateways member has the same name as the client NCP name. Verify that NCPROUTE has access to this data set member. Verify that NCPROUTE attempts to open the correct data set member. Correct the PROFILE and rename the PDS member, if needed. Contact the IBM Software Support Center if the problem persists.

Source Data Set: NRSTART

Procedure Name: gateways

EZB3968I Start of GATEWAYS processing:

Explanation: The gateways data set member is about to be processed. Messages about data set processing might follow.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSTART

Procedure Name: gateways

EZB3969E *statement*

Explanation: The statement shown is in error. A message describing the error will follow.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSTART

Procedure Name: gateways

EZB3970E Invalid gateway address "*gateway*"

Explanation: This statement's gateway is neither a resolvable name nor a valid dot-notation internet address. The statement is ignored.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSTART

Procedure Name: gateways

EZB3971E Zero metrics not allowed, changing to one

Explanation: This statement has a 0 metric, which is not valid, Metrics must be between 1 and 15. The metric is changed to 1.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the statement in the gateway member.

Source Data Set: NRSTART

Procedure Name: gateways

EZB3972E Gateway type “passive” not valid for active gateway

Explanation: An active gateway entry is qualified as a passive route. Active is the only valid route type for this definition. The statement is ignored.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the statement in the gateway member.

Source Data Set: NRSTART

Procedure Name: gateways

EZB3973I Opening GATEWAYS dataset for client *client*. *dataset*

Explanation: The gateways data set is being opened for the specified client. Entries in the data set are read in for input.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSTART

Procedure Name: gateways

EZB3974E First two keywords must be 'active' for active gateway.

Explanation: A GATEWAYS entry for the gateway definition contains an active route type, but the first two keywords are not defined as active. These keywords are required for an active gateway definition. If this entry is not for an active gateway, correct the route type. Active gateway entries identify only a router and have no destination information. The GATEWAYS entry is ignored.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the GATEWAYS entry.

Source Data Set: NRSTART

Procedure Name: gateways

EZB3975E Invalid gateway type *type*

Explanation: This statement does not end with a valid route type, either “active”, “passive” or “external”. The statement is ignored.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the statement in the gateways member.

Source Data Set: NRSTART

Procedure Name: gateways

EZB3976S nr_exact_find: no session with *client*

Explanation: A route to a destination from a nonexistent client is being requested. This is an internal error. The route is reported as not existing.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: NRTABLES

Procedure Name: nr_exact_find

EZB3977S nr_exact_find: no network hash table for client *client*

Explanation: Client *client* does not appear to have a required network hash table. This is an internal error.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: NRTABLES

Procedure Name: nr_exact_find

EZB3978S nr_kernel_find: no host hash table for client *client*

Explanation: Client *client* appears to be missing a required host hash table. This is an internal error.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: NRTABLES

Procedure Name: nr_kernel_find

EZB3979S nr_kernel_find: no network hash table for client *client*

Explanation: The session with *client* has been improperly initialized. A required network hash table is missing.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: NRTABLES

Procedure Name: nr_kernel_find

EZB3980E nradd: invalid address family

Explanation: An attempt is being made to add a route which is to a destination in an unsupported address family. Currently, NCPROUTE only supports routes to Internet addresses. The add is not performed.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Identify the machine that generated the route and correct the problem.

Source Data Set: NRTABLES

Procedure Name: nradd

EZB3981S nradd: no host hash table for client *client*

Explanation: A host route is being added, but the client's host hash table cannot be located. This is an internal error. The route is not added.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Call the IBM Software Support Center.

Source Data Set: NRTABLES

Procedure Name: nradd

EZB3982S nradd: no network hash table for client *client*

Explanation: A network or subnetwork route is being added, but the required network hash table for *client* cannot be located. This is an internal error. The route is not added.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Call the IBM Software Support Center.

Source Data Set: NRTABLES

Procedure Name: nradd

EZB3983E Modify command ignored, *type* trace levels exceeded

Explanation: An incorrect number of trace levels (-t's) were passed from a MODIFY command.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Specify a correct number of -t's in the MODIFY command line parameters.

Source Data Set: NRMAIN

Procedure Name:

EZB3984E error adding route to *host/net destination* through gateway

Explanation: The NCP client failed to add a route.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Examine previous error messages to determine the nature of nr_do_add's error.

Source Data Set: NRTABLES

Procedure Name: nradd

EZB3989I *changing/deleting route to interface interface (timed out?)*

Explanation: Either traffic to a local interface is being routed through a remote gateway, or the metric on the interface has increased to infinity. Usually this indicates that the interface has timed out and is considered down.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: If the interface is being assigned a new gateway, examine the contents of the last packet sent from the gateway. The gateway's address is found in an upcoming message. Look for a display of the local interface from the remote gateway. Correct the remote gateway if needed. If the interface is timing out, verify that no transmissions have been received over this interface for the last 3 minutes. If no transmissions are found, this

EZB3993E • EZB3997E

is correct behavior for NCPROUTE, and the physical lines or remote gateways should be examined to determine the cause of the problem.

Source Data Set: NRTABLES

Procedure Name: nrchange

EZB3993E could not delete route to *type destination* via router

Explanation: The route to *destination* is being deleted (possibly because the route is changing, in which case an add will follow) and nr_do_delete returned an error. *type* is one of Host, Network, Subnet, or (bad_type). (bad_type) indicates an internal error and should be reported. Additional errors might follow because of the unsuccessful attempt to remove the route. These should be ignored until this problem is resolved. The route remains.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Examine previous messages to determine the error that occurred in nr_do_delete, contact the IBM Software Support Center with this information.

Source Data Set: NRTABLES

Procedure Name: nrchange

EZB3994E Error adding route

Explanation: An error occurred while attempting to update a route. The route is not readded and the net effect is that the route is deleted.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Examine previous messages to determine the error that occurred in nr_do_add and contact the IBM Software Support Center with this information.

Source Data Set: NRTABLES

Procedure Name: nrchange

EZB3996I deleting route to interface *interface?* (timed out?)

Explanation: A route to the indicated interface is being deleted and the metric is less than infinity.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTABLES

Procedure Name: nrdelete

EZB3997E Error deleting route

Explanation: The NCP client failed to add a route. Additional errors might follow this one. Because the route was not deleted, these should be ignored until this problem is resolved.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Examine previous error messages to determine the error that occurred in nr_do_delete and contact the IBM Software Support Center with this information.

Source Data Set: NRTABLES

Procedure Name: nrdelete

EZB3998S **nr_client_init: no session established with** *client*

Explanation: A session with the indicated client does not appear to have been initialized properly.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: NRTABLES

Procedure Name: nr_client_init

EZB3999I **Establishing session with client** *client*

Explanation: A session is being established with a NCP client. This is required before PDUs can be processed from the client.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTABLES

Procedure Name: nr_establish_session

Chapter 5. EZB4xxxx messages

EZB4000I Terminating session with client *client*

Explanation: The session with the indicated client ends. This is an orderly shutdown, and occurs in the following cases: upon receipt of a hello from an active client, if the load of the Routing Information Table (RIT) is unsuccessful, or if an error occurs when processing the RIT. Other messages occur earlier that explain which case caused the session to be brought down.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None required. If the session is brought down in response to an error, examine previous messages to determine the nature of the error, and correct the problem.

Source Data Set: NRTABLES

Procedure Name: nr_terminate_session

EZB4001S Terminate Session: no session with client *client*

Explanation: The NCP client failed to add a route. unacceptable client handle. The client cannot be located in the session table. No session ends.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: NRTABLES

Procedure Name: nr_terminate_session

EZB4002I Chain *chain_number*

Explanation: The session table is being displayed. Following this message is the contents of chain *chain_number*.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTABLES

Procedure Name: dsp_sessions

EZB4003I Hosts:

Explanation: The session table is being displayed. Following this message is the contents of the host route table for the current session.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTABLES

Procedure Name: dsp_sessions

EZB4004I Networks:

Explanation: The session table is being displayed. Following this message is the contents of the network route table for the current session.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTABLES

Procedure Name: dsp_sessions

EZB4005I No Entries

Explanation: This particular route table is empty.

System Action: NCPROUTE continues

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTABLES

Procedure Name: dsp_rtbl

EZB4006I Subchain *subchain_number*

Explanation: A routing table is being displayed. The following messages contain the contents of the routing table chain *subchain_number*.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTABLES

Procedure Name: dsp_rtbl

EZB4007I Entry empty

Explanation: The route table subchain currently being displayed is empty.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTABLES

Procedure Name: dsp_rtbl

EZB4008I Entry: *entry*

Explanation: The route table subchain currently being displayed contains an entry.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTABLES

Procedure Name: dsp_rtbl

EZB4009I client *client*: timer_value minute timer expired for route to *destination*

Explanation: No Routing Information Protocol (RIP) packets have been received from the NCP client to the destination *destination* in the last *timer_value* minutes. It is assumed that the destination route is no longer active. Depending upon the *timer_value*, one of the following actions is taken:

Value Action

- | | |
|---|--|
| 3 | The route will have its metric changed to infinity for the next 2 minutes. The metric change is necessary to alert adjacent routers that the route to this destination is unreachable. If NCPROUTE receives any RIP packets for the NCP client from the destination router during the 2 minute time interval, NCPROUTE will restore the route by changing the metric to a valid one based upon the received RIP packet. The route will be deleted from the NCP client's routing table. |
| 5 | The route will be deleted from NCPROUTE's routing table for the NCP client. |

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: These actions require the following responses:

Value Action

- | | |
|---|---|
| 3 | If the route was broadcasted for a while, and then suddenly stopped, look for an adapter problem, or a problem with the physical line. |
| 5 | Examine the NCPROUTE trace output and determine when the broadcasting has stopped for the route to the destination router. If the route to <i>destination</i> was only broadcast once in response to the NCPROUTE's request for full route tables for the NCP client, then the problem may be with the way RIP packets are broadcasted. |

In these cases, determine if NCPROUTE is receiving the transport PDUs containing the RIP packets from the NCP client by obtaining a MORETRACE IPUP trace and look for "discarding broadcast" packet messages. This trace can confirm the lack of traffic on the interface used for the transport PDUs forwarded by the NCP client. If this is not the case, then determine if NCPROUTE is receiving the RIP packets over the NCP client's routing interface by obtaining a NCP line trace. See the NCP documentation for instructions on obtaining a line trace.

Source Data Set: NRTIMER

Procedure Name: client_timer

EZB4010I client *client*: 30 second timer expired (broadcast)

Explanation: Every 30 seconds, a timer expires that indicates that a client NCP must broadcast its routing tables to adjacent routers. NCPROUTE will build Routing Information Protocol (RIP) response packets for each of the client's interfaces and send them to the client in a transport PDU. The client transmit them on the specified interface.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTIMER

Procedure Name: client_timer

EZB4011E Invalid subnetwork mask *mask*

Explanation: In the line entry for the NCP client's gateways data set, the gateway definition has a subnetmask that is not valid. The subnetmask must be a resolvable subnetmask name, or a bit mask in dotted-decimal notation. The gateway entry is ignored.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the NCP client's gateways data set.

Source Data Set: NRSTART

Procedure Name: ParseOptions

EZB4012E Trace buffers not initialized for interface *interface* on client *client*

Explanation: One or both trace buffers could not be obtained for the specified interface during initialization. This is because of a lack of free storage within the region. Minimal tracing is performed.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Increase the region size for NCPROUTE and restart NCPROUTE.

Source Data Set: NRTRACE

Procedure Name: traceinit

EZB4013I Tracing *action* for client *client*

Explanation: Tracing is enabled or disabled for a client NCP.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTRACE

Procedure Name: various

EZB4014W Unknown RIP *rip_control* control value (*value*)

Explanation: The RIP control values (supply or receive), specified on the RIP_SUPPLY_CONTROL or RIP_RECEIVE_CONTROL entry in the NCPROUTE profile data set, or specified on the options statement entry for a client for all interfaces or for an interface, contains an incorrect value. In case of incorrect values, NCPROUTE will default the RIP supply control to 'RIP1' and receive control to 'ANY' for the NCP client.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the NCPROUTE profile or the NCP client's gateways data set by specifying a supported supply or receive control value.

Source Data Set: NRMAIN, NRSTART

Procedure Name: read_profile, ParseOptions

EZB4015I Client tracing actions started

Explanation: The current tracing level has been advanced to the "actions" level, which causes messages to be issued for actions, such as adding, changing, or deleting routes. Additional messages for actions, such as waiting for incoming packets and dynamic updates, are also issued.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTRACE

Procedure Name: trace_start

EZB4016I Client tracing packets started

Explanation: The current tracing level is advanced to the "packets" level, which displays the types of packets sent and received in addition to the output displayed at the "actions" level.

System Action: NCPROUTE continues. Correct the *hlq*.NSMAIN.DATA data set to include a server name for the indicated address, or enter the NSLOOKUP command, using the *server_name* and *server_address* parameters to specify

the default Domain Name Server. For more information about the *hlq.NSMAIN.DATA* data set, see the *z/OS Communications Server: IP Configuration Reference*.

System Programmer Response: None.

Source Data Set: NRTRACE

Procedure Name: trace_start

EZB4017I Client tracing history started

Explanation: The current tracing level is advanced to the "history" level, which displays history tracing data for each line in addition to output displayed at the "packets" level. The history tracing data is displayed whenever an interface becomes inactive. It shows the latest traces of actions, packets and packet contents before the interface became inactive.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTRACE

Procedure Name: trace_start

EZB4018I Client tracing packet contents started

Explanation: The current tracing level is advanced to the "packet contents" level, which displays the contents of packets sent or received in addition to output displayed at lower tracing levels. Additional messages, such as requests for full routing tables and unknown address family in routing information, are also issued.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTRACE

Procedure Name: trace_start

EZB4029I *timestamp:*

Explanation: A full time stamp is issued showing the date and time so that traces that exceed one calendar day can be interpreted correctly.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTRACE

Procedure Name: trace

EZB4030I *action destination destination router router, metric metric, flags flags*

Explanation: The route to the destination is being added, deleted, or changed depending on the action. The following action values are allowed:

Value Explanation

ADD The route to the destination is being added through the router at the specified metric.

CHANGE FROM

The route to the destination is being changed and the current values are displayed.

CHANGE TO

The route to the destination is being changed and the new values are displayed.

EZB4032E • EZB4039I

DELETE

The route to the destination is being deleted.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTRACE

Procedure Name: trace

EZB4032E Gateway address '*address*' not an routing interface in network

Explanation: The indicated passive gateway, defined in the NCP client's *hlq.ETC.GATEWAYS* data set, referenced an unknown routing interface based upon the gateway address. The passive route definition is ignored.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the client's *hlq.ETC.GATEWAYS* data set. Verify that the gateway address is correct and a valid routing interface is defined. See the *z/OS Communications Server: IP Configuration Reference* for more information about NCP Host Interface Definition.

Source Data Set: NRSTART

Procedure Name: gateways

EZB4036I CHANGE metric destination *destination*, router *router*, from old metric to new metric

Explanation: The metric for the route to the destination is being changed from the old metric to the new metric. This is always based on a Routing Information Protocol (RIP) packet from the router, which updates a previous route through the router.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTRACE

Procedure Name: tracenewmetric

EZB4038I *** Packet history for interface *interface* ***

Explanation: Tracing is set at the history level, and the history trace data for the inactivated interface *interface* is displayed.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTRACE

Procedure Name: dumpif

EZB4039I *** End packet history ***

Explanation: Tracing is set at the history level, and this message ends the history trace data for the deactivated interface.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTRACE

Procedure Name: dumpif

EZB4043I *direction: no packets*

Explanation: Either the input or output trace buffer, depending on the direction value, is empty.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTRACE

Procedure Name: dumptrace

EZB4044I *direction trace:*

Explanation: Tracing is currently at the packet history level. Either the input or output trace buffer is about to be displayed depending on the direction value.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTRACE

Procedure Name: dumptrace

EZB4045I *RIPcmd direction router -> port timestamp p*

Explanation: A Routing Information Protocol (RIP) datagram has been received, or is about to be sent out of NCPROUTE depending on the direction value. A value of "to" indicates an outbound datagram, while a value of "from" indicates an inbound datagram. The type of the datagram is indicated by *RIPcmd*, and can be either RESPONSE or REQUEST. Responses are displays of routes from one router to another, requests are requests for individual routes or full tables. The value of *port* is either the port number which the packet came in on, the value 0, which indicates that the datagram will be sent to the default *RouteD* port as described in ETC.SERVICES, or in the case of outbound datagrams with a nonzero value the port which the datagram will be sent to.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTRACE

Procedure Name: dumppacket

EZB4046S **Bad cmd hex direction router -> port timestamp size=size cp=directino packet=addr time=timestamp**

Explanation: A malformed packet has been encountered during trace. The source of the packet is either the router, if the value of *direction* is "from", or NCPROUTE if the value of *direction* is "to".

System Action: NCPROUTE continues

User or Operator Response: None.

System Programmer Response: If the source of the packet is NCPROUTE, contact the IBM Software Support Center. Otherwise locate the router and take corrective action.

Source Data Set: NRTRACE

Procedure Name: dumppacket Routing Information Protocol (RIP)

EZB4048E (truncated record, len *len*)

Explanation: A Routing Information Protocol (RIP) datagram was received that did not end on a route boundary. Either the packet was too large, or it has been truncated.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Verify that the packet received was built correctly and can be processed with other routers. Contact the IBM Software Support Center if the problem appears to be with NCPROUTE.

Source Data Set: NRTRACE

Procedure Name: dumppacket

EZB4049I destination *destination* metric *metric*

Explanation: A route to the destination at the indicated metric is being displayed from NCPROUTE or an adjacent router depending on the contents of the last EZB4045I message. The metrics displayed to NCPROUTE by other routers do not have the interface metrics added to them, this is done when the route is added to NCPROUTE's tables. Likewise, NCPROUTE displays routes at a metric which does not include the metric of the interface on which the route is being displayed. A route is never displayed on the interface from which it was received.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTRACE

Procedure Name: dumppacket

EZB4050I (request for full tables)

Explanation: A request for a complete routing table is being sent, or has been received depending on the contents of the last EZB4045I message. This message is sent during session initialization by NCPROUTE. In addition to foreign routers making this request, application programs can ask a client for its tables by sending a Routing Information Protocol (RIP) request for a route with an address family of 0 and a metric of 16 to a client NCP at the current *RouteD* port (by default, 520).

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTRACE

Procedure Name: dumppacket

EZB4051E unknown address family *family* metric *metric*

Explanation: A route in an unacceptable address family has been received, or is about to be sent depending on the contents of the last EZB4045I message. Currently, only AF_INET is supported, although AF_UNSPEC is allowed when making requests for full tables.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: If the route is being sent from NCPROUTE, contact the IBM Software Support Center. Otherwise locate the router listed in the last EZB4045I message and correct it so that noninternet routes are not sent to the client NCP.

Source Data Set: NRTRACE

Procedure Name: dumppacket

EZB4052I **traceon file = *dataset***

Explanation: A TRACEON packet was received. Tracing is requested to go to the specified data set.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTRACE

Procedure Name: dumppacket

EZB4055I **attempting to (re)start SNMP connection**

Explanation: NCPROUTE is attempting to establish a connection with the SNMP agent specified in the profile. This occurs during startup, and at any time when an SNMP request occurs and no connection exists with the agent.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: init_snmp

EZB4056W **no response from agent on *host***

Explanation: NCPROUTE was unable to establish a connection with the SNMP agent on host *host*.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Verify that an agent is running on *host*, and that IP routes exist between the local host and *host*. When tracing is activated on the SNMP agent, DPI requests should be seen coming from NCPROUTE.

Source Data Set: NRSNMP

Procedure Name: init_snmp

EZB4057W **Start the agent before issuing SNMP queries.**

Explanation: A reminder that the SNMP agent should be running before NCPROUTE will be able to process SNMP queries for its clients.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: init_snmp

EZB4058E **An error occurred while opening the SNMP socket:**

Explanation: An error occurred while attempting to open a socket for communicating with the SNMP agent. A more detailed tcperror() library message follows.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the problem as indicated by the error in the detailed tcperror() library message. Refer to the *z/OS C/C++ Run-Time Library Reference* for more information about socket() function errors.

Source Data Set: NRSNMP

Procedure Name: init_snmp

EZB4059I Connecting to agent *agent* on DPI port *port*

Explanation: NCPROUTE is attempting to connect to the SNMP agent using the DPI port.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: init_snmp

EZB4060E An error occurred while connecting to the SNMP agent:

Explanation: NCPROUTE was unsuccessful in connecting to the SNMP agent. A more specific error message will follow.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center

Source Data Set: NRSNMP

Procedure Name: init_snmp

EZB4061E A connection is required for processing SNMP requests.

Explanation: A reminder that the SNMP agent socket must be connected before SNMP queries are processed.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: init_snmp

EZB4062I SNMP DPI connection established

Explanation: The connection with the SNMP agent is established and SNMP queries will now be processed as they arrive.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: init_snmp

EZB4063E Unable to register *root* with SNMP agent.

Explanation: NCPROUTE was unable to register the MIB extension root with the agent. An I/O error occurred while sending the registration to the agent. The connection to the agent is closed and will be reopened for the next query.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Verify that the agent is operating correctly and that the agent's host is currently reachable from the local host.

Source Data Set: NRSNMP

Procedure Name: init_snmp

EZB4064I *root* **registered with SNMP agent**

Explanation: The MIB extension root has been registered with the SNMP agent.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: init_snmp

EZB4065E **Unable to establish a session with the SNMP agent.**

Explanation: An attempt to reestablish a connection with the SNMP agent has failed.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Verify that the SNMP agent is running. Verify that the SNMP statements in the NCPROUTE.PROFILE are correct, and verify that the host running the SNMP agent can be reached from the local host.

Source Data Set: NRSNMP

Procedure Name: snmp_input

EZB4066E **An incoming SNMP packet was discarded (noagent)**

Explanation: Because no connection exists with the SNMP agent, the SNMP packet being processed is discarded.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: snmp_input

EZB4067E **Error while forwarding SNMP packet to agent:**

Explanation: NCPROUTE was unable to forward the SNMP request up to the SNMP agent. A more specific error message follows. The SNMP packet is discarded.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Verify that the SNMP agent is running and that the host running the agent is reachable from the local host.

Source Data Set: NRSNMP

Procedure Name: snmp_input

EZB4068I SNMP response from local agent

Explanation: The SNMP agent has responded to a forwarded request.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: snmp_response

EZB4069E Error while sending SNMP reply to client *client*

Explanation: NCPROUTE was unable to send the response to a SNMP query back to the client. The packet is discarded.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Examine the following error message, and take whatever corrective action is recommended.

Source Data Set: NRSNMP

Procedure Name: snmp_response

EZB4070S Couldn't parse incoming DPI packet! (dropping) This suggests a problem with the SNMP agent.

Explanation: NCPROUTE could not parse an incoming DPI packet from the SNMP agent. The packet is damaged or has not been built properly. Either the SNMP agent or the SNMP dpi library is in error. Also, they may be at different maintenance levels.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: NRSNMP

Procedure Name: dpi_in

EZB4071I *rip_control rip_version* packets on interface *interface* not allowed

Explanation: A RIP Version 1 or Version 2 packet is ignored depending upon the settings of the RIP supply or receive controls specified in the NCP client's gateways data set for an interface or in the NCPROUTE profile data set. If there are no RIP control settings for an interface, NCPROUTE will use the one from the profile settings.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRINPUT, NROUTPUT

Procedure Name: rip_input, supply

EZB4072I SNMP: DPI GET request (*oid*) received.

Explanation: An SNMP DPI get request for the variable specified by the ASN.1 object identifier was received. The object identifier is a registered MIB extension, and if you remove the root of the object identifier (based on the contents of EZB4064I) you will have a client identifier followed by the original object identifier.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: dpi_in

EZB4073I SNMP sub-agent: DPI GET NEXT request (*oid*) received.

Explanation: An SNMP DPI get next request for the specified object identifier has been received.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: dpi_in

EZB4074I SNMP sub-agent: DPI SET request received

Explanation: An SNMP DPI set request was received. SNMP set commands are not supported in this release.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: dpi_in

EZB4075S Unexpected SNMP query type *type*; SNMP support appears incomplete in NCPROUTE.

Explanation: An unexpected SNMP DPI packet type has been received from the SNMP agent. The packet has already been validated by the DPI library routines, so this indicates an unsupported valid DPI packet type. An error is returned to the originator of the SNMP packet.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: NRSNMP

Procedure Name: dpi_in

EZB4076I RIP2 packet from router *router* not authorized

Explanation: A RIP Version 2 packet, received from *router*, is ignored due to a authentication key mismatch. Authentication is enabled for RIP Version 2 packets according to the interface options in the NCP client's gateways data set or in the NCPROUTE profile data set. If there are no interface settings, NCPROUTE will use the one from the profile settings.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRINPUT

Procedure Name: rip_input

EZB4077S SNMP sub-agent: received DPI request outside tree

Explanation: The SNMP agent has requested information or action on a MIB variable that is outside of the tree that NCPROUTE manages. This indicates an error in the agent code, because NCPROUTE did not register this variable. NO_SUCH_NAME is returned.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Compare the object identifier for the current DPI packet with the region of the MIB tree that NCPROUTE manages. The object identifier can be found in the last EZB4072I, EZB4073I, or EZB4074I message issued. The registered region can be found in the last EZB4064I message issued.

Source Data Set: NRSNMP

Procedure Name: dpi_in

EZB4078I SNMP sub-agent: received invalid DPI request outside tree

Explanation: The DPI packet from the agent refers to an OID which is not present in the managed tree.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: dpi_in

EZB4079I iproutedest.instance

Explanation: A request has been made for the destination of the specified route. Each route in the client's table is numbered starting with 0.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: snmp_lookup

EZB4080I iprouteifindex.destination

Explanation: A request has been made for the interface index associated with the route to *destination* for the client which forwarded this request to NCPROUTE.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: snmp_lookup

EZB4081I iproutemetric1. destination

Explanation: A request has been made for the metric associated with the route to the destination for the client that forwarded this request to NCPROUTE.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: snmp_lookup

EZB4082I *iproutemetric(2-4).destination (unsupported)*

Explanation: A request was made for one of the alternate route metrics that is not used under RIP. If a route to the specified destination exists, a value of '-1' is returned; otherwise, NO_SUCH_NAME will be returned.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: snmp_lookup

EZB4083I *iproutenexthop.instance*

Explanation: A request was made for the next hop for the specified route. Routes are specified by the instance number, which is the ordinal position within NCPROUTE's tables. If the specified instance exists, an Internet address will be returned; otherwise, NO_SUCH_NAME will be returned.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: snmp_lookup

EZB4084I *iproutetype.instance*

Explanation: A request was made for the route type for the specified route. If the specified instance exists, one of three values will be returned depending on the route state:

Value	Explanation
Direct	The route is to a directly connected destination.
Invalid	The route has an infinite metric.
Remote	The route is to an indirectly connected destination requiring one or more routers to reach.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: snmp_lookup

EZB4085I *iprouteproto.instance*

Explanation: A request was made for the mechanism by which the route for the specified instance was determined. Values returned will be 'RIP' or 'Other' depending on whether the route was learned dynamically or was manually entered.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

EZB4086I • EZB4090E

Procedure Name: snmp_lookup

EZB4086I *iprouteage.instance*

Explanation: A request was made for the age of the route of the specified instance. The value returned will be the length of time the route has been active.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: snmp_lookup

EZB4087I *iproutemask.instance*

Explanation: A request was made for the network or subnetwork mask of route of the specified instance. The value returned will be the network or subnetwork mask value in dotted-decimal notation.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: snmp_lookup

EZB4088E **A request was made for an unsupported MIB variable:** *variable*

Explanation: A request was made for an unsupported MIB variable. Only variables in the iproute group are returned. NO_SUCH_NAME will be returned for the unsupported MIB variable.

System Action: NCPROUTE continues.

User or Operator Response: Reissue the request with a valid MIB variable from the iproute group.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: snmp_lookup

EZB4089E **An attempt was made to change a MIB variable. SNMP set request is not currently supported.**

Explanation: An attempt was made to change a MIB variable. Variables in the iproute group may be queried but not changed. An SNMP_GEN_ERR is returned.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: dpi_set

EZB4090E **SNMP buffer overrun:** *number bytes*

Explanation: An SNMP request will exceed NCPROUTE's internal buffer size during an edit. This almost always indicates an incorrect request because no supported object identifier is large enough to cause this. The buffer is discarded.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Verify the object identifier being requested is for an iproute MIB variable. If it is, contact the IBM Software Support Center.

Source Data Set: NRSNMP

Procedure Name: edit_obj

EZB4125I ASN.1 type at *count* bytes into packet: Class=*class* Field=*field*

Explanation: An ASN.1 object such as an object identifier or an integer was located in the SNMP packet being displayed. The count is the offset into the packet where the object was located. The object's tag's class bits are formatted and displayed as *class*. An object's class can be viewed as the scope of the object identifier. ASN.1 defines four values:

Value Explanation

Universal

Well-known tags. NCPROUTE wants this type of tags.

Application-Wide

Tags local to the application. NCPROUTE does not have any of these.

Context-Specific

Used in ASN.1 constructors. Not applicable to NCPROUTE.

Private-Use

NCPROUTE does not use any private tags.

The object's tag's field bits are formatted and displayed as the field. The field is either primitive or constructed.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: decode

EZB4126E Modify command ignored, invalid client address

Explanation: The target client specified in the MODIFY command is not valid.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the address for the client target.

Source Data Set: NRMAIN

Procedure Name: do_modify

EZB4127E Modify command ignored, no session with client *client*

Explanation: The target client specified in the MODIFY command is unknown. No session exists between NCPROUTE and the client.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Wait for NCPROUTE to establish a session with the NCP client or respecify the target client using an address known by NCPROUTE.

Source Data Set: NRMAIN

Procedure Name: do_modify

EZB4128I **Reserved for addenda** (*type*)

Explanation: The type value is not a known object tag type. It was described as reserved for addenda in RFC 1158. See Appendix B, "Related protocol specifications (RFCs)," on page 271 for information about accessing RFCs.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: decode

EZB4129I *type*

Explanation: The formatted ASN.1 object tag's number. NCPROUTE formats Universal tags as defined in RFC 1158. See Appendix B, "Related protocol specifications (RFCs)," on page 271 for information about accessing RFCs.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: decode

EZB4130I **Misc type** *type*

Explanation: This object's class indicates that it is not a well-known type. The unformatted object type is displayed, NCPROUTE only formats objects with a class of Universal.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: decode

EZB4131I **Encountered editable OID at offset** *offset*

Explanation: An object identifier was located at the indicated offset into the packet. NCPROUTE will need to translate this OID to another region of the MIB tree so that the agent will recognize the request as being for one of NCPROUTE's clients and not for the host running the agent.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: decode

EZB4132I **Length =** *length*

Explanation: The length of the decoded ASN.1 object is displayed.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: decode

EZB4133I Adding type route route destination via gateway gateway, metric metric

Explanation: The indicated route, defined in the client's *hlq.ETC.GATEWAYS* data set, is added to the NCP's IP routing table. The route to the gateway will not be replaced by a competing RIP route.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSTART

Procedure Name: gateways

EZB4134W Subnetwork mask unknown for destination, using network route route

Explanation: The indicated subnetwork route, defined in the client's *GATEWAYS* data set, was explicitly coded as a "net" route type. Because the subnetwork mask for the destination subnetwork is unknown, NCPROUTE replaces the subnetwork route with a network route. NCPROUTE currently does not support variable subnetting.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTABLES

Procedure Name: nradd

EZB4135E Invalid host or (sub)network address '*destination*'

Explanation: In the line entry for the client's *hlq.ETC.GATEWAYS* data set, the gateway definition has an incorrect destination address. The *destination* must be either a resolvable host name or an Internet address in dotted notation. The gateway entry is ignored.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the client's *hlq.ETC.GATEWAYS* data set.

Source Data Set: NRSTART

Procedure Name: gateways

EZB4136W Second element ignored, changing to 'active'

Explanation: The second element in the active gateway entry is detected to be incorrect. NCPROUTE will change the element to be ACTIVE since it is likely that the gateway entry is in error. With this change, the active gateway entry is processed normally.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the client's *hlq.ETC.GATEWAYS* data set.

Source Data Set: NRSTART

Procedure Name: gateways

EZB4138W **Unknown next hop address** *ipaddr* **for route** *destination* **from router** *router*

Explanation: An unknown next hop address *ipaddr* was received in a RIP packet for a route to *destination* from from *router*. The route is ignored.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Locate the router that produced the packet and correct the problem. It may involve router reconfiguration.

Source Data Set: NRINPUT

Procedure Name: rip_input

EZB4139S **tblclr: no host hash table for client** *client*

Explanation: A required host hash table could not be located for the client. NCPROUTE was attempting to remove all routes for the specified client from its tables. Any routes will be left in NCPROUTE's tables.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: NRTABLES

Procedure Name: tblclr

EZB4141S **tblclr: no network hash table for client** *client*

Explanation: A required network hash table could not be located for the client, but the host hash table has already been found and cleared. NCPROUTE was attempting to remove all network routes for the client from its tables. Any network routes for the client remain in NCPROUTE's tables.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: NRTABLES

Procedure Name: tblclr

EZB4142E **The transmission of an 'Ack' to client** *client* **failed.**

Explanation: TCP/IP detected that an "Acknowledge" PDU could not be delivered successfully before any transmission was performed. The PDU is discarded. A more specific error message will immediately follow.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Examine the following error message and follow the documented procedures.

Source Data Set: NRPDUS

Procedure Name: send_ack

EZB4143S **No SNMP_AGENT statement in profile**

Explanation: A required SNMP_AGENT statement was missing from the NCPROUTE profile. This statement must identify the host which runs the SNMP agent which NCPROUTE will use to resolve queries. SNMP requests will not be honored by NCPROUTE (or its clients).

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the profile.

Source Data Set: NRMAIN

Procedure Name: read_profile

EZB4144S SNMP requests will not be honored.

Explanation: One or more severe errors were encountered which will result in a connection with the SNMP agent not being established. Until these errors are resolved, and NCPROUTE is restarted, SNMP requests will not be honored by NCPROUTE and its clients in turn.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct any previously identified errors.

Source Data Set: NRMAIN

Procedure Name: read_profile

EZB4145S * No SNMP_COMMUNITY statement in NCPROUTE profile

Explanation: A required SNMP_COMMUNITY statement is missing from the NCPROUTE profile. This statement identifies the SNMP community which will be used when forwarding SNMP requests to the agent. SNMP requests will not be honored.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the profile and restart NCPROUTE.

Source Data Set: NRMAIN

Procedure Name: read_profile

EZB4146W Required NCPROUTE profile dataset (*profile*) not found

Explanation: The optional profile configuration data set for NCPROUTE could not be opened successfully. This message indicates which data set the open was attempted on.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Verify that the profile data set is defined in //DD:NCPRPROF of the NCPROUTE start proc JCL and is accessible by NCPROUTE. If the data set is sequential, ensure that the FREE=CLOSE parameter is specified. Do not specify this parameter if the data set is partitioned.

Source Data Set: NRMAIN

Procedure Name: read_profile

EZB4147S Unknown keyword (*keyword*) in PROFILE

Explanation: An unknown keyword *keyword* was used in a statement in the NCPROUTE profile. The statement is ignored.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the NCPROUTE profile.

Source Data Set: NRMAIN

Procedure Name: read_profile

EZB4148E client *client*: failed attempt to add route to *destination*.

Explanation: NCPROUTE's attempt to add a route to the client was rejected by the client. Most likely the client's route tables are full so that it could not accept any more routes, or the NCP is attempting to add a route specified as PERM during generation.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Verify the routing table size values in the NUMROUTE keyword in the IPOWNER statement of the NCP generation. Increase the values, if necessary.

Source Data Set: NRPDUS

Procedure Name: add_fail

EZB4149E client *client*: failed attempt to delete route to *destination*.

Explanation: Client *client* rejected NCPROUTE's attempt to delete a route to *destination* from the client's tables. Most likely the route did not exist in the client's tables.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Contact the IBM Software Support Center.

Source Data Set: NRPDUS

Procedure Name: delete_fail

EZB4150I End of GATEWAYS processing

Explanation: Processing is completed for the GATEWAYS data set member.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSTART

Procedure Name: gateways

EZB4151T Out of memory while processing GATEWAYS

Explanation: NCPROUTE has exhausted available storage while processing a client's GATEWAYS data set member.

System Action: NCPROUTE ends abnormally.

User or Operator Response: None.

System Programmer Response: Increase NCPROUTE's region size and restart. Take into consideration that storage requirements are based on the number of clients being served and the number of routes being managed. If the problem still cannot be resolved, contact the IBM Software Support Center.

Source Data Set: NRSTART

Procedure Name: gateways

EZB4152I Adding active gateway *ip_addr*, metric *metric*

Explanation: The indicated active gateway, which is defined in the client's *hlq.ETC.GATEWAYS* data set, is added to NCPROUTE's routing table. The route to the active gateway will be treated as a network interface.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSTART

Procedure Name: gateways

EZB4154E Invalid option: *option*

Explanation: In the line entry for the client's *hlq.ETC.GATEWAYS* data set, the NCPROUTE's server options definition has an incorrect option. Although other options may be processed normally, the incorrect option is ignored.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the client's *hlq.ETC.GATEWAYS* data set. See the *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: NRSTART

Procedure Name: gateways

EZB4155E Invalid default router value: *value*

Explanation: In the line entry for the client's *hlq.ETC.GATEWAYS* data set, the NCPROUTE's server options definition has an incorrect default router value. The incorrect default router value is ignored and the default value is used.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the client's *hlq.ETC.GATEWAYS* data set by specifying a valid default router value. See the *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: NRSTART

Procedure Name: gateways

EZB4156E Invalid trace level: *level*

Explanation: In the line entry for the client's *hlq.ETC.GATEWAYS* data set, the NCPROUTE's server options definition has an incorrect trace level. The incorrect trace level is ignored and the default value is used.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the client's *hlq.ETC.GATEWAYS* data set by specifying a valid trace level value. See the *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: NRSTART

Procedure Name: gateways

EZB4157E Invalid type value: *value*

Explanation: In the line entry for a client's *hlq.ETC.GATEWAYS* data set, the options definition contains an invalid value. Although other options may be processed normally, the invalid option is ignored.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the client's *hlq.ETC.GATEWAYS* data set. See *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: NRSTART

Procedure Name: gateways

EZB4158I Global tracing at all levels suppressed

Explanation: Trace levels for all NCP clients are not displayed. This includes the trace levels specified in the clients' *hlq.ETC.GATEWAYS* data set.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTRACE

Procedure Name: global_trace_start

EZB4159I Global tracing actions started

Explanation: Trace levels for all NCP clients have been advanced to the "actions" level, which causes messages to be issued for actions such as adding, changing, or deleting a route. Additional messages for actions such as waiting for incoming packets and dynamic updates are also issued. The trace levels specified in the client's *hlq.ETC.GATEWAYS* data sets are also advanced to this level.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTRACE

Procedure Name: global_trace_start

EZB4160I Global tracing packets started

Explanation: Trace levels for all NCP clients have been advanced to the "packets" level, which displays the types of packets sent and received in addition to the output displayed at the "actions" level. The trace levels specified in the client's *hlq.ETC.GATEWAYS* data sets are also advanced to this level.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRTRACE

Procedure Name: global_trace_start

EZB4161T Global trace levels exceeded maximum of 2 -t's. For each client, use the MODIFY command or GATEWAYS dataset to specify additional tracing options.

Explanation: An incorrect number of trace levels (-t's) was passed from *hlq.SEZAINST(NCPROUTE)* start proc JCL. The parameter(s) could either be passed from the command line parameters or from the default parameter list in the start proc JCL.

System Action: NCPROUTE exits.

User or Operator Response: None.

System Programmer Response: Specify a correct number of -t's in the command line parameters or in the default parameter list of the start proc JCL. If a higher trace level is desired, specify the trace level in the options statement of a client's *hlq.ETC.GATEWAYS* data set. Another option is to use the MODIFY command to increase the trace levels for the client after NCPROUTE has started and established a session with the client. Refer to *z/OS Communications Server: IP Configuration Reference* for more information.

Source Data Set: NRMAIN

Procedure Name: parse_parms

EZB4162I Deferring add route to *destination*

Explanation: The addition of a new route to the specified destination in the NCP client's routing table is deferred until after the NCP client has finished initialization. The new route is copied to a static buffer, which is used after the Inactive Interface List PDU has been received from the NCP client. NCPROUTE will perform the add from the static buffer.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRPDUS

Procedure Name: defer_add

EZB4163E Too many deferred routes

Explanation: The number of static buffers to hold the new routes for deferred addition to the NCP clients' routing tables has exceeded the maximum of 500. This may happen when multiple NCP clients have not completed their initialization processes at the same time. At this time, the new routes are not added to the NCP client's routing tables. The static buffers do not become available until after NCPROUTE has received the Inactive Interface List PDUs from the NCP clients. These PDUs tell NCPROUTE that the NCP clients have completed their initialization processes.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Allow sufficient time for the NCP clients to complete their initialization processes so that the new routes can be added to the client's routing tables. Contact the IBM Software Support Center if the problem persists.

Source Data Set: NRPDUS

Procedure Name: defer_add

EZB4164I Init: compare with *client*

Explanation: The client's Internet address is being compared with one of NCPROUTE's session table entries for an exact match.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRPDUS

Procedure Name: ncp_initialized

EZB4165E Client *client* not in session table

Explanation: The client's Internet address was not found in the session table. This indicates that NCPROUTE has not established a session with the NCP client. NCPROUTE will not manage the routes for the client until a "Hello" PDU has been received for session establishment.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Allow sufficient time for the NCP client to issue the "Hello" PDUs so that a session can be established with NCPROUTE. If a Hello PDU was received and the problem persists, contact the IBM Software Support Center.

Source Data Set: NRPDUS

Procedure Name: ncp_initialized

EZB4166I Session with client *client* started

Explanation: NCPROUTE has successfully established a session with the specified NCP client through the handshaking process and is waiting to receive an Inactive Interface List PDU from the client. This PDU indicates that the client has completed initialization and is ready for route table management by NCPROUTE.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRPDUS

Procedure Name: recv_hello

EZB4167E SNMP requester *requester* is not reachable by NCP client *client*

Explanation: NCPROUTE was ready to transport the SNMP response packet to the NCP client to be forwarded to the SNMP requester, but the route to the SNMP requester was detected to be unavailable. A possible cause is that the NCP client's interface to the SNMP requester became inactive. The SNMP response packet is discarded.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Verify the status of the NCP client's interface to the SNMP requester. If the interface was inactive, activate the interface so that a route can be created; otherwise, determine whether the route to the SNMP requester was timed out. If the problem cannot be corrected, contact the IBM Software Support Center.

Source Data Set: NRSNMP

Procedure Name: snmp_response

EZB4168I SNMP lookup failed: *reason*

Explanation: NCPROUTE could not obtain the routing information based upon the SNMP request packet sent by the SNMP requester. In this case, NO_SUCH_NAME is returned to the SNMP requester. Reasons for the failure are:

Reason Explanation**NO INSTANCES PROVIDED**

The target route was not specified in the SNMP request; however, this is an exception to SNMP GET_NEXT requests.

ROUTE TABLE EMPTY

There were no routing table entries for this interface based upon the target route.

ROUTE NOT FOUND

The target route was not found in the routing table.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Verify the accuracy of the target route in the SNMP request. If the problem cannot be corrected, contact the IBM Software Support Center.

Source Data Set: NRSNMP

Procedure Name: snmp_lookup

EZB4172I SNMP reply sent to NCP client *client*

Explanation: NCPROUTE is sending the reply containing the SNMP response packet to the NCP client to be forwarded to the SNMP requester.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: snmp_response <

EZB4173I SNMP get_next: no session with client *client*

Explanation: During processing of the SNMP GET_NEXT request, NCPRROUTE has detected that there was no session with the NCP client. The SNMP request packet is discarded.

System Action: NCPRROUTE continues.

User or Operator Response: None.

System Programmer Response: If the session with the NCP client appears to have terminated, examine previous messages to determine the cause. When necessary, take corrective actions to allow NCPRROUTE to process the SNMP requests for the NCP client.

Source Data Set: NRSNMP

Procedure Name: get_next

EZB4174I SNMP get_next: searching for next route after *target*

Explanation: During processing of the SNMP GET_NEXT request, NCPRROUTE is searching for the next route after the target route in the routing tables for the NCP client.

System Action: NCPRROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: get_next

EZB4175I SNMP get_next: found next route *ip_addr*

Explanation: During processing of the SNMP GET_NEXT request, NCPRROUTE has found the next route entry after the target route in the routing tables for the NCP client. The entry contains the next route's *ip_addr*.

System Action: NCPRROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: get_next

EZB4176I SNMP get_next: compare with *target*

Explanation: The target route specified in the SNMP GET_NEXT request packet received by NCPRROUTE from an NCP client is being compared with one of NCPRROUTE's routing table entries for an exact match.

System Action: NCPRROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: get_next

EZB4177I SNMP get_next: addresses matched

Explanation: A match was found for the target route specified in the SNMP GET_NEXT request packet with one of NCPROUTE's routing table entries for an NCP client.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: get_next

EZB4178I RIP2 authentication action at level level (interface)

Explanation: RIP Version 2 authentication is enabled or disabled at the specified level for an interface for for all interfaces in the NCP client, or for all NCP clients.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRMAIN, NRSTART

Procedure Name: read_profile, ParseOptions

EZB4179E SNMP object length exceeded maximum 64K

Explanation: During the object data conversion process for an SNMP packet, NCPROUTE detected an incorrect length value in the header portion of the object data. NCPROUTE could not continue processing for the value has exceeded the maximum length of 64K. The SNMP packet may have been built incorrectly.

System Action: NCPROUTE exits.

User or Operator Response: None.

System Programmer Response: Contact your IBM Software Support Center.

Source Data Set: NRSNMP

Procedure Name: store_int

EZB4180I Packet from router router ignored (filtered out)

Explanation: NCPROUTE is ignoring the RIP packet as a result of being filtered out according to a RIP input or output filter defined in the NCP clients's gateways data set.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRINPUT

Procedure Name: rip_input

EZB4181I Interface interface skipped, interface is multicast-incapable

Explanation: The interface *interface* is skipped because the interface is not capable of multicasting RIP Version 2 packets. The RIP2 supply control has been configured for the interface and disallows the use of broadcasting RIP packets over the interface.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: When permitted, use the RIP2M supply control option to allow broadcasting RIP Version 1 packets over multicast-incapable interfaces and multicasting Version 2 packets over the multicast-capable interfaces.

Source Data Set: NROUTPUT

Procedure Name: toall_ifs

EZB4182I SNMP request received from NCP client *client*

Explanation: An SNMP query request was received from an NCP client for processing. The NCP client had received the query request from its SNMP client or the SNMP requester.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSNMP

Procedure Name: snmp_input

EZB4191I SNMP get-next: next route not found (*reason*)

Explanation: NCPROUTE is processing the SNMP GET_NEXT request but it could not obtain the routing information for the next route. In this case, NO_SUCH_NAME is returned to the SNMP requester. Reasons for the unavailable route are:

Reason Explanation

NOT IN TABLE

The target route was not found in the routing table. This implies that the route entry for the next route cannot be determined.

END OF TABLE

The target route was found at the end of the routing table but there were no more route entries for the next route.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: These reasons require the following responses:

Reason Response

NOT IN TABLE

Verify the accuracy of the target route in the SNMP request.

END OF TABLE

None.

If the problem cannot be corrected, contact the IBM Software Support Center.

Source Data Set: NRSNMP

Procedure Name: snmp_lookup

EZB4194I SNMP sub-agent received DPI request

Explanation: NCPROUTE is performing the role as an SNMP sub-agent and is in session with the SNMP agent over the Distributed Program Interface (DPI). NCPROUTE has received a DPI request from the SNMP agent for processing.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

EZB4195I • EZB4198I

Source Data Set: NRSNMP

Procedure Name: dpi_in

EZB4195I **Option(s):** *options*

Explanation: Additional NCPROUTE options, specified in a client's *hlq.ETC.GATEWAYS* data set member, are being processed. Some options may be overridden by the parameter list in the *hlq.SEZAINST(NCPROUT)* start proc JCL. See the *z/OS Communications Server: IP Configuration Reference* for more information.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSTART

Procedure Name: gateways

EZB4196I **Opening NCPROUTE profile data set** *datasetname*

Explanation: The specified NCPROUTE profile data set is being opened. Entries in the data set are read in for input.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRMAIN

Procedure Name: read_profile

EZB4197E **Invalid route type** *'type'*

Explanation: In the line entry for the NCP client's *hlq.ETC.GATEWAYS* data set, the gateway definition has an incorrect route type. Allowable route types are HOST for host route, NET for network or subnetwork route, and ACITIVE for a route to be treated as a network interface. The gateway entry is ignored.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: Correct the NCP client's *hlq.ETC.GATEWAYS* data set.

Source Data Set: NRSTART

Procedure Name: gateways

EZB4198I **(no etc.gateway definitions)**

Explanation: The NCP client's *hlq.ETC.GATEWAYS* data set contains no optional gateway definitions.

System Action: NCPROUTE continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: NRSTART

Procedure Name: gateways

Chapter 6. EZB6xxxx messages

EZB6473I TCP/IP STACK FUNCTIONS INITIALIZATION COMPLETE.

Explanation: TCP/IP has been successfully initialized.

System Action: TCP/IP continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: EZBTIINI

Chapter 7. EZB8xxxx messages

EZB8801I *probenm* ATTEMPTED, FFST NOT AVAILABLE.

Explanation: TCP/IP encountered an anomaly and attempted to execute a FFST probe. FFST was not available to service the request. The request was handled by furnishing the identification of the probe.

In the message text:

probenm

The probe identifier.

Source Data Set: EZBITZPF

System Action: TCP/IP continues.

User or Operator Response: Determine why FFST is not available. Gather what probe documentation is available and contact your IBM support center.

System Programmer Response: Determine why FFST is not available. Gather what probe documentation is available and contact your IBM support center.

Chapter 8. EZB9xxxx messages

Common messages

This section contains messages that are called by several application and function components. When these messages are called, only the message text will be appended to the message that called it. The message will inherit the prefix (EZA or EZB) of the message that called it. They are documented fully in *z/OS Communications Server: IP Messages Volume 1 (EZA)*.

The following is a complete list of these common messages.

- EZB9395I
- EZB9396I
- EZB9397I
- EZB9398I
- EZB9399I
- EZB9400I
- EZB9401I
- EZB9402I
- EZB9403I
- EZB9404I
- EZB9405I
- EZB9406I
- EZB9407I
- EZB9408I
- EZB9409I
- EZB9410I
- EZB9411I
- EZB9412I
- EZB9413I
- EZB9414I
- EZB9415I
- EZB9416I
- EZB9417I
- EZB9418I
- EZB9419I
- EZB9420I
- EZB9421I
- EZB9422I
- EZB9423I
- EZB9424I
- EZB9428I

Chapter 9. EZD0xxxx messages

EZD0001I SETTING VLAN ID NOT SUPPORTED FOR DEVICE *device_name*

Explanation: TCP/IP could not set the VLAN ID for this device. The OSA-Express adapter microcode level does not support setting the VLAN ID.

device_name is the name of the device.

System Action: TCP/IP continues with device activation but does not use a VLAN ID.

User or Operator Response: Inform the system programmer about the error.

System Programmer Response: Install a level of OSA-Express microcode that supports the VLAN ID function. Use the VTAM DISPLAY TRL command to determine the OSA-Express microcode level. For more information about the VTAM DISPLAY TRL command, refer to *z/OS Communications Server: SNA Operation*.

Source Data Set: TCPIP

Procedure Name: EZBIFIND

EZD0002I ERROR SETTING VLAN ID FOR DEVICE *device_name*

Explanation: An unexpected error occurred while setting the VLAN ID for the device.

device_name is the name of the device.

System Action: TCP/IP continues with device activation but does not use a VLAN ID.

User or Operator Response: Inform the system programmer about the error.

System Programmer Response: Obtain a TCP/IP CTRACE with the VTAM option and contact the IBM Software Support Center.

Source Data Set: TCPIP

Procedure Name: EZBIFIND

EZD0003I SETTING VLAN ID NOT SUPPORTED FOR INTERFACE *interface_name*

Explanation: TCP/IP could not set the VLAN ID for this interface. The OSA-Express adapter microcode level does not support setting the VLAN ID.

interface_name is the name of the interface.

System Action: TCP/IP continues with interface activation but does not use a VLAN ID.

User or Operator Response: Inform the system programmer about the error.

System Programmer Response: Install a level of OSA-Express microcode that supports the VLAN ID function. Use the VTAM DISPLAY TRL command to determine the OSA-Express microcode level. For more information about the VTAM DISPLAY TRL command, refer to *z/OS Communications Server: SNA Operation*.

Source Data Set: TCPIP

Procedure Name: EZBIFIND

EZD0004I ERROR SETTING VLAN ID FOR INTERFACE *interface_name*

Explanation: An unexpected error occurred while setting the VLAN ID for the interface.

interface_name is the name of the interface.

System Action: TCP/IP continues with interface activation but does not use a VLAN ID.

User or Operator Response: Inform the system programmer about the error.

System Programmer Response: Obtain a TCP/IP CTRACE with the VTAM option and contact the IBM Software Support Center.

Source Data Set: TCPIP

Procedure Name: EZBIFIND

EZD0005I SETTING VLAN USER PRIORITY NOT SUPPORTED FOR INTERFACE *interface_name*

Explanation: TCP/IP cannot use VLAN priority for this interface. The OSA-Express adapter microcode level does not support the VLAN priority function.

interface_name is the name of the interface.

System Action: TCP/IP continues with interface activation but does not use VLAN priority.

User or Operator Response: Inform the system programmer about the error.

System Programmer Response: Install a level of OSA-Express microcode that supports the VLAN priority function. Use the VTAM DISPLAY TRL command to determine the OSA-Express microcode level. For more information about the VTAM DISPLAY TRL command, refer to *z/OS Communications Server: SNA Operation*.

Source Data Set: TCPIP

Procedure Name: EZBIFIND

EZD0006I ERROR SETTING VLAN USER PRIORITY FOR INTERFACE *interface_name*

Explanation: An unexpected error occurred while setting the VLAN priority for the interface.

interface_name is the name of the interface.

System Action: TCP/IP continues with interface activation but does not use VLAN priority.

User or Operator Response: Inform the system programmer about the error.

System Programmer Response: Obtain a TCP/IP CTRACE with the VTAM option and contact the IBM Software Support Center.

Source Data Set: TCPIP

Procedure Name: EZBIFIND

EZD0007I CONNECTION TO *addr* CLEARED FOR INTERFACE *Interface_Name*

Explanation: TCPIP was notified that the MPCPTP6 connection to the specified IPv6 address for the specified interface is no longer active.

addr is the IPv6 address of the remote node.

Interface_Name is the name of the interface.

System Action: The specified connection is no longer available for use with TCPIP. TCPIP will attempt to recover the connection.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: EZBIFIUM

Procedure Name: CM_CLEAR_IND_ATMMPC

**EZD0008I TIMEOUT DURING MPCPTP OR MPCPTP6 CONTROL PACKET EXCHANGE —
DEACTIVATING** *dev_or_interface_name*

Explanation: The remote node is not responding during the MPCPTP or MPCPTP6 control packet exchange.

dev_or_interface_name is the name of the device or interface.

System Action: The IPv6 Interface or IPv4 Link on which the timeout occurred is deactivated.

User or Operator Response: Contact the system programmer.

System Programmer Response: Obtain a CTRACE with OPTIONS=VTAM,VTAMDATA (on both sides, if the remote node is an IBM z/OS host) and contact the IBM Software Support Center. If the remote node is not an IBM z/OS host, the IBM Software Support Center might instruct you to contact the service group for the remote node.

Source Data Set: EZBIFOUT

Procedure Name: Process_MPC_Rxmt_Queues

EZD0009I CONNECTION TO *addr* ACTIVE FOR INTERFACE *interface_name*

Explanation: The MPCPTP6 connection to the specified IPv6 address for the specified interface is now active.

addr is the IPv6 address of the remote node.

interface_name is the name of the interface.

System Action: The specified connection is now available for use by TCP/IP.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: EZBIFIUM

Procedure Name: DM_ACT_IND_MPC

EZD0010I ERROR: CODE= *Error_Code* DURING *Link_Control_Function* CONNECTION TO *addr* FOR INTERFACE *Interface_Name* DIAGNOSTIC CODE: *Internal_Diagnostic_Code*

Explanation: The Link Layer has detected an error during activation of a virtual connection (VC) for an MPCPTP6 interface.

Error_Code is the Data Link Control (DLC) Status Code for the link layer.

Link_Control_Function is the function being performed on the VC.

addr is the IPv6 address of the remote node.

Interface_Name is the name of the interface.

Internal_Diagnostic_Code is an internal diagnostic code for use by IBM.

System Action: TCP/IP does not activate the VC.

User or Operator Response: Consult the section on Data Link Control (DLC) Status Codes in *z/OS Communications Server: IP and SNA Codes* for a description of the status code for the link layer. If a hardware problem is indicated, correct the hardware problem and restart the MPCPTP6 interface.

System Programmer Response: Perform the action described in the *z/OS Communications Server: IP and SNA Codes* for the indicated status code.

Source Data Set: EZBIFIUT

Procedure Name: IUT_MESSAGE

EZD0011I USE OF SITE LOCAL ADDRESS *ipv6_address* IS NOT RECOMMENDED

Explanation: A site local IPv6 address was added to TCPIP as a home IP address either through manual configuration (in the TCPIP profile) or through autoconfiguration (from a received router advertisement). Because the Internet Engineering Task Force (IETF) is revisiting the concept of site local addresses, it is recommended that you do not use site local IPv6 addresses for z/OS Communications Server IP. Depending on the outcome of the IETF work, z/OS Communications Server IP might drop support for these addresses in a future release.

System Action: TCP/IP accepts the site local address as a home IP address.

User or Operator Response: Contact system programmer.

System Programmer Response: Consider changing the configuration to not use site local IPv6 addresses.

Source Data Set: EZBTIINI

| **Procedure Name:** EZB6PRTE

EZD0101I NETSTAT *versionRelease*

Explanation: This message displays the current version and release for the command being displayed in the LONG format. The message is followed by the output for the requested command report. For a detailed description of the report, see the TSO NETSTAT section of *z/OS Communications Server: IP System Administrator's Commands*.

System Action: The Display Netstat command continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: EZACDNE6

Procedure Name: procACCN6(), procALLC6(), procARP(), procBYTE6(), procCACH6(), procCNFG6(), procCONN6(), procDEVL6(), procDVCF6(), procHOME6(), procIDS6(), procND6(), procPORT6(), procROUT6(), procSOCK6(), procSTAT6(), procVCRT6(), procVDPT6(), procVIPA6()

EZD0800I *proc_name* IP ADDRESS *ip_address* IS NOT VALID FOR USE IN ENTERPRISE EXTENDER

Explanation: The IP address is not valid. The IP address was not defined to the TCP/IP stack or was defined as a loopback address.

proc_name is the name of the started task associated with the TCP/IP address space.

ip_address is the Internet protocol address defined in SNA for the Enterprise Extender connection.

System Action: The Enterprise Extender connection using this IP address as the local address fails.

User or Operator Response: None.

System Programmer Response: Ensure that the IP address for Enterprise Extender defined in SNA matches a static VIPA address defined in this TCP/IP stack. The IP address is defined in SNA using the IPADDR or HOSTNAME start options or the IPADDR or HOSTNAME GROUP operands in the XCA major node. For more information, refer to *z/OS Communications Server: SNA Resource Definition Reference*.

Source Data Set: EZBUDBYP

Procedure Name: OpenReq

EZD0801I *proc_name* IP ADDRESS *ip_address* IS NOT A VALID STATIC VIPA ADDRESS

Explanation: The local IP address requested for an Enterprise Extender connection is not defined as one of the static VIPA addresses in this TCP/IP stack.

proc_name is the name of the started task associated with the TCP/IP address

space. ip_address is the Internet protocol address defined in SNA for the Enterprise Extender connection.

System Action: The Enterprise Extender connection using this IP address as the local address fails.

User or Operator Response: None.

System Programmer Response: Ensure that the Enterprise Extender definitions defined in SNA match the static VIPA definitions in your TCP/IP profile. The IP address is defined in SNA using the IPADDR or HOSTNAME start options or the IPADDR or HOSTNAME GROUP operands in the XCA major node. For more information, refer to *z/OS Communications Server: SNA Resource Definition Reference*.

Source Data Set: EZBUDBYP

Procedure Name: OpenReq

Chapter 10. EZD1xxxx messages

EZD1161I DCAS CONFIGURATION SERVERTYPE IS UNDEFINED

Explanation: The combination of SERVERTYPE keyword definitions in the DCAS configuration file caused the SERVERTYPE keyword to be undefined to DCAS.

System Action: DCAS ends.

User or Operator Response: Contact the system programmer.

System Programmer Response: Refer to *z/OS Communications Server: IP Configuration Reference* for a list of the SERVERTYPE keyword values in the DCAS configuration file.

Source Data Set: dcasconf.c

Procedure Name: process_config_keyword()

EZD1162I DCAS CONFIGURATION SERVERTYPE *value* IS NOT SUPPORTED

Explanation: This message is issued when DCAS is processing the DCAS configuration file.

value is the SERVERTYPE keyword that is not supported.

System Action: DCAS ends.

User or Operator Response: Contact the system programmer.

System Programmer Response: Refer to *z/OS Communications Server: IP Configuration Reference* for a list of the SERVERTYPE keyword values in the DCAS configuration file.

Source Data Set: dcasconf.c

Procedure Name: process_config_keyword()

EZD1163I IPV6 DYNAMIC XCF INTERFACE *xcfinterface_name* WAS CREATED BUT THE SPECIFIED SOURCEVIPAINTERFACE *interface_name* WILL NOT BE USED

Explanation: The interface name specified by the SOURCEVIPAINTERFACE parameter on the IPCONFIG6 DYNAMICXCF statement was not an active or valid interface name. The interface name must be an active static VIPA interface name.

xcfinterface_name is the Dynamic XCF interface name that was created.

interface_name is the name specified on the SOURCEVIPAINTERFACE parameter for IPCONFIG6 DYNAMICXCF.

System Action: The dynamic XCF interface is created. TCP/IP continues.

User or Operator Response: Contact the system programmer.

System Programmer Response: Create a static VIPA with an interface name that matches the one specified on the IPCONFIG6 DYNAMICXCF SOURCEVIPAINTERFACE statement.

Source Data Set: EZBXFDYN

Procedure Name: GenDevLnk

EZD1164I IPV6 TCPSTACKSOURCEVIPA INTERFACE *interface_name* WAS NOT USED BY *tcp_jobname*

Explanation: An outbound connection request was processed but the interface configured with TCPSTACKSOURCEVIPA could not be used to determine the source IPv6 address because the interface was not an active static VIPA interface or an active dynamic VIPA interface.

interface_name is the name specified on the TCPSTACKSOURCEVIPA parameter of the IPCONFIG6 statement.

tcp_jobname is the name of the job associated with the procedure that was used to start TCP/IP.

EZD1165I • EZD1166E

| **System Action:** TCP/IP continues. To avoid flooding the system console, this informational message will not be issued again for at least five minutes.

| **User or Operator Response:** Contact the system programmer.

| **System Programmer Response:** Change the TCPSTACKSOURCEVIPA interface name to be an active static or dynamic VIPA interface.

| **Source Data Set:** EZBX6UTL

| **Procedure Name:** EZBX6SSV

EZD1165I DVIPA INTERFACE *interface_name* IS ALREADY DEFINED WITH *ip_addr*

| **Explanation:** The TCPIP detected that the interface *interface_name* has already been defined as a DVIPA interface with a conflicting definition.

| For a VIPADEFINE/VIPABACKUP configuration statement, the interface has already been defined either by a previous VIPADEFINE/VIPABACKUP with a different IP address, or by a SIOCSVIPA IOCTL or BIND to an IP address within a configured VIPARANGE.

| *interface_name* is the interface name specified on the VIPADEFINE/VIPABACKUP statement.

| *ip_addr* is the IP address specified on the previous VIPADEFINE/VIPABACKUP statement.

| **System Action:** Processing continues. The VIPABACKUP or VIPADEFINE statement is rejected.

| **User or Operator Response:** Contact the system programmer.

| **System Programmer Response:** Correct the VIPABACKUP or VIPADEFINE statement with a different IP address or with a different interface name.

| **Source Data Set:** EZBX6DVI

| **Procedure Name:** PreValidateVBKUP6, PreValidateVDEF6

EZD1166E *tcpstackname* DELAYING SYSPLEX PROFILE PROCESSING - *application* IS NOT ACTIVE

| **Explanation:** The TCP/IP stack delayed joining the sysplex group and delayed processing sysplex definitions within the profile (VIPADYNAMIC and IPCONFIG/IPCONFIG6 DYNAMICXCF statements).

| *tcpstackname* is the name of the TCP/IP stack.

| *application* is the name of the application that is not active and is either OMPROUTE or VTAM.

| **System Action:** TCP/IP continues.

| **User or Operator Response:** If *application* is VTAM, start VTAM.

| If *application* is OMPROUTE, start OMPROUTE. This message will only be issued for OMPROUTE if GLOBALCONFIG SYSPLEXMONITOR DELAYJOIN was specified in a profile.

| If you want the TCP/IP stack to immediately join the sysplex group rather than waiting for OMPROUTE to activate, issue the VARY TCPIP,,OBEYFILE command with GLOBALCONFIG SYSPLEXMONITOR NODELAYJOIN specified. See the *z/OS Communications Server: IP Configuration Reference* for information about the DELAYJOIN keyword.

| When OMPROUTE activation is complete or a VARY TCPIP,,OBEYFILE command with GLOBALCONFIG SYSPLEXMONITOR NODELAYJOIN has been specified, TCP/IP will join the sysplex group and finish processing the sysplex definitions.

| **System Programmer Response:** If VTAM or OMPROUTE cannot be started, contact the IBM Software Support Center with the system log.

| **Source Data Set:** EZBXFPDM

| **Procedure Name:** EZBXFPDM

EZD1167I DVIPA INTERFACE *interface_name* IS CONFIGURED FROM A VIPARANGE

Explanation: The interface name specified on a VIPADELETE was defined on a VIPARANGE statement. The interface cannot be deleted by using a VIPADELETE statement.

interface_name is the interface name specified on the VIPADELETE statement.

System Action: Processing continues. The VIPADELETE statement is rejected.

User or Operator Response: Contact the system programmer.

System Programmer Response: Change the VIPADELETE to specify a valid interface name, or use a VIPARANGE DELETE (instead of a VIPADELETE) to delete the configured VIPARANGE statement.

Source Data Set: EZBX6DVI

Procedure Name: ValidateVDEL6

EZD1168I VIPADISTRIBUTE WITH THE PORT KEYWORD REJECTED FOR INTERFACE *interface_name*

Explanation: The PORT keyword was specified on a VIPADISTRIBUTE statement for a Dynamic VIPA (DVIPA) that already had a VIPADISTRIBUTE statement specified without a PORT keyword, indicating dynamic ports.

interface_name is the interface name specified on the rejected VIPADISTRIBUTE statement with the PORT keyword.

System Action: Processing continues. The VIPADISTRIBUTE statement is rejected.

User or Operator Response: Contact the System Programmer.

System Programmer Response: To disable dynamic ports, delete all previous VIPADISTRIBUTE statements for this DVIPA. Then reissue the VIPADISTRIBUTE with the PORT keyword.

Source Data Set: EZBX6DVI

Procedure Name: ValidateVDIST6

EZD1169I VIPADISTRIBUTE WITHOUT THE PORT KEYWORD REJECTED FOR INTERFACE *interface_name*

Explanation: The PORT keyword was not specified on a VIPADISTRIBUTE statement indicating dynamic ports for a Dynamic VIPA (DVIPA) that already had a VIPADISTRIBUTE statement specified with a PORT keyword.

interface_name is the interface name specified on the rejected VIPADISTRIBUTE statement without the PORT keyword.

System Action: Processing continues. The VIPADISTRIBUTE statement is rejected.

User or Operator Response: Contact the System Programmer.

System Programmer Response: To enable dynamic ports, delete all previous VIPADISTRIBUTE statements for this DVIPA. Then reissue the VIPADISTRIBUTE without the PORT keyword.

Source Data Set: EZBX6DVI

Procedure Name: ValidateVDIST6

EZD1170E *tcpstackname* WAS NOT ABLE TO GET TCP/IP *storagetype* STORAGE

Explanation: TCP/IP was not able to satisfy a request for storage.

tcpstackname is the name of the TCP/IP stack.

storagetype is the type of storage that was unavailable. The value for *storagetype* is either private or ECSA (Extended Common Storage Area)

System Action: TCP/IP continues.

- If the GLOBALCONFIG SYSPLEXMONITOR RECOVERY option is active and this stack is not the only member of the TCP/IP sysplex group, the following RECOVERY actions will occur:
 - This stack will leave the sysplex group.
 - All DVIPAs defined on this stack will be deleted.
 - This stack will no longer participate in sysplex distribution (as a distributor, target, or backup for DVIPAs).

EZD1215I

- | • If the GLOBALCONFIG SYSPLEXMONITOR NORECOVERY option is active, no action will be taken.
- | Regardless of the RECOVERY/NORECOVERY setting, if the problem is corrected, this operator message will be deleted. See the *z/OS Communications Server: IP Configuration Guide* for more information about Sysplex problem detection and recovery. See the *z/OS Communications Server: IP Configuration Reference* for the definition of RECOVERY and NORECOVERY in the GLOBALCONFIG statement.
- | **User or Operator Response:** Save the TCP/IP profile and system log. If a dump was not created, then take a dump of the TCP/IP address space and dataspace.
- | **System Programmer Response:** Use the display TCP/IP storage command D TCPIP,,STOR to determine the current status of TCP/IP private and ECSA storage. See the *z/OS Communications Server: IP System Administrator's Commands* for more information about the D TCPIP,,STOR command.
- | If storage limits were previously set using the GLOBALCONFIG ECSALIMIT or GLOBALCONFIG POOLLIMIT statement, it might be necessary to raise these limits to correct the problem; look for earlier TCP/IP warning messages concerning these limits. See the *z/OS Communications Server: IP Configuration Reference* for more information about using ECSALIMIT and POOLLIMIT on the GLOBALCONFIG statement.
- | For ECSA storage exhaustion, determine which jobs or address spaces are using an excessive amount of storage. To determine the users of ECSA storage, common storage tracking should be enabled. Refer to the *z/OS MVS Initialization and Tuning Guide* for information about requesting common storage tracking. Use the IPCS VERBEXIT VSMDATA OWNCOMM SUMMARY command to determine how much storage is used by each job. Refer to the *z/OS MVS Diagnosis: Tools and Service Aids* for information about the IPCS VERBEXIT VSMDATA command.
- | If the storage problem cannot be corrected, contact the IBM System Support Center with the documentation taken when the problem occurred.
- | If the storage problem can be corrected:
 - | • If RECOVERY is active, enable the stack to rejoin the sysplex group. After message EZZ9676E is issued, reapply the sysplex profile definitions by issuing the VARY TCPIP,,OBEYFILE command. This will cause the stack to rejoin the sysplex group.
 - | • If NORECOVERY is active, no further actions are needed.
- | **Source Data Set:** EZBXFPDC
- | **Procedure Name:** EZBXFPDC

| EZD1215I MLSCHK STARTED STACK *tcpjobname* USER *stkuser* SECLABEL *stksl*

- | **Explanation:** This message is written the first time the multilevel security consistency check is performed on a stack. It is also written each time the stack determines that the system has returned to MLACTIVE after being changed to NOMLACTIVE. When running on an MLACTIVE system, TCPIP performs a multilevel security consistency check in any of the following situations:
 - | • After initial profile processing.
 - | • After every VARY TCPIP,,OBEYFILE command.
 - | • After SYSPLEX dynamic VIPA changes.
 - | • When it receives an ENF signal from RACF indicating a RACLIST REFRESH was done for the SERVAUTH or SECLABEL class.
- | *tcpjobname* is the name of the TCPIP job.
- | *stkuser* is the user ID of this TCPIP job.
- | *stksl* is the security label of this TCPIP job, or <NONE> if the job was started without a security label.
- | **System Action:** Processing continues.
- | **User or Operator Response:** None.
- | **System Programmer Response:** None. See the *z/OS Communications Server: IP Diagnosis Guide* for information about how to set the user and seclabel values.
- | **Source Data Set:** EZBTIMDF

| **Procedure Name:** MLSCheck

| **EZD1216I MLSCHK SUCCEEDED STACK *tcpjobname* IS MULTILEVEL SECURE**

| **Explanation:** When running on an MACTIVE system, TCPIP performs a multilevel security consistency check in any of the following situations:

- | • After initial profile processing.
- | • After every VARY TCPIP,,OBEYFILE command.
- | • After SYSPLEX dynamic VIPA changes.
- | • When it receives an ENF signal from RACF indicating a RACLIST REFRESH was done for the SERVAUTH or SECLABEL class.

| No configuration inconsistencies were found. This message is issued the first time multilevel security checking completes successfully after the stack is started or the first time it completes successfully after a failure.

| *tcpjobname* is the job TCPIP JOBNAME.

| **System Action:** Processing continues.

| **User or Operator Response:** None.

| **System Programmer Response:** None.

| **Source Data Set:** EZBTIMDF

| **Procedure Name:** MLSCheck

| **EZD1217I MLSCHK FAILED STACK *tcpjobname* - *count* MESSAGES WRITTEN TO JOBLOG**

| **Explanation:** When running on an MACTIVE system, TCPIP performs a multilevel security consistency check in any of the following situations:

- | • After initial profile processing.
- | • After every VARY TCPIP,,OBEYFILE command.
- | • After SYSPLEX dynamic VIPA changes.
- | • When it receives an ENF signal from RACF indicating a RACLIST REFRESH was done for the SERVAUTH or SECLABEL class.

| At least one configuration inconsistency was found and reported in a message.

| *tcpjobname* is the TCPIP JOBNAME.

| *count* is the number of problem messages written to the job log.

| **System Action:** The GLOBALCONFIG [NO]MLSCHKTERMINATE setting in TCPIP PROFILE determines the action, as follows:

- | • MLSCHKTERMINATE indicates that this stack will end because of the reported errors.
- | • NOMLSCHKTERMINATE indicates that this stack will not end even though it cannot correctly enforce multilevel security policies.

| **User or Operator Response:** Save the job log and contact the System Programmer.

| **System Programmer Response:** Review the MLSCHK messages in the job log. Correct the indicated problems in TCPIP PROFILE or security server profiles. See the *z/OS Communications Server: IP Diagnosis Guide* for more information about diagnosing TCPIP access control problems.

| **Source Data Set:** EZBTIMDF

| **Procedure Name:** MLSCheck

EZD1218I **MLSCHK ENDED STACK** *tcpjobname*

Explanation: This message is written when a stack that has been performing multilevel security consistency checks determines that the system is now NOMLACTIVE. This stack has discontinued checking TCP/IP and security server configurations for multilevel security consistency.

tcpjobname is the name of the TCPIP job.

System Action: Processing continues.

User or Operator Response: None.

System Programmer Response: None.

Source Data Set: EZBTIMDF

Procedure Name: MLSCheck

EZD1219I **MLSCHK NETACCESS NOT CONFIGURED WHEN MLACTIVE**

Explanation: The TCPIP PROFILE for this stack does not contain a valid NETACCESS statement when running in a multilevel secure environment. The stack does not enforce multilevel security policies when NETACCESS is not configured. The NETACCESS statement must specify both INBound and OUTBound and must contain at least one valid security zone definition.

System Action: TCPIP continues.

User or Operator Response: Contact the System Programmer.

System Programmer Response: Add a NETACCESS INBound OUTBound statement with at least one network security zone to the PROFILE for this stack.

Source Data Set: EZBTIMDF

Procedure Name: MLSCheck

EZD1220I **MLSCHK REQUIRED SERVAUTH PROFILE NOT FOUND RESNM** *resourcename*

Explanation: In a multilevel secure environment, a required security product profile could not be found.

resourcename is the name of the resource that must be covered by a profile in the SERVAUTH class.

System Action: TCPIP continues.

User or Operator Response: Contact the System Programmer.

System Programmer Response: Define a security product profile that covers the resource name in the SERVAUTH class. Verify that the SERVAUTH class is active. If changes have been made, issue RACF command SETROPTS RACLIST(SERVAUTH) REFRESH or equivalent security product command. See the *z/OS Communications Server: IP Configuration Guide* for information about the syntax of *resourcename*.

Source Data Set: EZBTIMDF

Procedure Name: MLSCheck

EZD1221I **MLSCHK STACKACCESS PROFILE HAS WRONG SECLABEL** *profsl* **RESNM** *resourcename* **PRFNM** *profilename*

Explanation: While TCPIP is running in a multilevel secure environment, the STACKACCESS profile must have the same security label as the TCPIP job. The TCPIP job security label is displayed in message EZD1215I. The security label must remain defined and active for the duration of the TCPIP job. If any changes are made to either the seclabel class or the servauth class, that class must be refreshed.

profsl is the security label defined in the SECLABEL field in this SERVAUTH profile. If the SECLABEL field is blank, *profsl* will be <NONE>.

resourcename is the name of the STACKACCESS resource.

profilename is the SERVAUTH profile found for this resource.

| **System Action:** TCPIP continues.

| **User or Operator Response:** Contact the System Programmer.

| **System Programmer Response:** Verify that the security label is defined, and active on this system. If necessary, alter the SECLABEL on the profile in the SERVAUTH class to the same security label as the TCPIP job. You might need to define a less generic profile that covers this resource name in the SERVAUTH class. If any changes are made to either the seclabel class or the servauth class, that class must be refreshed. Issue RACF command SETROPTS RACLIST(class name) REFRESH. See the *z/OS Communications Server: IP Diagnosis Guide* for more information about diagnosing TCPIP access control problems.

| **Source Data Set:** EZBTIMDF

| **Procedure Name:** MLSCheck

| **EZD1222I** **MLSCHK LOCAL INTERFACE IS NOT IN A NETWORK SECURITY ZONE** *ipaddress* **IF**
| *ifcname*

| **Explanation:** In a multilevel secure environment, all interface addresses must be configured into a NETACCESS security zone.

| *ipaddress* is the IP address on the local interface.

| *ifcname* is the name of the INTERFACE or IPv4 LINK statement that defined this IP address.

| **System Action:** TCPIP continues.

| **User or Operator Response:** Contact the System Programmer.

| **System Programmer Response:** Verify that this IP address is intended to be defined to this TCPIP stack. If so, modify the NETACCESS statement so that it maps the IP address into the correct security zone. See the *z/OS Communications Server: IP Configuration Guide* for information about configuring Network Access Control.

| **Source Data Set:** EZBTIMDF

| **Procedure Name:** MLSCheck

| **EZD1223I** **MLSCHK LOCAL ZONE HAS INCORRECT SECLABEL** *zonesl* **IPADR** *ipaddress* **IF** *ifcname* **ZONE**
| *zonename zoneentry RESNM resourcename PRFNM profilename*

| **Explanation:** In a multilevel secure environment, an interface is configured into a NETACCESS security zone with an incorrect security label for the TCPIP job that owns the interface. See message EZD1215I for the security label of the TCPIP job. Local interface addresses must be in zones with the same security label as the TCPIP job. VIPAs on an unrestricted stack may be in a security zone with any security label that is currently active on that system. VIPAs on a restricted stack must be in zones with a security label that is not SYSMULTI and is equivalent to the TCPIP job.

| *zonesl* is the security label defined in the SECLABEL field in this SERVAUTH profile. If the SECLABEL field is blank, *zonesl* will be <NONE>.

| *ipaddress* is the IPv4 or IPv6 address on the local interface.

| *ifcname* is the name of the INTERFACE or IPv4 LINK statement that defined this IP address.

| *zonename* is the security zone name defined in the NETACCESS zone entry for this IP address.

| *zoneentry* is the NETACCESS zone entry for this IP address. *zoneentry* will one of the following:

- | • DEFAULT
- | • DEFAULTHOME
- | • *ipaddress/masklength*

| *resourcename* is the name of the SERVAUTH resource.

| *profilename* is the name of the SERVAUTH profile that covers this resource name.

| **System Action:** TCPIP continues.

| **User or Operator Response:** Contact the System Programmer.

| **System Programmer Response:** Verify that:

- | • The IP address is intended to be owned by this TCPIP JOB.
 - | • The NETACCESS statement maps the IP address into the correct security zone.
 - | • The resource name is covered by the intended profile in the SERVAUTH class.
 - | • The profile has the correct security label defined.
 - | • The security label is defined and active on this system.
- | See the *z/OS Communications Server: IP Diagnosis Guide* for more information about diagnosing TCPIP access control problems.
- | **Source Data Set:** EZBTIMDF
- | **Procedure Name:** MLSCheck
-

| **EZD1224I MLSCHK STACK SECLABEL *stksl* IS NOT VALID**

| **Explanation:** When running on an MACTIVE system, the TCPIP job must run under a user ID with a security label. The security label must remain defined and active while TCPIP is running.

| *stksl* is the security label of this TCPIP job, or <NONE> if the job was started without a security label.

| **System Action:** TCPIP continues.

| **User or Operator Response:** Contact the System Programmer.

| **System Programmer Response:** Verify that:

- | • The stack is running under the correct user ID.
- | • The USER profile has the correct default seclabel.
- | • The security label is defined and active on this system.
- | • The user ID has READ authority to the security label.

| If the user ID or user security label are incorrect, stop TCPIP, correct the problem and start it again.

| **Source Data Set:** EZBTIMDF

| **Procedure Name:** MLSCheck

| **EZD1225I MLSCHK *iptype ipaddr* IS NOT IN A NETWORK SECURITY ZONE**

| **Explanation:** In a multilevel secure environment, *ipaddr* must be configured into a NETACCESS security zone with a security label the same as the TCPIP job.

| *iptype* is either INADDR_ANY, IN6ADDR_ANY or TCPSTACKSOURCEVIPA.

| *ipaddr* is the associated IP address.

| **System Action:** TCPIP continues.

| **User or Operator Response:** Contact the System Programmer.

| **System Programmer Response:** If the *iptype* is TCPSTACKSOURCEVIPA, verify that this IP address is intended to be used by this TCPIP JOB. Modify the NETACCESS statement so that it maps the IP address into the correct security zone.

| **Source Data Set:** EZBTIMDF

| **Procedure Name:** MLSCheck

| **EZD1226I MLSCHK *iptype ipaddr* HAS INCORRECT SECLABEL *zonesl* ZONE *zonename zoneentry* RESNM
| *resourcenam* PRFNM *profilename***

| **Explanation:** In a multilevel secure environment, *ipaddr* must be configured into a NETACCESS security zone with a security label the same as the TCPIP job. See message EZD1215I for the security label of the TCPIP job.

| *iptype* is one of INADDR_ANY, IN6ADDR_ANY or TCPSTACKSOURCEVIPA.

| *ipaddr* is the associated IP address.

| *zonesl* is the security label defined in the SECLABEL field in this SERVAUTH profile. If the SECLABEL field is blank, *zonesl* will be <NONE>.

| *zonename* is the security zone name defined in the NETACCESS zone entry for this IP address.

| *zoneentry* is the NETACCESS zone entry for this IP address. *zoneentry* is either:

- | • DEFAULT
- | • DEFAULTHOME
- | • ipaddress/masklength.

| *resourcename* is the name of the SERVAUTH resource.

| *profilename* is the name of the SERVAUTH profile that covers this resource name.

| **System Action:** TCPIP continues.

| **User or Operator Response:** Contact the System Programmer.

| **System Programmer Response:** If the *iptype* is TCPSTACKSOURCEVIPA, verify that this IP address is intended to be used by this TCPIP job. Verify that:

- | • The NETACCESS statement maps the IP address into the correct security zone.
- | • The resource name is covered by the intended profile in the SERVAUTH class.
- | • The profile has the correct security label defined.
- | • The security label is defined and active on this system.

| See the *z/OS Communications Server: IP Diagnosis Guide* for more information about diagnosing TCPIP access control problems.

| **Source Data Set:** EZBTIMDF

| **Procedure Name:** MLSCheck

EZD1227I MLSCHK NETACCESS *value* NOT CONFIGURED WHEN MLACTIVE

| **Explanation:** The TCPIP PROFILE for this stack does not contain a valid NETACCESS statement when running in a multilevel secure environment. The stack does not fully enforce multilevel security policies when NETACCESS is not completely configured. The NETACCESS statement must specify both INBound and OUTBound and must contain at least one valid security zone definition.

| *value* is one of the following:

- | • INBOUND to indicate that the NETACCESS statement INBound parameter is missing.
- | • OUTBOUND to indicate that the NETACCESS statement OUTBound parameter is missing.
- | • SECURITY ZONE to indicate that the NETACCESS statement does not contain any zone entries.

| **System Action:** TCPIP continues.

| **User or Operator Response:** Contact the System Programmer.

| **System Programmer Response:** Add a NETACCESS INBound OUTBound statement with at least one network security zone to the PROFILE for this stack.

| **Source Data Set:** EZBTIMDF

| **Procedure Name:** MLSCheck

EZD1228I MLSCHK INTERFACE *ifcname* AUTOCONFIGURED WHEN MLACTIVE

| **Explanation:** In a multilevel secure environment, an IPv6 interface is found to be incorrectly configured to allow IP addresses to be dynamically autoconfigured. In a multilevel secure network all IP addresses must be manually configured so they can be correctly configured into NETACCESS security zones. All IPv6 INTERFACE statements must specify INTFID to complete the Link Local address. For all IPv6 interface types that support autoconfiguration, the INTERFACE statement must specify at least one IPADDR to disable autoconfiguration. If IPCONFIG6 DYNAMICXCF is configured, it must also specify INTFID to complete the Link Local address.

- | *ifcname* is the name of the INTERFACE statement that incorrectly allows autoconfigured addresses.
- | **System Action:** Processing continues.
- | **User or Operator Response:** Contact the system programmer.
- | **System Programmer Response:** Verify the following:
 - | • The named INTERFACE statement specifies a manual interface ID with the INTFID parameter.
 - | • If IPCONFIG6 DYNAMICXCF is configured, the statement also specifies a manual interface ID with the INTFID parameter.
 - | • If the named INTERFACE statement is of a type that supports dynamic address autoconfiguration, the statement also specifies at least one manually configured prefix or IP address with the IPADDR parameter.
- | See the *z/OS Communications Server: IP Configuration Guide* for more information about IPv6 dynamic address autoconfiguration.
- | **Source Data Set:** EZBTIMDF
- | **Procedure Name:** MLSCheck

Appendix A. Message ranges and the components that issue them

Volume 1 ranges

Number Span	Component
EZA0519 - EZA0661	TCP/IP Utilities
EZA1450 - EZA2947	FTP
EZA2950 - EZA3172	Network Computing
EZA3950 - EZA4173	Network Database
EZA4250 - EZA4257	Portmapper
EZA4275 - EZA4361	Remote Procedure Call
EZA4375 - EZA4432	REXEC/D
EZA4726 - EZA4809	REXEC
EZA4986 - EZA5049	RSH
EZA5125 - EZA5651	SMTP
EZA5775 - EZA6230	SNALINK & SNALU6.2
EZA6275 - EZA6635	SQESERVE
EZA8200 - EZA8417	Telnet
EZA8500 - EZA9744	Common SayCalRe
EZAIN11	TCP/IP

Volume 2 ranges

Number Span	Component
EZB0600 - EZB0857	LPD
EZB0900 - EZB1100	LPT
EZB1200 - EZB1230	X Window
EZB2000 - EZB2498	XNX25
EZB2500 - EZB2785	Retired — do not use
EZB3000 - EZB3566	DNS
EZB3825 - EZB4198	NCPROUTE
EZB6473	Initialization
EZB8801	FFST™
EZD0001 - EZD0050	Interface Layer
EZD0101 - EZD0799	Netstat
EZD0800 - EZD0801	Enterprise Extender
EZD0810 - EZD1160	IPSEC
EZD1161 - EZD1162	DCAS configuration file
EZD1163 - EZD1214	DVIPA

Number Span	Component
EZD1215 - EZD1230	Multilevel Security Enhancements

Volume 3 ranges

Number Span	Component
EZY0002 - EZY0058	Packet Trace
EZY0620 - EZY0974	Network Print facility
EZY1020 - EZY1035	IMS Socket
EZY1120 - EZY1130	Sockets Extended
EZY1218 - EZY1352	CICS®
EZY1372 - EZY1424	REXX Socket
EZY1870 - EZY2145	MVS Platform
EZY2370 - EZY2443	MISCSRV
EZY2632 - EZY2832	FTP
EZY3720 - EZY4199	Sockets
EZY4200 - EZY4484	Telnet DBCS
EZY5398	MVS Platform
EZY6001 - EZY6077	Restartable VMCF
EZY6101 - EZY6261	FFS Writer for JES
EZYFS01 - EZYFS95 EZYFT01 - EZYFT99	FTP
EZYPR001 - EZYPR214	Network Print Facility ISPF
EZYRC01 - EZYRC24	OE REXEC
EZYRD01 - EZYRD39	OE REXECD
EZYRP35 - EZYRP72 (EZA4331 - EZA4335)	OE RPC
EZYRS01 - EZYRS66	OE RSHD
EZYTE01 - EZYTE87 EZYTO01 - EZYTO12 EZYTS02 - EZYTS18 EZYTU01 - EZYTU36 EZYTY01 - EZYTY14	OS/390 UNIX Telnet
EZYXM01 - EZYXR42	OE OSF/Motif API
EZYXU01 - EZYXU99	OE OSF/Motif User Interface Language Compiler
EZYXW01 - EZYXW69	OE x Window

Volume 4 ranges

Number Span	Component
EZZ0053 - EZZ0720	Configuration & Initialization
EZZ2351 - EZZ3089	ONETSTAT & NETSTAT
EZZ3105 - EZZ3133	OPING
EZZ3200 - EZZ3249	SNMP Subagent
EZZ3250 - EZZ3255	Operator Command
EZZ3300 - EZZ3344	OSNMP

Number Span	Component
EZZ4200 - EZZ4247	Initialization
EZZ4300 - EZZ4349	Interface Layer
EZZ4350	Initialization
EZZ4360 - EZZ4367	Storage Utilization
EZZ4820 - EZZ5027	OE ROUTED
EZZ6002 - EZZ6084	Telnet
EZZ6201 - EZZ6319	SNMP Agent
EZZ6351 - EZZ6388	OTRACERT
EZZ6452 - EZZ6705	DNS
EZZ7001 - EZZ7050	TFTP
EZZ7051 - EZZ7062	TIMED
EZZ7251 - EZZ7331	DHCP
EZZ7350 - EZZ7448	NSM
EZZ7450 - EZZ7452	Initialization
EZZ7475 - EZZ7478	Stack Configuration
EZZ7500 - EZZ7738	Sendmail & Popper
EZZ7800 - EZZ8111	OMPROUTE
EZZ8201 - EZZ8247	SLA Subagent
EZZ8252 - EZZ8260	D TCP/IP,,SYSPLEX
EZZ8301 - EZZ8319	VIPA
EZZ8320 - EZZ8331	HOSTNAME, DOMAINNAME, DNSDOMAINNAME, HOST
EZZ8340 - EZZ8345	Misc. Command
EZZ8392 - EZZ8423	Trap Forwarder Daemon
EZZ8431 - EZZ8451	Service Policy Agent
EZZ8453 - EZZ8460	Storage Display
EZZ8461 - EZZ8477	Sysplex Distributor
EZZ8495- EZZ8503	Intrusion Detection Services (SYSLOG)
EZZ8504 - EZZ8511	Intrusion Detection Services Statistics (TRMDSTAT)
EZZ8540	Intrusion Detection Services (SYSLOG)
EZZ8600 - EZZ8635	DCAS
EZZ8638 - EZZ8659	Intrusion Detection Services SYSLOG
EZZ8761 - EZZ8770	Intrusion Detection Services Console
EZZ8771 - EZZ8778	Policy Agent (PAGENT) Messages
EZZ8791 - EZZ9290	DNS BIND 9
EZZ9291 - EZZ9304	System Resolver

Number Span	Component
EZZ9315	Initialization
EZZ9316 - EZZ9327	Intrusion Detection Services SYSLOG
EZZ9331 - EZZ9599	DNS BIND 9
EZZ9600 - EZZ9602	SNTP
EZZ9650 - EZZ9669	VIPA
EZZ9670 - EZZ9679	Sysplex Distributor
EZZ9680 - EZZ9724	DNS Bind 9
EZZ9780 - EZZ9784	Neighbor Discovery
EZZ9790 - EZZ9817	FTP
EZZ9891 - EZZ9993	Sendmail
SNM000 - SNM129	SNMPLIB

Appendix B. Related protocol specifications (RFCs)

This appendix lists the related protocol specifications for TCP/IP. The Internet Protocol suite is still evolving through requests for comments (RFC). New protocols are being designed and implemented by researchers and are brought to the attention of the Internet community in the form of RFCs. Some of these protocols are so useful that they become recommended protocols. That is, all future implementations for TCP/IP are recommended to implement these particular functions or protocols. These become the *de facto* standards, on which the TCP/IP protocol suite is built.

You can request RFCs through electronic mail, from the automated Network Information Center (NIC) mail server, by sending a message to `service@nic.ddn.mil` with a subject line of RFC *nnnn* for text versions or a subject line of RFC *nnnn*.PS for PostScript versions. To request a copy of the RFC index, send a message with a subject line of RFC INDEX.

For more information, contact `nic@nic.ddn.mil` or at:

Government Systems, Inc.
Attn: Network Information Center
14200 Park Meadow Drive
Suite 200
Chantilly, VA 22021

Hard copies of all RFCs are available from the NIC, either individually or by subscription. Online copies are available at the following Web address:
<http://www.rfc-editor.org/rfc.html>.

See "Internet Drafts" on page 280 for draft RFCs implemented in this and previous Communications Server releases.

Many features of TCP/IP Services are based on the following RFCs:

RFC	Title and Author
768	<i>User Datagram Protocol</i> J. Postel
791	<i>Internet Protocol</i> J. Postel
792	<i>Internet Control Message Protocol</i> J. Postel
793	<i>Transmission Control Protocol</i> J. Postel
821	<i>Simple Mail Transfer Protocol</i> J. Postel
822	<i>Standard for the Format of ARPA Internet Text Messages</i> D. Crocker
823	<i>DARPA Internet Gateway</i> R. Hinden, A. Sheltzer
826	<i>Ethernet Address Resolution Protocol or Converting Network Protocol Addresses to 48.Bit Ethernet Address for Transmission on Ethernet Hardware</i> D. Plummer
854	<i>Telnet Protocol Specification</i> J. Postel, J. Reynolds
855	<i>Telnet Option Specification</i> J. Postel, J. Reynolds
856	<i>Telnet Binary Transmission</i> J. Postel, J. Reynolds

- 857 *Telnet Echo Option* J. Postel, J. Reynolds
- 858 *Telnet Suppress Go Ahead Option* J. Postel, J. Reynolds
- 859 *Telnet Status Option* J. Postel, J. Reynolds
- 860 *Telnet Timing Mark Option* J. Postel, J. Reynolds
- 861 *Telnet Extended Options—List Option* J. Postel, J. Reynolds
- 862 *Echo Protocol* J. Postel
- 863 *Discard Protocol* J. Postel
- 864 *Character Generator Protocol* J. Postel
- 877 *Standard for the Transmission of IP Datagrams over Public Data Networks* J. Korb
- 885 *Telnet End of Record Option* J. Postel
- 894 *Standard for the transmission of IP datgrams over Ethernet networks* C. Hornig
- 896 *Congestion Control in IP/TCP Internetworks* J. Nagle
- 903 *Reverse Address Resolution Protocol* R. Finlayson, T. Mann, J. Mogul, M. Theimer
- 904 *Exterior Gateway Protocol Formal Specification* D. Mills
- 919 *Broadcasting Internet Datagrams* J. Mogul
- 922 *Broadcasting Internet Datagrams in the Presence of Subnets* J. Mogul
- 950 *Internet Standard Subnetting Procedure* J. Mogul, J. Postel
- 951 *Bootstrap Protocol* W.J. Croft, J. Gilmore
- 952 *DoD Internet Host Table Specification* K. Harrenstien, M. Stahl, E. Feinler
- 959 *File Transfer Protocol* J. Postel, J. Reynolds
- 974 *Mail Routing and the Domain Name System* C. Partridge
- 1001 *Protocol Standard for a NetBIOS service on a TCP/UDP transport: Concepts and ;methods* NetBios Working Group in the Defense Advanced Research Projects Agency, Internet Activities Board, End-to-End Services Task Force
- 1002 *Protocol Standard for a NetBIOS service on a TCP/UDP transport: Detailed Specifications* NetBios Working Group in the Defense Advanced Research Projects Agency, Internet Activities Board, End-to-End Services Task Force
- 1006 *ISO Transport Service on top of the TCP Version 3* M. Rose, D. Cass
- 1009 *Requirements for Internet Gateways* R. Braden, J. Postel
- 1011 *Official Internet Protocols* J. Reynolds, J. Postel
- 1013 *X Window System Protocol, Version 11: Alpha Update* R. Scheifler
- 1014 *XDR: External Data Representation Standard* Sun Microsystems Incorporated
- 1027 *Using ARP to Implement Transparent Subnet Gateways* S. Carl-Mitchell, J. Quarterman
- 1032 *Domain Administrators Guide* M. Stahl
- 1033 *Domain Administrators Operations Guide* M. Lottor
- 1034 *Domain Names—Concepts and Facilities* P. Mockapetris
- 1035 *Domain Names—Implementation and Specification* P. Mockapetris

- 1042 *Standard for the Transmission of IP Datagrams over IEEE 802 Networks* J. Postel, J. Reynolds
- 1044 *Internet Protocol on Network System's HYPERchannel: Protocol Specification* K. Hardwick, J. Lekashman
- 1055 *Nonstandard for Transmission of IP Datagrams over Serial Lines: SLIP* J. Romkey
- 1057 *RPC: Remote Procedure Call Protocol Version 2 Specification* Sun Microsystems Incorporated
- 1058 *Routing Information Protocol* C. Hedrick
- 1060 *Assigned Numbers* J. Reynolds, J. Postel
- 1073 *Telnet Window Size Option* D. Waitzman
- 1079 *Telnet Terminal Speed Option* C. Hedrick
- 1091 *Telnet Terminal-Type Option* J. VanBokkelen
- 1094 *NFS: Network File System Protocol Specification* Sun Microsystems Incorporated
- 1096 *Telnet X Display Location Option* G. Marcy
- 1101 *DNS encoding of network names and other types* P. Mockapetris
- 1112 *Host Extensions for IP Multicasting* S. Deering
- 1118 *Hitchhikers Guide to the Internet* E. Krol
- 1122 *Requirements for Internet Hosts—Communication Layers* R. Braden
- 1123 *Requirements for Internet Hosts—Application and Support* R. Braden
- 1155 *Structure and Identification of Management Information for TCP/IP-Based Internets* M. Rose, K. McCloghrie
- 1156 *Management Information Base for Network Management of TCP/IP-Based Internets* K. McCloghrie, M. Rose
- 1157 *Simple Network Management Protocol (SNMP)* J. Case, M. Fedor, M. Schoffstall, C. Davin
- 1158 *Management Information Base for Network Management of TCP/IP-based internets: MIB-II* M. Rose
- 1179 *Line Printer Daemon Protocol* The Wollongong Group, L. McLaughlin III
- 1180 *TCP/IP Tutorial* T. Socolofsky, C. Kale
- 1183 *New DNS RR Definitions* C. Everhart, L. Mamakos, R. Ullmann, P. Mockapetris, (Updates RFC 1034, RFC 1035)
- 1184 *Telnet Linemode Option* D. Borman
- 1187 *Bulk Table Retrieval with the SNMP* M. Rose, K. McCloghrie, J. Davin
- 1188 *Proposed Standard for the Transmission of IP Datagrams over FDDI Networks* D. Katz
- 1191 *Path MTU Discovery* J. Mogul, S. Deering
- 1198 *FYI on the X Window System* R. Scheifler
- 1207 *FYI on Questions and Answers: Answers to Commonly Asked "Experienced Internet User" Questions* G. Malkin, A. Marine, J. Reynolds

- 1208 *Glossary of Networking Terms* O. Jacobsen, D.Lynch
- 1213 *Management Information Base for Network Management of TCP/IP-Based Internets: MIB-II* K. McCloghrie, M. Rose
- 1215 *Convention for Defining Traps for Use with the SNMP* M. Rose
- 1228 *SNMP-DPI Simple Network Management Protocol Distributed Program Interface* G.C. Carpenter, B. Wijnen
- 1229 *Extensions to the Generic-Interface MIB* K. McCloghrie
- 1230 *IEEE 802.4 Token Bus MIB* K. McCloghrie, R. Fox
- 1231 *IEEE 802.5 Token Ring MIB* K. McCloghrie, R. Fox, E. Decker
- 1236 *IP to X.121 Address Mapping for DDN* L. Morales, P. Hasse
- 1267 *A Border Gateway Protocol 3 (BGP-3)* K. Lougheed, Y. Rekhter
- 1268 *Application of the Border Gateway Protocol in the Internet* Y. Rekhter, P. Gross
- 1269 *Definitions of Managed Objects for the Border Gateway Protocol (Version 3)* S. Willis, J. Burruss
- 1270 *SNMP Communications Services* F. Kastenholz, ed.
- 1321 *The MD5 Message-Digest Algorithm* R. Rivest
- 1323 *TCP Extensions for High Performance* V. Jacobson, R. Braden, D. Borman
- 1325 *FYI on Questions and Answers: Answers to Commonly Asked "New Internet User" Questions* G. Malkin, A. Marine
- 1340 *Assigned Numbers* J. Reynolds, J. Postel
- 1348 *DNS NSAP RRs* B. Manning
- 1349 *Type of Service in the Internet Protocol Suite* P. Almquist
- 1350 *TFTP Protocol* K.R. Sollins
- 1351 *SNMP Administrative Model* J. Davin, J. Galvin, K. McCloghrie
- 1352 *SNMP Security Protocols* J. Galvin, K. McCloghrie, J. Davin
- 1353 *Definitions of Managed Objects for Administration of SNMP Parties* K. McCloghrie, J. Davin, J. Galvin
- 1354 *IP Forwarding Table MIB* F. Baker
- 1356 *Multiprotocol Interconnect on X.25 and ISDN in the Packet Mode* A. Malis, D. Robinson, R. Ullmann
- 1363 *A Proposed Flow Specification* C. Partridge
- 1372 *Telnet Remote Flow Control Option* D. Borman, C. L. Hedrick
- 1374 *IP and ARP on HIPPI* J. Renwick, A. Nicholson
- 1381 *SNMP MIB Extension for X.25 LAPB* D. Throop, F. Baker
- 1382 *SNMP MIB Extension for the X.25 Packet Layer* D. Throop
- 1387 *RIP Version 2 Protocol Analysis* G. Malkin
- 1388 *RIP Version 2—Carrying Additional Information* G. Malkin
- 1389 *RIP Version 2 MIB Extension* G. Malkin
- 1390 *Transmission of IP and ARP over FDDI Networks* D. Katz

- 1393 *Traceroute Using an IP Option* G. Malkin
- 1397 *Default Route Advertisement In BGP2 And BGP3 Versions of the Border Gateway Protocol* D. Haskin
- 1398 *Definitions of Managed Objects for the Ethernet-Like Interface Types* F. Kastenholz
- 1408 *Telnet Environment Option* D.Borman, Ed.
- 1416 *Telnet Authentication Option* D. Borman, ed.
- 1464 *Using the Domain Name System to Store Arbitrary String Attributes* R. Rosenbaum
- 1469 *IP Multicast over Token-Ring Local Area Networks* T. Pusateri
- 1497 *BOOTP Vendor Information Extensions* J.Reynolds
- 1533 *DHCP Options and BOOTP Vendor Extensions* S.Alexander, R.Droms
- 1534 *Interoperation Between DHCP and BOOTP* R.Droms
- 1535 *A Security Problem and Proposed Correction With Widely Deployed DNS Software* E. Gavron
- 1536 *Common DNS Implementation Errors and Suggested Fixes* A. Kumar, J. Postel, C. Neuman, P. Danzig, S.Miller
- 1537 *Common DNS Data File Configuration Errors* P. Beertema
- 1540 *IAB Official Protocol Standards* J. Postel
- 1541 *Dynamic Host Configuration Protocol* R.Droms
- 1542 *Clarifications and Extensions for the Bootstrap Protocol* W.Wimer
- 1571 *Telnet Environment Option Interoperability Issues* D. Borman
- 1572 *Telnet Environment Option* S. Alexander
- 1577 *Classical IP and ARP over ATM* M. Laubach
- 1583 *OSPF Version 2* J. Moy
- 1591 *Domain Name System Structure and Delegation* J. Postel
- 1592 *Simple Network Management Protocol Distributed Protocol Interface Version 2.0* B. Wijnen, G. Carpenter, K. Curran, A. Sehgal, G. Waters
- 1594 *FYI on Questions and Answers: Answers to Commonly Asked "New Internet User" Questions* A. Marine, J. Reynolds, G. Malkin
- 1646 *TN3270 Extensions for LUname and Printer Selection* C.Graves, T.Butts, M.Angel
- 1647 *TN3270 Enhancement* B.Kelly
- 1695 *Definitions of Managed Objects for ATM Management Version 8.0 Using SMIPv2* M. Ahmed, K. Tesink
- 1706 *DNS NSAP Resource Records* B. Manning, R. Colella
- 1713 *Tools for DNS debugging* A. Romao
- 1723 *RIP Version 2—Carrying Additional Information* G. Malkin
- 1766 *Tags for the Identification of Languages* H. Alvestrand
- 1794 *DNS Support for Load Balancing* T. Brisco

- 1832 *XDR: External Data Representation Standard* R. Srinivasan
- 1840 *Schema Publishing in X500 Directory* G.Mansfield, P.Rajeev, S.Raghavan, T.Howes
- 1850 *OSPF Version 2 Management Information Base* F. Baker, R. Coltun
- 1876 *A Means for Expressing Location Information in the Domain Name System* C. Davis, P. Vixie, T. Goodwin, I. Dickinson
- 1886 *DNS Extensions to support IP version 6* S. Thomson, C. Huitema
- 1901 *Introduction to Community-Based SNMPv2* J. Case, K. McCloghrie, M. Rose, S. Waldbusser
- 1902 *Structure of Management Information for Version 2 of the Simple Network Management Protocol (SNMPv2)* J. Case, K. McCloghrie, M. Rose, S. Waldbusser
- 1903 *Textual Conventions for Version 2 of the Simple Network Management Protocol (SNMPv2)* J. Case, K. McCloghrie, M. Rose, S. Waldbusser
- 1904 *Conformance Statements for Version 2 of the Simple Network Management Protocol (SNMPv2)* J. Case, K. McCloghrie, M. Rose, S. Waldbusser
- 1905 *Protocols Operations for Version 2 of the Simple Network Management Protocol (SNMPv2)* J. Case, K. McCloghrie, M. Rose, S. Waldbusser
- 1906 *Transport Mappings for Version 2 of the Simple Network Management Protocol (SNMPv2)* J. Case, K. McCloghrie, M. Rose, S. Waldbusser
- 1907 *Management Information Base for Version 2 of the Simple Network Management Protocol (SNMPv2)* J. Case, K. McCloghrie, M. Rose, S. Waldbusser
- 1908 *Coexistence between Version 1 and Version 2 of the Internet-Standard Network Management Framework* J. Case, K. McCloghrie, M. Rose, S. Waldbusser
- 1912 *Common DNS Operational and Configuration Errors* D. Barr
- 1918 *Address Allocation for Private Internets* Y. Rekhter, B. Moskowitz, D. Karrenberg, G.J. de Groot, E. Lear
- 1928 *SOCKS Protocol Version 5* M. Leech, M. Ganis, Y. Lee, R. Kuris, D. Koblas, L. Jones
- 1939 *Post Office Protocol-Version 3* J. Myers, M. Rose
- 1981 *Path MTU Discovery for IP version 6* J. McCann, S. Deering, J. Mogul
- 1982 *Serial Number Arithmetic* R. Elz, R. Bush
- 1995 *Incremental Zone Transfer in DNS* M. Ohta
- 1996 *A Mechanism for Prompt Notification of Zone Changes (DNS NOTIFY)* P. Vixie
- 2010 *Operational Criteria for Root Name Servers* B. Manning, P. Vixie
- 2011 *SNMPv2 Management Information Base for the Internet Protocol Using SMIv2* K. McCloghrie
- 2012 *SNMPv2 Management Information Base for the Transmission Control Protocol Using SMIv2* K. McCloghrie
- 2013 *SNMPv2 Management Information Base for the User Datagram Protocol Using SMIv2* K. McCloghrie
- 2030 *Simple Network Time Protocol* D. Mills

- 2052 *A DNS RR for specifying the location of services (DNS SRV)* A. Gulbrandsen, P. Vixie
- 2065 *Domain Name System Security Extensions* D. Eastlake, C. Kaufman
- 2080 *RIPng for IPv6* G. Malkin, R. Minnear
- 2096 *IP Forwarding Table MIB* F. Baker
- 2104 *HMAC: Keyed-Hashing for Message Authentication* H. Krawczyk, M. Bellare, R. Canetti
- 2132 *DHCP Options and BOOTP Vendor Extensions* S. Alexander, R. Droms
- 2133 *Basic Socket Interface Extensions for IPv6* R. Gilligan, S. Thomson, J. Bound, W. Stevens
- 2136 *Dynamic Updates in the Domain Name System (DNS UPDATE)* P.Vixie, Ed.,S.Thomson, Y.Rekhter,J.Bound
- 2137 *Secure Domain Name System Dynamic Update* D. Eastlake
- 2163 *Using the Internet DNS to Distribute MIXER Conformant Global Address Mapping (MCGAM)* C. Allocchio
- 2168 *Resolution of Uniform Resource Identifiers using the Domain Name System* R. Daniel, M. Mealling
- 2178 *OSPF Version 2* J. Moy
- 2181 *Clarifications to the DNS Specification* R. Elz, R. Bush
- 2205 *Resource ReSerVation Protocol (RSVP) Version 1* R. Braden, L. Zhang, S. Berson, S. Herzog, S. Jamin
- 2210 *The Use of RSVP with IETF Integrated Services* J. Wroclawski
- 2211 *Specification of the Controlled-Load Network Element Service* J. Wroclawski
- 2212 *Specification of Guaranteed Quality of Service* S. Shenker, C. Partridge, R. Guerin
- 2215 *General Characterization Parameters for Integrated Service Network Elements* S. Shenker, J. Wroclawski
- 2219 *Use of DNS Aliases for Network Services* M. Hamilton, R. Wright
- 2228 *FTP Security Extensions* M. Horowitz, S. Lunt
- 2230 *Key Exchange Delegation Record for the DNS* R. Atkinson
- 2233 *The Interfaces Group MIB Using SMIV2* K. McCloghrie, F. Kastenholz
- 2240 *A Legal Basis for Domain Name Allocation* O. Vaughn
- 2246 *The TLS Protocol Version 1.0* T. Dierks, C. Allen
- 2251 *Lightweight Directory Access Protocol (v3)* M.Wahl,T.Howes, S.Kille
- 2308 *Negative Caching of DNS Queries (DNS NCACHE)* M. Andrews
- 2317 *Classless IN-ADDR.ARPA delegation* H. Eidnes, G. de Groot, P. Vixie
- 2320 *Definitions of Managed Objects for Classical IP and ARP over ATM Using SMIV2* M. Greene, J. Luciani, K. White, T. Kuo
- 2328 *OSPF Version 2* J. Moy
- 2345 *Domain Names and Company Name Retrieval* J. Klensin, T. Wolf, G. Oglesby
- 2352 *A Convention for Using Legal Names as Domain Names* O. Vaughn

- 2355 *TN3270 Enhancements* B. Kelly
- 2373 *IP Version 6 Addressing Architecture* R. Hinden, M. O'Dell, S. Deering
- 2374 *An IPv6 Aggregatable Global Unicast Address Format* R. Hinden, M. O'Dell, S. Deering
- 2375 *IPv6 Multicast Address Assignments* R. Hinden, S. Deering
- 2389 *Feature negotiation mechanism for the File Transfer Protocol* P. Hethmon, R. Elz
- 2401 *Security Architecture for Internet Protocol* S.Kent, R. Atkinson
- 2402 *IP Authentication Header* S.Kent, R. Atkinson
- 2403 *HMAC-MD5-96 within ESP and AH* Madson, R. Glenn
- 2404 *The Use of HMAC-MD5-96 within ESP and AH* C. Madson, R. Glenn
- 2405 *The ESP DES-CBC Cipher Algorithm With Explicit IVC* Madson, N. Doraswamy
- 2406 *IP Encapsulating Security Payload (ESP)* S.Kent, R. Atkinson
- 2409 *The Internet Key Exchange (IKE)* D. Harkins, D. Carrel
- 2410 *The NULL Encryption Algorithm and Its Use With IPsec* R. Glenn, S. Kent,
- 2428 *FTP Extensions for IPv6 and NATs* M. Allman, S. Ostermann, C. Metz
- 2460 *Internet Protocol, Version 6 (IPv6) Specification* S. Deering, R. Hinden
- 2461 *Neighbor Discovery for IP Version 6 (IPv6)* T. Narten, E. Nordmark, W. Simpson
- 2462 *IPv6 Stateless Address Autoconfiguration* S. Thomson, T. Narten
- 2464 *Transmission of IPv6 Packets over Ethernet Networks* M. Crawford
- 2474 *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers* K. Nichols, S. Blake, F. Baker, D. Black
- 2487 *SMTP Service Extension for Secure SMTP over TLS* P. Hoffman
- 2505 *Anti-Spam Recommendations for SMTP MTAs* G. Lindberg
- 2535 *Domain Name System Security Extensions* D. Eastlake
- 2539 *Storage of Diffie-Hellman Keys in the Domain Name System (DNS)* D. Eastlake
- 2570 *Introduction to Version 3 of the Internet-standard Network Management Framework* J. Case, R. Mundy, D. Partain, B. Stewart
- 2571 *An Architecture for Describing SNMP Management Frameworks* D. Harrington, R. Presuhn, B. Wijnen
- 2572 *Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)* J. Case, D. Harrington, R. Presuhn, B. Wijnen
- 2573 *SNMP Applications* D. Levi, P. Meyer, B. Stewart
- 2574 *User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)* U. Blumenthal, B. Wijnen
- 2575 *View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)* B. Wijnen, R. Presuhn, K. McCloghrie
- 2576 *Co-Existence between Version 1, Version 2 and Version 3 of the Internet-standard Network Management Framework* R. Frye, D. Levi, S. Routhier, B. Wijnen

- 2578 *Structure of Management Information Version 2 (SMIv2)* K. McCloghrie, D. Perkins, J. Schoenwaelder
- 2635 *Don't SPEW A Set of Guidelines for Mass Unsolicited Mailings and Postings (spam*)* S.Hambridge, A.Lunde
- 2640 *Internationalization of the File Transfer Protocol* B. Curtin
- 2665 *Definitions of Managed Objects for the Ethernet-like Interface Types* J. Flick, J. Johnson
- 2672 *Non-Terminal DNS Name Redirection* M. Crawford
- 2710 *Multicast Listener Discovery (MLD) for IPv6* S. Deering, W. Fenner, B. Haberman
- 2711 *IPv6 Router Alert Option* C. Partridge, A. Jackson
- 2740 *OSPF for IPv6* R. Coltun, D. Ferguson, J. Moy
- 2758 *Definitions of Managed Objects for Service Level Agreements Performance Monitoring* K. White
- 2845 *Secret Key Transaction Authentication for DNS (TSIG)* P. Vixie, O. Gudmundsson, D. Eastlake, B. Wellington
- 2874 *DNS Extensions to Support IPv6 Address Aggregation and Renumbering* M. Crawford, C. Huitema
- 2941 *Telnet Authentication Option* T. Ts'o, ed., J. Altman
- 2942 *Telnet Authentication: Kerberos Version 5* T. Ts'o
- 2946 *Telnet Data Encryption Option* T. Ts'o
- 2952 *Telnet Encryption: DES 64 bit Cipher Feedback* T. Ts'o
- 2953 *Telnet Encryption: DES 64 bit Output Feedback* T. Ts'o, ed.
- 3060 *Policy Core Information Model—Version 1 Specification* B. Moore, E. Ellesson, J. Strassner, A. Westerinen
- 3363 *Representing Internet Protocol version 6 (IPv6) Addresses in the Domain Name System* R.Bush, A.Durand, B.Fink, O.Gudmundsson,T.Hain
- 3390 *Increasing TCP's Initial Window* M. Allman, S. Floyd, C. Partridge
- 3411 *An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks* D.Harrington, R.Presuhn, B.Wijnen
- 3412 *Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)* J.Case, D.Harrington, R.Presuhn, B.Wijnen
- 3413 *Simple Network Management Protocol (SNMP) Applications* D.Levi, P. Meyer, B.Stewart
- 3414 *User- Based Security Model (USM) for version 3 of the Simple Network management Protocol (SNMPv3)* U. Blumenthal, B. Wijnen
- 3415 *View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)* B.Wijnen, R.Presuhn, K.McCloghrie
- 3484 *Default Address Selection for Internet Protocol version 6 (IPv6)* R. Draves
- 3493 *Basic Socket Interface Extensions for IPv6* R. Gilligan, S. Thomson, J. Bound, J. McCann, W. Stevens
- 3513 *Internet Protocol Version 6 (IPv6) Addressing Architecture* R. Hinden, S. Deering

Internet Drafts

Internet drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Other groups may also distribute working documents as Internet drafts. You can see Internet drafts at <http://www.ietf.org/ID.html>.

Several areas of IPv6 implementation include elements of the following Internet drafts and are subject to change during the RFC review process.

Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification

A. Conta, S. Deering

Appendix C. Information APARs

This appendix lists information APARs for IP and SNA documents.

Notes:

1. Information APARs contain updates to previous editions of the manuals listed below. Documents updated for V1R6 are complete except for the updates contained in the information APARs that might be issued after V1R6 documents went to press.
2. Information APARs are predefined for z/OS V1R6 Communications Server and might not contain updates.
3. Information APARs for z/OS documents are in the document called *z/OS and z/OS.e DOC APAR and PTF ++HOLD Documentation*, which can be found at http://publibz.boulder.ibm.com:80/cgi-bin/bookmgr_OS390/BOOKS/ZIDOCMST/CCONTENTS.

Information APARs for IP documents

Table 2 lists information APARs for IP documents.

Table 2. IP information APARs for z/OS Communications Server

Title	V1R6	V1R5	V1R4	V1R2
New Function Summary (both IP and SNA)	II13824			
Quick Reference (both IP and SNA)	II13831		II13246	II12500
IP and SNA Codes	II13842		II13254	II12504
IP API Guide	II13844	II13577	II13255 II13790	II12861 II13655
IP CICS Sockets Guide		II13578	II13257	II12862 II13627
IP Configuration Guide	II13826	II13568	II13244 II13541 II13652 II13646	II12498 II13087 II13364 II13634 II13651
IP Configuration Reference	II13827	II13569 II13789	II13245 II13521 II13647 II13739	II12499 II13394 II13637
IP Diagnosis	II13836	II13571	II13249 II13493	II12503 II13473
IP Messages Volume 1	II13838	II13572	II13624 II13250	II12857 II13229 II13405
IP Messages Volume 2	II13839	II13573	II13251	II12858
IP Messages Volume 3	II13840	II13574	II13252	II12859
IP Messages Volume 4	II13841	II13575	II13253 II13628	II12860

Table 2. IP information APARs for z/OS Communications Server (continued)

Title	V1R6	V1R5	V1R4	V1R2
IP Migration		II13566	II13242 II13738	II12497 II13636
IP Network and Application Design Guide	II13825	II13567	II13243	
IP Network Print Facility				II12864
IP Programmer's Reference	II13843	II13581	II13256	II12505
IP User's Guide and Commands	II13832	II13570	II13247	II12501 II13404
IP System Admin Commands	II13833	II13580	II13248 II13792	II12502 II13793

Information APARs for SNA documents

Table 3 lists information APARs for SNA documents.

Table 3. SNA information APARs for z/OS Communications Server

Title	V1R6	V1R5	V1R4	V1R2
New Function Summary (both IP and SNA)	II13824			
Quick Reference (both IP and SNA)	II13831		II13246	II12500
IP and SNA Codes	II13842		II13254	II12504
SNA Customization	II13857	II13560	II13240	II12872
SNA Diagnosis		II13558	II13236 II13735	II12490 II13034
SNA Diagnosis, Vol. 1: Techniques and Procedures	II13852			
SNA Diagnosis, Vol. 2: FFST Dumps and the VIT	II13853			
SNA Messages	II13854	II13559	II13238 II13736	II12491
SNA Network Implementation Guide	II13849	II13555	II13234 II13733	II13635 II12487
SNA Operation	II13851	II13557	II13237	II12489
SNA Migration		II13554	II13233 II13732	II12486
SNA Programming	II13858		II13241	II13033
SNA Resource Definition Reference	II13850	II13556	II13235 II13734	II12488
SNA Data Areas, Vol. 1 and 2			II13239	II12492
SNA Data Areas, 1	II13855			
SNA Data Areas, 2	II13856			

Other information APARs

Table 4 lists information APARs not related to documents.

Table 4. Non-document information APARs

Content	Number
index of recommended maintenance for VTAM	II11220
index of Communication Server IP information APARs	II12028
AHHC, MPC, and CTC	II01501
Collecting TCPIP CTRACES	II12014
CSM for VTAM	II12657
CSM for TCP/IP	II12658
DLUR/DLUS for z/OS V1R2	II12986
DOCUMENTATION REQUIRED FOR OSA/2, OSA EXPRESS AND OSA QDIO	II13016
DYNAMIC VIPA (BIND)	II13215
DNS — common problems and solutions	II13453
Enterprise Extender	II12223
FTPing doc to z/OS Support	II12030
FTP problems	II12079
Generic resources	II10986
HPR	II10953
iQDIO	II11220
LPR problems	II12022
MNPS	II10370
NCPROUTE problems	II12025
OMPROUTE	II12026
OROUTED problems	II12024
PASCAL API	II11814
Performance	II11710 II11711 II11712
Resolver	II13398 II13399 II13452
Socket API	II11996 II12020
SMTP problems	II12023
SNMP	II13477 II13478
SYSLOGD howto	II12021
TCPIP connection states	II12449
Telnet	II11574 II13135
TN3270 TELNET SSL common problems	II13369

Appendix D. Accessibility

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use software products successfully. The major accessibility features in z/OS enable users to:

- Use assistive technologies such as screen readers and screen magnifier software
- Operate specific or equivalent features using only the keyboard
- Customize display attributes such as color, contrast, and font size

Using assistive technologies

Assistive technology products, such as screen readers, function with the user interfaces found in z/OS. Consult the assistive technology documentation for specific information when using such products to access z/OS interfaces.

Keyboard navigation of the user interface

Users can access z/OS user interfaces using TSO/E or ISPF. Refer to *z/OS TSO/E Primer*, *z/OS TSO/E User's Guide*, and *z/OS ISPF User's Guide Volume I* for information about accessing TSO/E and ISPF interfaces. These guides describe how to use TSO/E and ISPF, including the use of keyboard shortcuts or function keys (PF keys). Each guide includes the default settings for the PF keys and explains how to modify their functions.

z/OS information

z/OS information is accessible using screen readers with the BookServer/Library Server versions of z/OS books in the Internet library at:

www.ibm.com/servers/eserver/zseries/zos/bkserv/

One exception is command syntax that is published in railroad track format; screen-readable copies of z/OS books with that syntax information are separately available in HTML zipped file form upon request to usib2hpd@vnet.ibm.com.

Notices

IBM may not offer all of the products, services, or features discussed in this document. 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 in 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:

Site Counsel
IBM Corporation
P.O. Box 12195
3039 Cornwallis Road
Research Triangle Park, North Carolina 27709-2195
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 information 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 of 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.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

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 the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. 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's application programming interfaces.

Each copy or any portion of these sample programs or any derivative work must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

IBM is required to include the following statements in order to distribute portions of this document and the software described herein to which contributions have been made by The University of California. Portions herein © Copyright 1979, 1980, 1983, 1986, Regents of the University of California. Reproduced by permission. Portions herein were developed at the Electrical Engineering and Computer Sciences Department at the Berkeley campus of the University of California under the auspices of the Regents of the University of California.

Portions of this publication relating to RPC are Copyright © Sun Microsystems, Inc., 1988, 1989.

Some portions of this publication relating to X Window System** are Copyright © 1987, 1988 by Digital Equipment Corporation, Maynard, Massachusetts, and the Massachusetts Institute Of Technology, Cambridge, Massachusetts. All Rights Reserved.

Some portions of this publication relating to X Window System are Copyright © 1986, 1987, 1988 by Hewlett-Packard Corporation.

Permission to use, copy, modify, and distribute the M.I.T., Digital Equipment Corporation, and Hewlett-Packard Corporation portions of this software and its documentation for any purpose without fee is hereby granted, provided that the above copyright notice appears in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the names of M.I.T., Digital, and Hewlett-Packard not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. M.I.T., Digital, and Hewlett-Packard make no representation about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

Copyright © 1983, 1995-1997 Eric P. Allman

Copyright © 1988, 1993 The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:
This product includes software developed by the University of California, Berkeley and its contributors.
4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software program contains code, and/or derivatives or modifications of code originating from the software program "Popper." Popper is Copyright ©1989-1991 The Regents of the University of California, All Rights Reserved. Popper was created by Austin Shelton, Information Systems and Technology, University of California, Berkeley.

Permission from the Regents of the University of California to use, copy, modify, and distribute the "Popper" software contained herein for any purpose, without fee, and without a written agreement is hereby granted, provided that the above copyright notice and this paragraph and the following two paragraphs appear in all copies. HOWEVER, ADDITIONAL PERMISSIONS MAY BE NECESSARY FROM OTHER PERSONS OR ENTITIES, TO USE DERIVATIVES OR MODIFICATIONS OF POPPER.

IN NO EVENT SHALL THE UNIVERSITY OF CALIFORNIA BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, INCLUDING LOST PROFITS, ARISING OUT OF THE USE OF THE POPPER SOFTWARE, OR ITS DERIVATIVES OR MODIFICATIONS, AND ITS DOCUMENTATION, EVEN IF THE UNIVERSITY OF CALIFORNIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE UNIVERSITY OF CALIFORNIA SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE POPPER SOFTWARE PROVIDED HEREUNDER IS ON AN "AS IS" BASIS, AND THE UNIVERSITY OF CALIFORNIA HAS NO OBLIGATIONS TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

Copyright © 1983 The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms are permitted provided that the above copyright notice and this paragraph are duplicated in all such forms and that any documentation, advertising materials, and other materials related to such distribution and use acknowledge that the software was developed by the University of California, Berkeley. The name of the University may not be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED ``AS IS'' AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Copyright © 1991, 1993 The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:
This product includes software developed by the University of California, Berkeley and its contributors.
4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright © 1990 by the Massachusetts Institute of Technology

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of M.I.T. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Furthermore

if you modify this software you must label your software as modified software and not distribute it in such a fashion that it might be confused with the original M.I.T. software. M.I.T. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

Copyright © 1998 by the FundsXpress, INC. All rights reserved.

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of FundsXpress not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. FundsXpress makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Copyright © 1999, 2000 Internet Software Consortium.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND INTERNET SOFTWARE CONSORTIUM DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL INTERNET SOFTWARE CONSORTIUM BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright © 1995-1998 Eric Young (eay@cryptsoft.com) All rights reserved.

This package is an SSL implementation written by Eric Young (eay@cryptsoft.com). The implementation was written so as to conform with Netscape's SSL.

This library is free for commercial and non-commercial use as long as the following conditions are adhered to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@cryptsoft.com).

Copyright remains Eric Young's, and as such any Copyright notices in the code are not to be removed. If this package is used in a product, Eric Young should be

given attribution as the author of the parts of the library used. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement: "This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)". The word 'cryptographic' can be left out if the routines from the library being used are not cryptographic related.
4. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include acknowledgement:
"This product includes software written by Tim Hudson (tjh@cryptsoft.com)"

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The license and distribution terms for any publicly available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution license [including the GNU Public License.]

This product includes cryptographic software written by Eric Young.

Copyright © 1999, 2000 Internet Software Consortium.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND INTERNET SOFTWARE CONSORTIUM DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL INTERNET SOFTWARE CONSORTIUM BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright © 2004 IBM Corporation and its licensors, including Sendmail, Inc., and the Regents of the University of California. All rights reserved.

Copyright © 1999,2000,2001 Compaq Computer Corporation

Copyright © 1999,2000,2001 Hewlett-Packard Company

Copyright © 1999,2000,2001 IBM Corporation

Copyright © 1999,2000,2001 Hummingbird Communications Ltd.

Copyright © 1999,2000,2001 Silicon Graphics, Inc.

Copyright © 1999,2000,2001 Sun Microsystems, Inc.

Copyright © 1999,2000,2001 The Open Group

All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

X Window System is a trademark of The Open Group.

If you are viewing this information softcopy, photographs and color illustrations may not appear.

You can obtain softcopy from the z/OS Collection (SK3T-4269), which contains BookManager and PDF formats of unlicensed books and the z/OS Licensed Product Library (LK3T-4307), which contains BookManager and PDF formats of licensed books.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

Advanced Peer-to-Peer Networking	MVS/SP
AFP	MVS/XA
AD/Cycle	NetView
AIX	Network Station
AIX/ESA	Nways
AnyNet	Notes
APL2	OfficeVision/MVS
AS/400	OfficeVision/VM
AT	Open Class
BookManager	OpenEdition
BookMaster	OS/2
C/370	OS/390
CICS	OS/400
CICS/ESA	Parallel Sysplex
C/MVS	PR/SM
Common User Access	PROFS
C Set ++	PS/2
CT	RACF
CUA	Redbooks
DB2	Resource Link
DFSMSdfp	RETAIN
DFSMShsm	RISC System/6000
DFSMS/MVS	RMF
DPI	RS/6000
Domino	S/370
DRDA	S/390
Enterprise Systems Architecture/370	S/390 Parallel Enterprise Server
ESCON	SAA
eServer	SecureWay
ES/3090	SP
ES/9000	SP2
ES/9370	SQL/DS
EtherStreamer	System/360
Extended Services	System/370
FFST	System/390
FFST/2	SystemView
First Failure Support Technology	Tivoli
GDDM	TURBOWAYS
IBM	VM/ESA
IBMLink	VSE/ESA
IMS	VTAM
IMS/ESA	WebSphere
Java	XT
HiperSockets	z/Architecture
Language Environment	z/OS
LANStreamer	zSeries
Library Reader	z/VM
LPDA	400
Micro Channel	3090
Multiprise	3890
MVS	
MVS/DFP	
MVS/ESA	

DB2 and NetView are registered trademarks of International Business Machines Corporation or Tivoli Systems Inc. in the U.S., other countries, or both.

The following terms are trademarks of other companies:

ATM is a trademark of Adobe Systems, Incorporated.

BSC is a trademark of BusiSoft Corporation.

CSA is a trademark of Canadian Standards Association.

DCE is a trademark of The Open Software Foundation.

HYPERchannel is a trademark of Network Systems Corporation.

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

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

ActionMedia, LANDesk, MMX, Pentium, and ProShare are trademarks of Intel Corporation in the United States, other countries, or both. For a complete list of Intel trademarks, see <http://www.intel.com/sites/corporate/tradmarx.htm> .

Other company, product, and service names may be trademarks or service marks of others.

Bibliography

z/OS Communications Server information

This section contains descriptions of the documents in the z/OS Communications Server library.

z/OS Communications Server documentation is available:

- Online at the z/OS Internet Library web page at <http://www.ibm.com/servers/eserver/zseries/zos/bkserv>
- In softcopy on CD-ROM collections. See “Softcopy information” on page xi.

z/OS Communications Server library

z/OS Communications Server documents are available on the CD-ROM accompanying z/OS (SK3T-4269 or SK3T-4307). Unlicensed documents can be viewed at the z/OS Internet library site.

Updates to documents are available on RETAIN[®] and in information APARs (info APARs). See Appendix C, “Information APARs,” on page 281 for a list of the documents and the info APARs associated with them.

Info APARs for z/OS documents are in the document called *z/OS and z/OS.e DOC APAR and PTF ++HOLD Documentation* which can be found at http://publibz.boulder.ibm.com:80/cgi-bin/bookmgr_OS390/BOOKS/ZIDOCMST/CCONTENTS.

Planning

Title	Number	Description
<i>z/OS Communications Server: New Function Summary</i>	GC31-8771	This document is intended to help you plan for new IP for SNA function, whether you are migrating from a previous version or installing z/OS for the first time. It summarizes what is new in the release and identifies the suggested and required modifications needed to use the enhanced functions.
<i>z/OS Communications Server: IPv6 Network and Application Design Guide</i>	SC31-8885	This document is a high-level introduction to IPv6. It describes concepts of z/OS Communications Server's support of IPv6, coexistence with IPv4, and migration issues.

Resource definition, configuration, and tuning

Title	Number	Description
<i>z/OS Communications Server: IP Configuration Guide</i>	SC31-8775	This document describes the major concepts involved in understanding and configuring an IP network. Familiarity with the z/OS operating system, IP protocols, z/OS UNIX System Services, and IBM Time Sharing Option (TSO) is recommended. Use this document in conjunction with the <i>z/OS Communications Server: IP Configuration Reference</i> .

Title	Number	Description
<i>z/OS Communications Server: IP Configuration Reference</i>	SC31-8776	This document presents information for people who want to administer and maintain IP. Use this document in conjunction with the <i>z/OS Communications Server: IP Configuration Guide</i> . The information in this document includes: <ul style="list-style-type: none"> • TCP/IP configuration data sets • Configuration statements • Translation tables • SMF records • Protocol number and port assignments
<i>z/OS Communications Server: SNA Network Implementation Guide</i>	SC31-8777	This document presents the major concepts involved in implementing an SNA network. Use this document in conjunction with the <i>z/OS Communications Server: SNA Resource Definition Reference</i> .
<i>z/OS Communications Server: SNA Resource Definition Reference</i>	SC31-8778	This document describes each SNA definition statement, start option, and macroinstruction for user tables. It also describes NCP definition statements that affect SNA. Use this document in conjunction with the <i>z/OS Communications Server: SNA Network Implementation Guide</i> .
<i>z/OS Communications Server: SNA Resource Definition Samples</i>	SC31-8836	This document contains sample definitions to help you implement SNA functions in your networks, and includes sample major node definitions.
<i>z/OS Communications Server: AnyNet SNA over TCP/IP</i>	SC31-8832	This guide provides information to help you install, configure, use, and diagnose SNA over TCP/IP.
<i>z/OS Communications Server: AnyNet Sockets over SNA</i>	SC31-8831	This guide provides information to help you install, configure, use, and diagnose sockets over SNA. It also provides information to help you prepare application programs to use sockets over SNA.
<i>z/OS Communications Server: IP Network Print Facility</i>	SC31-8833	This document is for system programmers and network administrators who need to prepare their network to route SNA, JES2, or JES3 printer output to remote printers using TCP/IP Services.

Operation

Title	Number	Description
<i>z/OS Communications Server: IP User's Guide and Commands</i>	SC31-8780	This document describes how to use TCP/IP applications. It contains requests that allow a user to log on to a remote host using Telnet, transfer data sets using FTP, send and receive electronic mail, print on remote printers, and authenticate network users.
<i>z/OS Communications Server: IP System Administrator's Commands</i>	SC31-8781	This document describes the functions and commands helpful in configuring or monitoring your system. It contains system administrator's commands, such as TSO NETSTAT, PING, TRACERTE and their UNIX counterparts. It also includes TSO and MVS commands commonly used during the IP configuration process.
<i>z/OS Communications Server: SNA Operation</i>	SC31-8779	This document serves as a reference for programmers and operators requiring detailed information about specific operator commands.
<i>z/OS Communications Server: Quick Reference</i>	SX75-0124	This document contains essential information about SNA and IP commands.

Customization

Title	Number	Description
<i>z/OS Communications Server: SNA Customization</i>	LY43-0092	<p>This document enables you to customize SNA, and includes the following:</p> <ul style="list-style-type: none"> • Communication network management (CNM) routing table • Logon-interpret routine requirements • Logon manager installation-wide exit routine for the CLU search exit • TSO/SNA installation-wide exit routines • SNA installation-wide exit routines

Writing application programs

Title	Number	Description
<i>z/OS Communications Server: IP Application Programming Interface Guide</i>	SC31-8788	This document describes the syntax and semantics of program source code necessary to write your own application programming interface (API) into TCP/IP. You can use this interface as the communication base for writing your own client or server application. You can also use this document to adapt your existing applications to communicate with each other using sockets over TCP/IP.
<i>z/OS Communications Server: IP CICS Sockets Guide</i>	SC31-8807	This document is for programmers who want to set up, write application programs for, and diagnose problems with the socket interface for CICS using z/OS TCP/IP.
<i>z/OS Communications Server: IP IMS Sockets Guide</i>	SC31-8830	This document is for programmers who want application programs that use the IMS™ TCP/IP application development services provided by IBM's TCP/IP Services.
<i>z/OS Communications Server: IP Programmer's Reference</i>	SC31-8787	This document describes the syntax and semantics of a set of high-level application functions that you can use to program your own applications in a TCP/IP environment. These functions provide support for application facilities, such as user authentication, distributed databases, distributed processing, network management, and device sharing. Familiarity with the z/OS operating system, TCP/IP protocols, and IBM Time Sharing Option (TSO) is recommended.
<i>z/OS Communications Server: SNA Programming</i>	SC31-8829	This document describes how to use SNA macroinstructions to send data to and receive data from (1) a terminal in either the same or a different domain, or (2) another application program in either the same or a different domain.
<i>z/OS Communications Server: SNA Programmer's LU 6.2 Guide</i>	SC31-8811	This document describes how to use the SNA LU 6.2 application programming interface for host application programs. This document applies to programs that use only LU 6.2 sessions or that use LU 6.2 sessions along with other session types. (Only LU 6.2 sessions are covered in this document.)
<i>z/OS Communications Server: SNA Programmer's LU 6.2 Reference</i>	SC31-8810	This document provides reference material for the SNA LU 6.2 programming interface for host application programs.
<i>z/OS Communications Server: CSM Guide</i>	SC31-8808	This document describes how applications use the communications storage manager.

Title	Number	Description
<i>z/OS Communications Server: CMIP Services and Topology Agent Guide</i>	SC31-8828	This document describes the Common Management Information Protocol (CMIP) programming interface for application programmers to use in coding CMIP application programs. The document provides guide and reference information about CMIP services and the SNA topology agent.

Diagnosis

Title	Number	Description
<i>z/OS Communications Server: IP Diagnosis Guide</i>	GC31-8782	This document explains how to diagnose TCP/IP problems and how to determine whether a specific problem is in the TCP/IP product code. It explains how to gather information for and describe problems to the IBM Software Support Center.
<i>z/OS Communications Server: SNA Diagnosis Vol 1, Techniques and Procedures and z/OS Communications Server: SNA Diagnosis Vol 2, FFST Dumps and the VIT</i>	LY43-0088 LY43-0089	These documents help you identify an SNA problem, classify it, and collect information about it before you call the IBM Support Center. The information collected includes traces, dumps, and other problem documentation.
<i>z/OS Communications Server: SNA Data Areas Volume 1 and z/OS Communications Server: SNA Data Areas Volume 2</i>	LY43-0090 LY43-0091	These documents describe SNA data areas and can be used to read an SNA dump. They are intended for IBM programming service representatives and customer personnel who are diagnosing problems with SNA.

Messages and codes

Title	Number	Description
<i>z/OS Communications Server: SNA Messages</i>	SC31-8790	This document describes the ELM, IKT, IST, ISU, IUT, IVT, and USS messages. Other information in this document includes: <ul style="list-style-type: none"> • Command and RU types in SNA messages • Node and ID types in SNA messages • Supplemental message-related information
<i>z/OS Communications Server: IP Messages Volume 1 (EZA)</i>	SC31-8783	This volume contains TCP/IP messages beginning with EZA.
<i>z/OS Communications Server: IP Messages Volume 2 (EZB)</i>	SC31-8784	This volume contains TCP/IP messages beginning with EZB.
<i>z/OS Communications Server: IP Messages Volume 3 (EZY)</i>	SC31-8785	This volume contains TCP/IP messages beginning with EZY.
<i>z/OS Communications Server: IP Messages Volume 4 (EZZ-SNM)</i>	SC31-8786	This volume contains TCP/IP messages beginning with EZZ and SNM.
<i>z/OS Communications Server: IP and SNA Codes</i>	SC31-8791	This document describes codes and other information that appear in z/OS Communications Server messages.

APPC Application Suite

Title	Number	Description
<i>z/OS Communications Server: APPC Application Suite User's Guide</i>	SC31-8809	This documents the end-user interface (concepts, commands, and messages) for the AFTP, ANAME, and APING facilities of the APPC application suite. Although its primary audience is the end user, administrators and application programmers may also find it useful.

Title	Number	Description
<i>z/OS Communications Server: APPC Application Suite Administration</i>	SC31-8835	This document contains the information that administrators need to configure the APPC application suite and to manage the APING, ANAME, AFTP, and A3270 servers.
<i>z/OS Communications Server: APPC Application Suite Programming</i>	SC31-8834	This document provides the information application programmers need to add the functions of the AFTP and ANAME APIs to their application programs.

Communicating Your Comments to IBM

If you especially like or dislike anything about this document, please use one of the methods listed below to send your comments to IBM. Whichever method you choose, make sure you send your name, address, and telephone number if you would like a reply.

Feel free to comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this document. However, the comments you send should pertain to only the information in this manual and the way in which the information is presented. To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Please send your comments to us in either of the following ways:

- If you prefer to send comments by FAX, use this number: 1+919-254-9823
- If you prefer to send comments electronically, use this address:
 - comsvrcf@us.ibm.com.
- If you prefer to send comments by post, use this address:
 - International Business Machines Corporation
 - Attn: z/OS Communications Server Information Development
 - P.O. Box 12195, 3039 Cornwallis Road
 - Department AKCA, Building 501
 - Research Triangle Park, North Carolina 27709-2195

Make sure to include the following in your note:

- Title and publication number of this document
- Page number or topic to which your comment applies.



Program Number: 5694-A01 and 5655-G52

Printed in USA

SC31-8784-04



Spine information:



z/OS Communications Server

z/OS V1R6.0 CS: IP Messages Volume 2 (EZB, EZD)

Version 1
Release 6