HP 9000 Networking

NetWare® 4.1/9000
NetWare Client for DOS and MS Windows Technical Reference

HP Part No. J2771-90016
Printed in U.S.A. 12/96

Edition 1

Notice

Hewlett-Packard makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material. This product is based in whole or in part on technology developed by Novell, Inc

Hewlett-Packard assumes no responsibility for the use or reliability of its software on equipment that is not furnished by Hewlett-Packard

This document contains proprietary information, which is protected by copyright. All rights are reserved. No part of this document may be photocopied, reproduced, or translated into another language without the prior written consent of Hewlett-Packard Company.The information contained in this document is subject to change without notice.

UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Limited.
Microsoft®, MS®, and MS-DOS® are registered trademarks, and Windows is a trademark of Microsoft Corporation.
NetWare, and Novell are registered trademarks of Novell, Inc.

© Copyright 1996, Hewlett-Packard Company

Restricted Rights Legend

Use, duplication or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 for DoD agencies, Computer Software Restricted Rights clause at FAR 52.227-19 for other agencies.

Hewlett-Packard Co.
19420 Homestead Road
Cupertino, CA 95014 USA

Printing History

The manual printing date and part number indicate its current edition. The printing date will change when a new edition is printed. Minor changes may be made at reprint without changing the printing date. The manual part number will change when extensive changes are made.

Manual updates may be issued between editions to correct errors or document product changes. To ensure that you receive the updated or new editions, you should subscribe to the appropriate product support service. See your HP sales representative for details.

First Edition: December, 1996

Preface

Introduction

*NetWare Client for DOS and MS Windows Technical Reference p*rovides you with detailed information to configure the NetWare® DOS Requester™ software, modify the NetWare Client™ configuration file, and troubleshoot client workstation error messages in order to manage client workstations on a NetWare network.

This document is for supervisors responsible for managing NetWare client workstations.

NetWare Client for DOS and MS Windows Technical Reference covers concepts and procedures for configuring and using NetWare workstation software on NetWare 2, NetWare 3™, and NetWare 4™ networks. References are made to each version of NetWare. Ignore any references which do not pertain to the version of NetWare you are connecting to.

Use *NetWare Client for DOS and MS Windows User Guide* for procedures and information on installation and basic client workstation setup.

Contents Overview

To configure your NetWare Client software, use the chapters as described in the following checklist.

- Use Chapter 1, “Optimizing the NetWare Client Software,” to learn how to improve workstation performance by using Packet Burst™ and Large Internet Packets (LIP) and to enhance security on client workstations by using NCP™ packet signatures.
- Use Chapter 2, “NET.CFG Options Reference,” to learn how to set up and modify the NetWare Client (NET.CFG) configuration file and to reference information for setting NET.CFG option parameters.
- Use Chapter 3, “Command Line Parameters Reference,” to learn how to reference information for setting command line parameters.
- Use Chapter 4, “System Messages,” to receive an explanation of each client workstation system message and a recommendation on a course of action for each message.

Documentation Conventions

This manual uses the following Novell® conventions:

Asterisk (*)

An asterisk denotes a trademarked name belonging to a third-party company. Novell trademarks are denoted with specific trademark symbols (®, ™, etc.).

An ownership listing of all (Novell and third-party) trademarks cited in a manual can be found either on the disclaimer page in the front or in a "Trademarks" section at the back of printed manuals. A trademarks list is also available in the DynaText* online documentation.

Commands

Boldface characters indicate items that you type, such as commands and options. You can use any combination of uppercase and lowercase letters.

For example:

```
C:\A INSTALL
```

Delimiter Bar (|)

In syntax examples, a delimiter bar separating two command options indicates that you can choose one of the options.

For example:

```
–S | –R
```

Do not type the bar.

DOS Commands

DOS commands and command option letters are shown in uppercase letters. For example: FTPD.

Because DOS is not case-sensitive, you can type DOS commands in uppercase or lowercase letters.

DOS Filenames, Directory Names, and Pathnames

DOS filenames, directory names, and pathnames are shown in uppercase letters. For example, AUTOEXEC.BAT.

Because DOS is not case-sensitive, you can type these names in uppercase or

lowercase letters.

Ellipses

Ellipses in syntax examples indicate that parameters, options, or settings can be repeated.

For example, in the command

```
LOGIN SERVER1/SUPERVISOR /option...
```

you could replace option with any number of available options.

Emphasis

Italic type also indicates emphasized text. For example:

Remember to load the driver before you install the application.

Key Names

Angle brackets surround the name of a key. For example, <Enter> corresponds to the Enter key on your keyboard. <Ctrl>+<c> means hold down the Ctrl key and simultaneously type the letter c (in lowercase, in this case).

NET.CFG File Section Headings and Parameter Settings

NET.CFG section headings and parameter settings are shown in uppercase when used as a reference item and lower case when used in syntax or working examples.

For example:

[Begin example]

NETBIOS VERIFY TIMEOUT specifies how often in (ticks) NetBIOS sends a keep-alive packet to the other side of a session to preserve the session.

If no packets are being exchanged on the NetBIOS session by the software that established the session, NetBIOS sends packets at regular intervals to make sure that the session is still valid.

Syntax	`netbios verify timeout` *`number`* Replace *number* with a number of ticks.

Default	54 (approximately 3 seconds)
Range	4 to 65,535

Example	To make NetBIOS wait longer before sending a request-for-acknowledgment packet, you could place the following lines in your NET.CFG file:
	`netbios`
	`netbios verify timeout 1350`

[End example]

Because interpretation of this file is not case-sensitive, you can type its contents in uppercase or lowercase letters.

Options

In syntax examples, braces indicate that you are required to choose one of the enclosed options. For example, the following notation means that you must include a 0 or a 1 in the command:

```
{0, 1}
```

Square Brackets

In syntax examples, boldface type enclosed in square brackets indicates command options that you can type as needed. For example:

```
FTP [ –D ] [ –F ]
```

System Response

Monospace type shows system-generated responses that appear on your workstation screen. For example:

`TNVT220>`

UNIX Commands

UNIX® commands are shown in boldface letters. For example, vi. Because UNIX is case-sensitive, these commands are usually lowercase. Type UNIX commands exactly as shown.

UNIX Filenames, Directory Names, and Pathnames

UNIX filenames, directory names, and pathnames are shown in italics. For

example, /etc/hosts.

Because UNIX is case-sensitive, these names usually are in lowercase letters. Type UNIX filenames exactly as shown.

Variables

Italic type indicates variables—descriptive item names, such as command parameters—that you replace with appropriate values.

For example, in the command

```
FTP –F remote_host
```

you type the name of a computer on your network in place of remote_host.

Supplemental Documentation

The following publications provide supplemental information specifically related to the NetWare Client for DOS and MS Windows technology and software:

- "The Functions and Operations of the NetWare DOS Requester 1.1," Novell Application Notes, May 94, Vol. 5, No. 5 (Novell part no. 164-000031-005)
- "Installing and Configuring Novell's Token-Ring Source Routing Drivers," NetWare Application Notes, Oct 91 (Novell part no. 164-000030-010)
- "Logging In to IBM LAN Server and NetWare from a DOS Workstation," NetWare Application Notes, Nov 91 (Novell part no. 164-000030-011)
- "Managing Memory in a DOS Workstation: Part 1," NetWare Application Notes, Aug 92 (Novell part no. 164-000031-008)
- "Managing Memory in a DOS Workstation: Part 2," NetWare Application Notes, Oct 92 (Novell part no. 164-000031-010)
- "Managing Memory in a DOS Workstation: Using Novell DOS 7," NetWare Application Notes, Oct 93 (Novell part no. 164-000032-010)
- "Migrating Ethernet Frame Types from 802.3 Raw to IEEE 802.2," NetWare Application Notes, Sep 93 (Novell part no. 164-000032-009)
- "Multilingual PC Setup with DR DOS," NetWare Application Notes, Sep 93 (Novell part no. 164-000032-009)
- "NET.CFG Parameters for the NetWare DOS Requester 1.1," Novell Application Notes, Jun 94, Vol. 5, No. 6 (Novell part no. 164-000036-006)
- "NetWare and LAN Server Client Interoperability via ODINSUP: Part 1,"

NetWare Application Notes, Sep 92 (Novell part no. 164-000031-009)

- "NetWare and LAN Server Client Interoperability via ODINSUP: Part 2," NetWare Application Notes, Nov 92 (Novell part no. 164-000031-011)
- "NetWare and Windows for Workgroups 3.1 Interoperability," NetWare Application Notes, Mar 93 (Novell part no. 164-000032-003)
- NetWare Client for DOS and MS Windows User Guide, Novell Publication (Novell part no. 100-001623-002)
- "ODINSUP Interoperability Configurations for DOS Workstations," NetWare Application Notes, Feb 93 (Novell part no. 164-000032-002)
- "Using the DOS Requester with NetWare 4.0," NetWare Application Notes, Apr 93 (Novell part no. 164-000032-004)
- "Understanding Token-Ring Source Routing," NetWare Application Notes, May 91 (Novell part no. 164-000030-005)
- "Workstation Memory Management: Using QEMM386, 386 To The Max, and MS-DOS 6," NetWare Application Notes, Dec 93 (Novell part no. 164-000032-012)

Contents

1 Optimizing the NetWare Client Software

Overview 1-2

Introduction 1-3

Increasing Speed 1-4

Using the Packet Burst Protocol 1-4
Requirement for Packet Burst 1-4
How Packet Burst Works 1-4
When to Use Packet Burst 1-5
Configuring for Packet Burst 1-5
Disabling Packet Burst 1-5
Using Large Internet Packet Functionality 1-5
How Large Internet Packet Works 1-6
When to Use Large Internet Packet 1-6
Configuring for Large Internet Packet 1-6
Disabling LIP 1-7

Improving Security 1-8

Using NCP Packet Signature to Improve Security 1-8
How NCP Packet Signature Works 1-8
When to Use NCP Packet Signature 1-9
NCP Packet Signature Options 1-9
Effective Packet Signature Levels 1-10
Examples of Using Packet Signature Levels 1-10
All Information on the Server Is Sensitive 1-10
Sensitive and Nonsensitive Information Reside on the Same Server 1-11
Client Workstation Users Often Change Locations 1-11
Client Workstation Is Publicly Accessible 1-11
Installing NCP Packet Signature 1-11
Workstation Setting 1-11
Server Setting 1-12
Disabling Packet Signature 1-12

Contents

Troubleshooting NCP Packet Signature 1-13
Client Workstations Are Not Signing Packets 1-13
Client Workstations Cannot Log In 1-13
The Error Message “Error Receiving from the Network” Appears 1-14
Third-Party NLM Programs Do Not Work 1-14
Insecure Client Workstations Log In to a Secure Server 1-14

Using Other Client Security Guidelines 1-15

Additional Information 1-16

2 NET.CFG Options Reference

Overview 2-2

Introduction 2-3

Creating and Modifying a NET.CFG File 2-4
Entering Options and Parameters into the NET.CFG File 2-4
Sample NET.CFG File 2-5

Using NET.CFG Options and Parameters 2-7

Using the NET.CFG Reference Pages 2-12

Desktop SNMP Option 2-13
Available Parameters and Values for the Desktop SNMP Option 2-13
DESKTOP SNMP 2-13
Asynchronous Timeout Connections 2-14
ASYNCHRONOUS TIMEOUT number 2-14
Community Types and Names 2-15
MONITOR COMMUNITY [“name | public | private”] 2-17
CONTROL COMMUNITY [“name | public | private”] 2-18

Contents

TRAP COMMUNITY [“name | public | private”] 2-18
Community Access Management 2-18
ENABLE MONITOR COMMUNITY [specified | any | off | omitted] 2-20
ENABLE CONTROL COMMUNITY [specified | any | off | omitted] 2-20
ENABLE TRAP COMMUNITY [specified | off | omitted] 2-20
MIB-II (Management Information Base) Support 2-22
System and SNMP Groups 2-22
SNMPENABLEAUTHENTRAP [on | off] 2-24
SYSCONTACT “contact” 2-24
SYSLOCATION “location” 2-25
SYSNAME “name” 2-25
Interface Group 2-25
TCP/IP Groups 2-26
Example of NET.CFG File Including Each Group Support 2-26

Link Driver Option 2-27

Available Parameters and Values for the Link Driver Option 2-27
LINK DRIVER *driver_name* *2-27*
ALTERNATE 2-28
BUS ID name number 2-28
DMA [#1 | #2] channel_number 2-29
FRAME frame_type_name [addressing_mode] 2-30
Frame Types, Protocols, and LAN Drivers 2-31
Ethernet LAN Drivers 2-33
Token-Ring LAN Drivers 2-33
IRQ [#1 | #2] interrupt_request_number 2-34
MAX FRAME SIZE number 2-34
MEM [#1 | #2] hex_starting_address [hex_length] 2-35
NODE ADDRESS hex_address [mode] 2-36
LANSUP 2-37
PORT [#1 | #2] hex_starting_address [hex_number_of_ports] 2-38
PROTOCOL “name” hex_protocol_ID frame_type 2-39
Defined Protocols and Frame Types 2-39
SLOT number 2-40

Contents

Listing of Commonly Used ODI LAN Drivers 2-41

Link Support Option 2-46

Available Parameters and Values for the Link Support Option 2-46
LINK SUPPORT 2-46
BUFFERS communication_number [buffer_size] 2-47
MAX BOARDS number 2-49
MAX STACKS number 2-50
MEMPOOL number [k] 2-50

NetWare DOS Requester Option 2-52

Current Core Virtual Loadable Module (VLM) Programs 2-52
Current Non-Core Virtual Loadable Module Programs 2-53
Compatibility with NetWare Shell Parameters 2-54
Managing the NetWare DOS Requester 2-56
Optimizing the NetWare DOS Requester 2-57
Best Performance 2-57
Best Conventional Memory Usage 2-59
Best Compromise 2-60
Available Parameters and Values for the NetWare DOS Requester Option 2-61
NETWARE DOS REQUESTER 2-63
AUTO LARGE TABLE=[on | off] 2-63
AUTO RECONNECT=[on | off] 2-64
AUTO RETRY=number 2-64
AVERAGE NAME LENGTH=number 2-65
BIND RECONNECT=[on | off] 2-66
BROADCAST RETRIES=number 2-66
BROADCAST SEND DELAY=number 2-67
BROADCAST TIMEOUT=number 2-67
CACHE BUFFER SIZE=number 2-68
CACHE BUFFERS=number 2-69
CACHE WRITES=[on | off] 2-69
CHECKSUM=number 2-70
CONFIRM CRITICAL ERROR ACTION=[on | off] 2-71

Contents

CONNECTIONS=number 2-72
DOS NAME="name" 2-72
EOJ=[on | off] 2-73
EXCLUDE VLM=path_vlm 2-74
FIRST NETWORK DRIVE=drive_letter 2-74
FORCE FIRST NETWORK DRIVE=[on | off] 2-75
HANDLE NET ERRORS=[on | off] 2-75
LARGE INTERNET PACKETS=[on | off] 2-76
LIP START SIZE=number 2-77
LOAD CONN TABLE LOW=[on | off] 2-77
LOAD LOW CONN=[on | off] 2-78
LOAD LOW IPXNCP=[on | off] 2-79
LOAD LOW REDIR=[on | off] 2-79
LOCAL PRINTERS=number 2-80
LOCK DELAY=number 2-81
LOCK RETRIES=number 2-81
LONG MACHINE TYPE="name" 2-82
MAX TASKS=number 2-83
MESSAGE LEVEL=number 2-83
MESSAGE TIMEOUT=number 2-84
MINIMUM TIME TO NET=number 2-85
NAME CONTEXT="name_context" 2-85
NETWARE PROTOCOL=netware_protocol_list 2-86
NETWORK PRINTERS=number 2-87
PB BUFFERS=number 2-88
PBURST READ WINDOWS SIZE=number 2-88
PBURST WRITE WINDOWS SIZE=number 2-89
PREFERRED SERVER="server_name" 2-89
PREFERRED TREE="tree_name" 2-90
PREFERRED WORKGROUP="workgroup_name" 2-91
PRINT BUFFER SIZE=number 2-91
PRINT HEADER=number 2-92
PRINT TAIL=number 2-92
READ ONLY COMPATIBILITY=[on | off] 2-93

Contents

RESPONDER=[on | off] 2-94
SEARCH MODE=number 2-94
SET STATION TIME=[on | off] 2-96
SHORT MACHINE TYPE="name" 2-96
SHOW DOTS=[on | off] 2-97
SIGNATURE LEVEL=number 2-97
TRUE COMMIT=[on | off] 2-98
USE DEFAULTS=[on | off] 2-99
VLM=path_VLM 2-100
WORKGROUP NET=workgroup_net_address 2-101

Protocol IPX Option 2-103

Available Parameters and Values for the Protocol IPX Option 2-103
PROTOCOL IPX 2-103
BIND LAN_driver_name [#number] 2-104
INT64 [on | off] 2-104
INT7A [on | off] 2-105
IPATCH byte_offset, value 2-106
IPX PACKET SIZE LIMIT number 2-106
IPX RETRY COUNT number 2-107
IPX SOCKETS number 2-107

Protocol SPX Option 2-109

Available Parameters and Values for the Protocol SPX Option 2-109
PROTOCOL SPX 2-109
MINIMUM SPX RETRIES number 2-110
SPX ABORT TIMEOUT number 2-110
SPX CONNECTIONS number 2-111
SPX LISTEN TIMEOUT number 2-111
SPX VERIFY TIMEOUT number 2-112

Protocol TCPIP Option 2-114

Available Parameters and Values for the Protocol TCPIP Option 2-114
PROTOCOL TCPIP 2-115

Contents

LAN Drivers 2-115
BIND odi_driver [number frame_type network_name] 2-116
IP Addresses 2-117
IP_ADDRESS ip_address [network_name] 2-118
IP_NETMASK net_mask_address [network_name] 2-119
IP_ROUTER ip_address [network_name] 2-120
Connection Sockets 2-120
Transmission Control Protocol (TCP) Sockets 2-121
TCP_SOCKETS number 2-121
User Datagram Protocol (UDP) Sockets 2-122
UDP_SOCKETS number 2-122
Raw Sockets 2-123
RAW_SOCKETS number 2-123
Additional Support 2-124
NO_BOOTP 2-124
PATH TCP_CFG [[drive:]path [; ...]] 2-125

Transport Provider IPX | UDP Option 2-126

Available Parameters and Values for the Transport Provider IPX | UDP Option 2-126
TRANSPORT PROVIDER IPX | UDP 2-126
TRAP TARGET ipxaddress | ipaddress 2-127

3 Command Line Parameters Reference

Overview 3-2

Introduction 3-3

Core NetWare Client Software 3-4

IPXODI.COM 3-5
LSL.EXE 3-6
ODI LAN driver.COM 3-7
VLM.EXE 3-7

Contents

DOSNP Software 3-10

4 System Messages

1

Optimizing the NetWare Client Software

Overview

This chapter explains how to optimize the NetWare® Client™ software for increasing the speed of client workstations by using the Packet Burst™ protocol and Large Internet Packets (LIP). It also explains how to protect information on client workstations.

The following topics are covered in this chapter.

Topic
Increasing Speed
Improving Security
Using Other Client Security Guidelines

Introduction

You can increase the speed and improve the security of client workstations by using the Packet Burst protocol and Large Internet Packets (LIPs), and by implementing the NCP™ packet signature feature available in NetWare 4™ and 3.12 software.

Increasing Speed

NetWare 3.12 and 4 support the Packet Burst and Large Internet Packet technologies which increase the access speed of network resources and services for client workstations.

Using the Packet Burst Protocol

The Packet Burst protocol allows high-performance data transmission between client workstations and servers.

Some network topologies, such as Ethernet and token ring, allow large packets to be sent over the network. The LIP (Large Internet Packet) capability enhances throughput over bridges or routers by increasing the packet size.

The following sections provide you with information and procedures for setting parameters used in the client workstation configuration file (NET.CFG).

Packet Burst on the client workstation is enabled automatically in the NetWare DOS Requester™ software.

Requirement for Packet Burst

The Packet Burst protocol code requires about 6 KB of memory. However, as a default, the NetWare DOS Requester uses the Open Data-Link Interface™ architecture for Packet Burst and doesn't require additional workstation memory.

How Packet Burst Works

At connection time, maximum burst sizes are negotiated with each server. Since Packet Burst is established with each connection, it's possible to "burst" with one server but not with another.

Once you establish a Packet Burst connection between a client workstation and a NetWare server, the client workstation automatically uses the Packet Burst service whenever an application requests to write more than one physical packet of data.

When to Use Packet Burst

Packet Burst is not required for every installation; however, disabling LIP will results in noticeable speed degradation. Some network supervisors might choose not to use Packet Burst because some of the servers that the client workstations are connecting to do not support it.

Configuring for Packet Burst

Although Packet Burst is automatically enabled in the NetWare DOS Requester, you can configure it for your needs.

See “PB BUFFERS=number” , “PBURST READ WINDOWS SIZE=number” , and “PBURST WRITE WINDOWS SIZE=number” for details on how to configure for Packet Burst.

Disabling Packet Burst

To disable Packet Burst at client workstations, add this line to the NET.CFG file under the “NetWare DOS Requester” option heading:

```
pb buffers = 0
```

For example, you would type

```
netware dos requester
   pb buffers=0
```

Using Large Internet Packet Functionality

Large Internet Packet (LIP) functionality allows the packet size to be increased from the default of 576 bytes. LIP is enabled automatically in the NetWare DOS Requester software.

Previously, the size of packets that cross bridges or routers on NetWare networks was limited to 576 total bytes. Some network architectures like Ethernet and token ring allow larger packets to be sent over the network.

By allowing the packet size to be increased, LIP enhances the throughput over bridges and routers if the routers aren’t limited to the smaller packet size.

The following sections provide you with information and procedures for setting parameters used in the client workstation configuration file (NET.CFG).

The Large Internet Packet technology on the client workstation is enabled automatically in the NetWare DOS Requester software.

NOTE: Some LAN drivers might not operate correctly using this parameter. If you experience trouble, disable this parameter or update the version of your LAN driver.

How Large Internet Packet Works

In previous NetWare versions, the NetWare Client™ software initiated a negotiation with the NetWare server to determine an acceptable packet size.

If the NetWare server software detected a router between it and the client workstation, the server returned a maximum packet size of 576 bytes to the NetWare Client software.

In the current NetWare version, the NetWare Client software still initiates packet size negotiation. However, because of LIP, the NetWare server no longer returns a packet size of 576 bytes when a router is detected.

Instead, the NetWare Client software negotiates with the NetWare server software to agree on the largest packet size available.

When to Use Large Internet Packet

Large Internet Packet is not required for every installation; however, disabling LIP results in noticeable speed degradation. Some network supervisors might choose not to use Large Internet Packet because some of the servers that the client workstations are connecting to do not support it, such as NetWare 2 and NetWare 3.11 and earlier.

Configuring for Large Internet Packet

Although LIP is automatically enabled in the NetWare DOS Requester, you can configure it for your needs.

See “LARGE INTERNET PACKETS=[on | off]” for details on how to configure for Packet Burst.

Disabling LIP

To disable LIP functionality at the client workstation, add this line to the NET.CFG file under the “NetWare DOS Requester” option heading:

```
large internet packets = off
```

For example, you would type

```
netware dos requester
   large internet packets = off
```

Improving Security

You can increase the security of your network by using the NCP packet signature feature available in NetWare 4 and 3.12.

The following sections provide you with information and procedures for setting a parameter used in the client workstation configuration (NET.CFG) file and the SET command used at each NetWare server.

Using NCP Packet Signature to Improve Security

NCP packet signature is an enhanced security feature that protects servers and client workstations using the NetWare Core Protocol™ architecture by preventing packet forgery.

The NCP packet signature is optional because the packet signature process consumes CPU resources and slows performance, both for the client workstation and the NetWare server.

Without the NCP packet signature installed, a knowledgeable network operator can manipulate the client workstation software to send a forged NCP request to a NetWare server. By forging the proper NCP request packet, an intruder can gain rights to access all network resources.

How NCP Packet Signature Works

NCP packet signature prevents forgery by requiring the server and the client workstation to “sign” each NCP packet, using the RSA public and private key encryption. The packet signature changes with every packet.

NCP packets with incorrect signatures are discarded without breaking the client workstation’s connection with the server. However, an alert message about the source of the invalid packet is sent to the error log, the affected client workstation, and the NetWare server console.

If NCP packet signature is installed on the server and all of the network client workstations, it is virtually impossible to forge an NCP packet that would appear valid.

When to Use NCP Packet Signature

NCP packet signature is not required for every installation. Some network supervisors might choose not to use it because they can tolerate certain security risks.

Tolerable Security Risks The following are examples of network situations that might not need NCP packet signature:

- Only executable programs reside on the server
- All client workstation users on the network are known and trusted by the network supervisor
- Data on the NetWare server is not sensitive; access, loss, or corruption of this data would not affect operations

Serious Security Risks NCP packet signature is recommended for security risks such as these:

- Unauthorized client workstation users on the network
- Easy physical access to the network cabling system
- An unattended, publicly accessible client workstation within your network

NCP Packet Signature Options

Several signature options are available, ranging from never signing NCP packets to always signing NCP packets. NetWare servers and network client workstations both have four signature levels, which are explained in the following table.

Table 1-1 NCP Packet Signature Levels

Level Number	Explanation
0	Doesn’t sign packets.
1	Signs packets only if the server requests it (NetWare server NCP option is 2 or higher).
2	Signs packets if the server is capable of signing (NetWare server NCP option is 1 or higher).
3	Signs packets and requires the server to sign packets (or logging in will fail).

Effective Packet Signature Levels

The signature levels for the server and the client workstations combine to determine the overall level of NCP packet signature on the network called the effective packet signature level.

Some combinations of server and client packet signature levels might slow performance. However, low-CPU-demand systems might not show any performance degradation.

You can choose the packet signature level that meets both their performance needs and their security requirements.

The following table shows the interactive relationship between the server packet signature levels and the client workstation signature levels.

Table 1-2 Effective Packet Signature Combinations of Server and Client Workstations

IF	Server = 0	Server = 1	Server = 2	Server = 3
Client Workstation = 0	No packet signature	No packet signature	No packet signature	No logging in
Client Workstation = 1	No packet signature	No packet signature	Packet signature	Packet signature
Client Workstation = 2	No packet signature	Packet signature	Packet signature	Packet signature
Client Workstation = 3	No logging in	Packet signature	Packet signature	Packet signature

Examples of Using Packet Signature Levels

This section includes some examples of when you would use different signature levels.

All Information on the Server Is Sensitive

Example	If an intruder gains access to any information on the NetWare server, it could damage the company.
Solution	The network supervisor sets the server to level 3 and all client workstations to level 3 for maximum protection.

Sensitive and Nonsensitive Information Reside on the Same Server

Example	The NetWare server has a directory for executable programs and a separate directory for corporate finances (such as accounts receivable).
Solution	The network supervisor sets the server to level 2 and the client workstations that need access to accounts receivable to level 3. All other client workstations remain at the default level 1.

Client Workstation Users Often Change Locations

Example	The network supervisor is uncertain which employees will be using which client workstations, and the NetWare server contains some sensitive data.
Solution	The network supervisor sets the server to level 3. Client workstations remain at the default level 1.

Client Workstation Is Publicly Accessible

Example	An unattended client workstation is set up for public access to nonsensitive information, but another server on the network contains sensitive information.
Solution	The network supervisor sets the sensitive server to level 3 and the unattended client workstation to level 0.

Installing NCP Packet Signature

To install the NCP packet signature support, you must set a parameter used in the NET.CFG file on each client workstation and a SET command used at each NetWare server.

Workstation Setting

To install NCP packet signature on a DOS or MS Windows client workstation, add this line to the NET.CFG file under the NetWare DOS Requester option:

```
signature level = number
```

For example, you would type

```
netware dos requester
   signature level = 2
```

Replace number with 0, 1, 2, or 3. The default is level 1, which provides the most flexibility while still offering protection from forged packets.

See “SIGNATURE LEVEL=number” for details on how to configure for NCP packet signature support on the client workstation.

NOTE: Some LAN drivers might not operate correctly using this parameter. If you experience trouble, disable this parameter or update the version of your LAN driver.

Server Setting

To ensure that the SET parameter “NCP PACKET SIGNATURE OPTION” is added to the system at each server you want NCP packet signature support on, type the following command at each server console:

```
SET NCP PACKET SIGNATURE OPTION = number <Enter>
```

Replace number with 0, 1, 2, or 3. The default is level 1, which provides the most flexibility while still offering protection from forged packets.

See “Preventing Packet Forgery,” in Chapter 7 of Supervising the NetWork for details on how to configure for NCP packet signature support on the server.

Disabling Packet Signature

To disable NCP packet signature support at the client workstation, add this line to the NET.CFG file under the “NetWare DOS Requester” option heading:

```
signature level = 0
```

For example, you would type

```
netware dos requester
   signature level = 0
```

For explanations of packet signature levels and their combined use, see Table 1-1.

Troubleshooting NCP Packet Signature

This section describes some solutions to problems that might be associated with using NCP packet signature.

Client Workstations Are Not Signing Packets

Problem	Client workstations are not signing NCP packets.
Solution	The SECURITY.VLM file is not loading. Ensure the signature level on the client workstation is not set to 0. SECURITY.VLM loads by default when the client signature level is set to 1, 2 or 3. Use the VLM /V4 command line parameter when loading the VLM software to display load time information.

Client Workstations Cannot Log In

Problem	Client workstations cannot log in.
Solution	Make sure the packet signature levels on the server and the client workstation are correct. See “Effective Packet Signature Levels”. Note that the following situations do not allow logging in: • Server packet signature = 3, client workstation signature = 0 • Server packet signature = 0, client workstation signature = 3 • The LOGIN utility is an older version that doesn’t support packet signature • The NetWare DOS Requester or the shell is an older version that doesn’t support packet signature

The Error Message "Error Receiving from the Network" Appears

Problem	The error message "Error receiving from the network" appears.
Solution	The client workstation is using a version of LOGIN.EXE file that doesn't include NCP packet signature. Make sure the new LOGIN.EXE and other new utility files are installed on all NetWare servers on the network.

Third-Party NLM Programs Do Not Work

Problem	Third-party NetWare Loadable Module™ (NLM) programs do not work.
Solution	If the SET parameter "Allow Change to Client Rights" is set to "OFF," some third-party NLM™ programs might not function. Set this parameter to "ON."

Insecure Client Workstations Log In to a Secure Server

Problem	Insecure client workstations log in to a secure server.
Solution	The client workstations are using an old LOGIN.EXE file that does not include NCP packet signature. Make sure the new LOGIN.EXE and other new utility files are installed on all servers on the network. Use the "Preferred Server" option in the NET.CFG file for all client workstations that have access to secure servers (level 3). See "PREFERRED SERVER="server_name" for details on how to configure for this option.

Using Other Client Security Guidelines

In addition to installing NCP packet signature, you can use other NetWare security features and protective measures to keep client workstations secure.

We suggest the following security guidelines for client workstations:

- Use only the most current versions of system software, NetWare Client software, and patches.
- Check for viruses regularly.
- Use the SECURITY utility to detect vulnerable access points to the server.
- Enable intruder detection and lockout.
- Advise users to log out when their client workstations are unattended.
- Enable NCP packet signature level 3 on all unattended client workstations.
- Require passwords of at least five characters on all accounts.
- Force password changes at least every three months.
- Require unique passwords.
- Limit the number of grace logins.
- Limit the number of concurrent connections.
- Enforce login time restrictions and station restrictions.

Additional Information

Topic	Reference
Setting up and modify your NET.CFG file for Packet Burst, LIP, and NCP packet signatures	Chapter 2, “NET.CFG Options Reference”
Setting up Packet Burst support on a NetWare server	Chapter 7, “Maintaining the NetWare Server,” in *Supervising the Network*
Setting up Large Internet Packet (LIP) support on a NetWare server	Chapter 7, “Maintaining the NetWare Server,” in *Supervising the Network*
Setting up NCP packet signature support on a NetWare server	Chapter 7, “Maintaining the NetWare Server,” in *Supervising the Network*

2

NET.CFG Options Reference

Overview

This chapter explains how to create or modify a NET.CFG file and contains an alphabetical listing and discussions of the currently available NET.CFG file options.

The following topics are covered in this chapter.

Topic
Creating and Modifying a NET.CFG File
Using NET.CFG Options and Parameters
Using the NET.CFG Reference Pages

Introduction

NET.CFG is the configuration file that you use to specify nondefault value settings for your NetWare Client™ software configuration options.

Use entries in the NET.CFG file to change the client workstation's network environment or configuration. For example, you might want to change the configuration in these cases:

- You changed the default hardware settings on the network board
- You are using multiple protocols
- You are using the Novell® LAN Workplace® software

Creating and Modifying a NET.CFG File

1 Use a DOS text editor to type section headings and options in an existing NET.CFG file or a NET.CFG file that you create to set up your client workstation configuration.

   The default location of the NET.CFG file is C:\NWCLIENT.

2 After a NET.CFG file is created or modified, copy or save the NET.CFG file to the client workstation diskette or directory.

   If all client workstations use the same NET.CFG file, you can save time by copying the file onto a master client workstation diskette, then copying the NET.CFG file to each client workstation. (The default location of the NET.CFG file is C:\NWCLIENT.)

   If each client workstation requires a unique NET.CFG file, you must copy a unique file to each client workstation diskette or directory.

Entering Options and Parameters into the NET.CFG File

Use the following conventions when creating or modifying a NET.CFG file:

- Type one option or parameter per line.

  Options and parameters are not case-sensitive unless used in quotation marks. Blank lines are ignored, but they can be helpful in separating the options and parameters to make the NET.CFG file easier to read.

- Enter options at the left margin of the file with no spaces before or after them.

  Each option can have several parameters.

- Enter parameters, one per line, below the option that they apply to, and indent each parameter line.

  Use the Spacebar or Tab key to indent parameters. Parameters must be indented at least one space.

- Place a hard return at the end of every line in the file, including the last line.

  If you don’t put a return at the end of the last line, that line is ignored.

- Precede comment lines with a semicolon.

- User a semicolon (;) in front of a line to disable the option and parameter or write a comment.

If a semicolon is placed in front of a NET.CFG option line, all of the parameters listed below are disabled.

The following figure illustrates the NET.CFG file format.

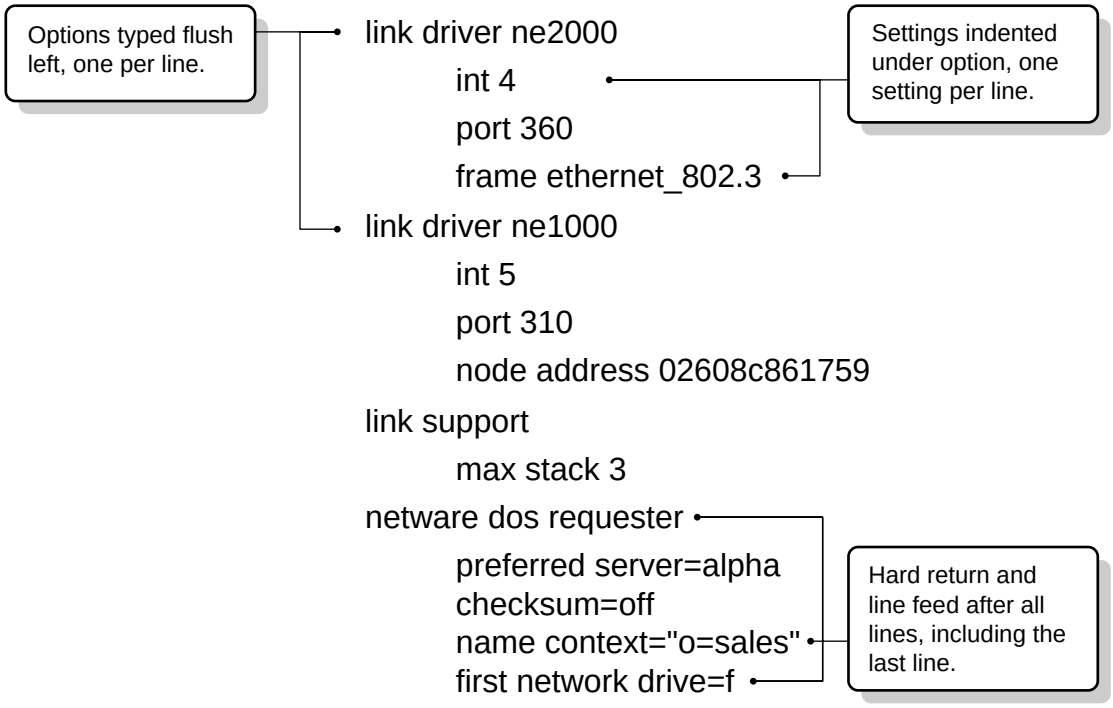

Figure 2-1 NET.CFG File Format

Sample NET.CFG File

Following is a sample NET.CFG file that

- Changes the IRQ on the link driver to 4
- Changes the port to 340
- Sets up F: as the first network drive when logging in to the network

```
link driver ne2000
   ; Change the interrupt (IRQ) to 4
   int #1 4
```

```
      ; Change the port to 340 (hex)
      port #1 340
   netware dos requester
      ; Set up F: as the first drive on network
      first network drive = f
```

NOTE: Changing the value setting in the NET.CFG does not change the hardware setting for the device you are using. Run the appropriate configuration utility or manually change the appropriate jumper settings to correspond with value setting used in the NET.CFG file.

Using NET.CFG Options and Parameters

NetWare and Personal NetWare™ workstations support the following configuration options in the NET.CFG file:

- Desktop SNMP Option
- Link Driver Option
- Link Support Option
- NetWare DOS Requester Option
- Protocol IPX Option
- Protocol SPX Option
- Protocol TCPIP Option
- Transport Provider IPX | UDP Option

NOTE: Because many products use the NET.CFG file for managing configuration information, this might not be a comprehensive list of all the options used in your NET.CFG file.

See the third-party manufacturer's product documentation or the most current README file provided with this NetWare Client for DOS and MS Windows kit for information.

Table 2-1 shows the various parameters and values that you can use. Notice the following markings in figure text:

* The default value is the maximum value for NetWare 2 and NetWare 3™ networks.

+This option is valid for NetWare 4™ networks only.

~ Defaults vary depending on your network setup. See the corresponding option heading in this chapter for specific information.

Table 2-1 NET.CFG Options

desktop snmp	
asynchronous timeout *number*	20 ticks
control community [“*name* | public | private”]	public
enable control community [specified | any | off | omitted]	specified
enable monitor community [specified | any | off | omitted]	specified
enable trap community [specified | off | omitted]	specified
monitor community [“*name* | public | private”]	public
snmpenableauthentrap [on | off]	off
syscontact “*contact*”	(none)
syslocation “*location*”	(none)
sysname “*name*”	(none)
trap community [“*name* | public | private”]	public
link driver driver_name	
alternate	(none)
bus *name number*	(autodetect name), -1 (0FFh)
dma [#1 | #2] *channel_number*	#1, 3
frame *frame_type_name* [*addressing_mode*]	~
irq [#1 | #2] *interrupt_request_number*	#1, 3
max frame size *number*	1024
mem [#1 | #2] *hex_starting_address* [*hex_length*]	#1, D000, ~
node address *hex_address* [*mode*]	(none)
port [#1 | #2] *hex_starting_address* [*hex_number_of_ports*]	#1, 300, ~
protocol “*name*” *hex_protocol_id frame_type*	(none)
slot *number*	1
link support	
buffers *communication_number* [*buffer_size*]	0, 1130
max boards *number*	4
max stacks *number*	4
netware dos requester	
auto large table=[on | off]	off
auto reconnect=[on | off]	on
auto retry=*number*	0
average name length=*number*	48
bind reconnect=[on | off]	off
broadcast retries=*number*	3

Table 2-1 NET.CFG Options

broadcast send delay=*number*	0
broadcast timeout=*number*	2 ticks
cache buffer size=*number*	(media maximum: 64 bytes)
cache buffers=*number*	5
cache writes=[on | off]	on
checksum=*number*	1
confirm critical error action=[on | off]	on
connections=*number**	8
dos name=”*name*”	msdos
eoj=[on | off]	on
exclude vim=*path_vim*	(none)
first network drive=*drive_letter*	(first available drive)
force first network drive=[on | off]	off
handle net errors=[on | off]	on
large internet packets=[on | off]*	on
lip start size *number*	0
load conn table low=[on | off]	off
load low conn=[on | off]	on
local printers=*number*	3
lock delay=*number*	1 tick
lock retries=*number*	1 tick
long machine type=”*name*”	ibm-pc
max tasks=*number*	31
message level=*number*	1
message timeout=*number*	0
minimum time to net=*number*	0
name context=”*name_context*”+	root
netware protocol=*netware_protocol_list*	nds bind pnw
network printers=*number**	3
pb buffers=*number*	3
pburst read windows size=*number*	16
pburst write windows size=*number*	10
preferred server=”*server_name*”	(none)
preferred tree=”*tree_name*”+	(none)
preferred workgroup=”*workgroup_name*”	(none)
print buffer size=*number**	64
print header=*number**	64
print tail=*number*	16
read only compatibility=[on | off]	off
responder=[on | off]	on

Table 2-1 NET.CFG Options

search mode=*number*	1
set station time=[on | off]	on
show dots=[on | off]	off
short machine type="*name*'	ibm
signature level=*number*	1
true commit=[on | off]	off
use defaults=[on | off]	on
vim=*path_vim*	(none)
workgroup net=*workgroup_net_address*	(none)
protocol ipx	
bind *lan_driver_name* [#*number*]	(none)
int64 [on | off]	on
int7a [on | off]	on
ipatch *byte_offset, value*	(none)
ipx packet size limit *number*	(lesser of either 4160 or size specified by LAN driver)
ipx retry count *number*	20
ipx sockets *number*	20
protocol spx	
minimum spx retries *number*	20
spx abort timeout *number*	540 (=30 seconds)
spx connections *number*	15
spx listen timeout *number*	108 (=6 seconds)
spx verify timeout *number*	54 (=3 seconds)
protocol tcpip	
bind odi_driver [*number frame_type network_name*]	~
ip_address ip_address [*network_name*]	(none)
ip_netmask net_mask_address [*network_name*]	(none)
ip_router ip_address [*network_name*]	(none)
raw_sockets *number*	1
nb_adapter [0 | 1]	0
nb_brdcast [0 | 1]	1
nb_commands *number*	8
nb_domain *domain_name*	(none)
nb_sessions *number*	4

Table 2-1 NET.CFG Options

no_bootp path tcp_cfg [[*drive*:]*path* [; ...]] tcp_sockets *number* udp_sockets *number*	(none) c (drive) \net\tcp (path) 8 8
transport provider ipx | udp	
trap target *ipxaddress* | *ipaddress*	(none)

Using the NET.CFG Reference Pages

The following figure explains how to read the following NET.CFG reference pages in this chapter.

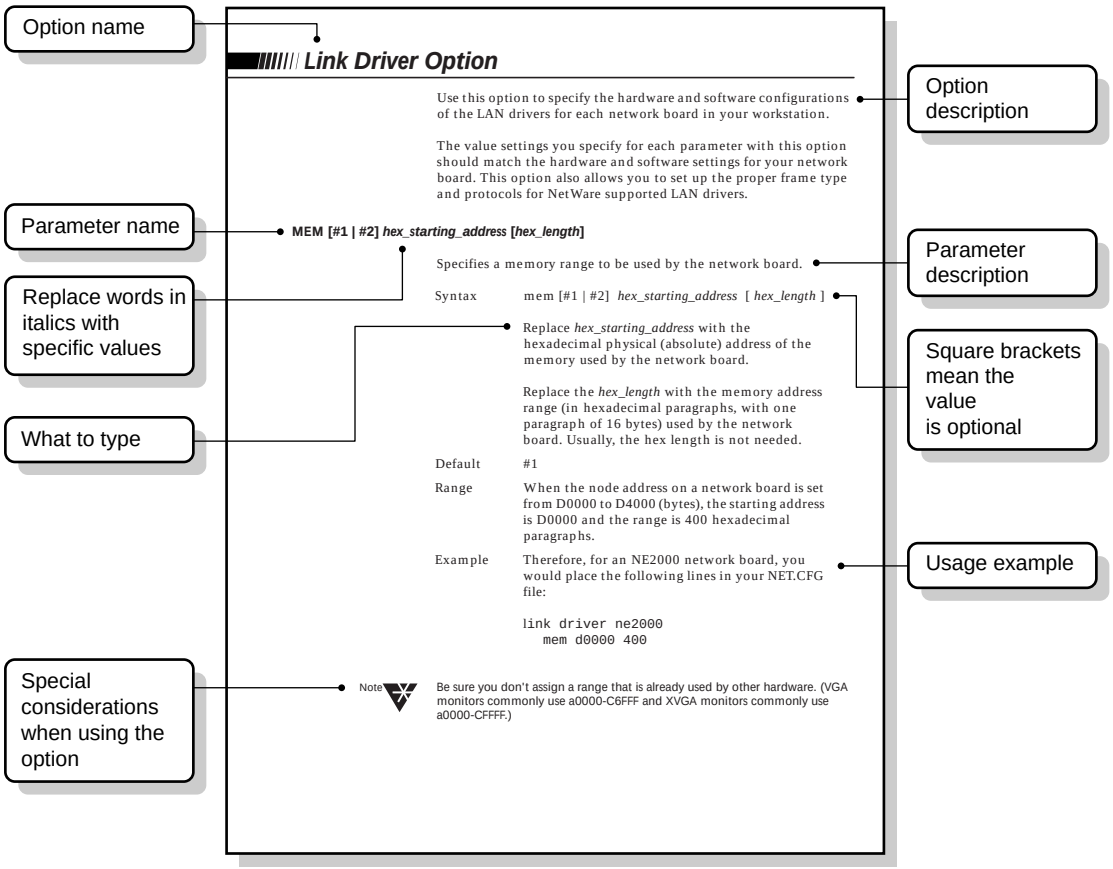

Figure 2-2 Format of NET.CFG Reference Pages

Desktop SNMP Option

Use this option to manage MIB-II support and communities for SNMP desktops on NetWare and Personal NetWare networks.

Available Parameters and Values for the Desktop SNMP Option

This option has the following categories, parameters, and values.

Category	Parameters and Values
Asynchronous Timeout Connections	ASYNCHRONOUS TIMEOUT number
Community Types and Names	MONITOR COMMUNITY [“name | public | private”] CONTROL COMMUNITY [“name | public | private”] TRAP COMMUNITY [“name | public | private”]
Community Access Management	ENABLE MONITOR COMMUNITY [specified | any | off | omitted] ENABLE CONTROL COMMUNITY [specified | any | off | omitted] ENABLE TRAP COMMUNITY [specified | off | omitted]
MIB-II (Management Information Base) Support	SNMPENABLEAUTHENTRAP [on | off] SYSCONTACT “contact” SYSLOCATION “location” SYSNAME “name”

DESKTOP SNMP

Specifies the desktop SNMP option you are making configurations for.

Syntax	desktop snmp *parameter_name value* Replace *parameter_name* with the name of the parameter you want to use. Replace *value* with the number or setting which corresponds with the parameter name.

Example	To identify the name of the system administrator for MIB II, you would place these lines in your NET.CFG file:
	
	`desktop snmp`
	`  sysname "Suzanne Morley x893"`

NOTE: For transport providers used with SNMP desktops, see "Transport Provider IPX | UDP Option".

Asynchronous Timeout Connections

Use this parameter and value to manage the timeout for asynchronous connections.

Parameter and Value
ASYNCHRONOUS TIMEOUT number

Desktop SNMP provides a way to monitor and control asynchronous connections for managing Desktop SNMP and other SNMP entities over asynchronous lines.

ASYNCHRONOUS TIMEOUT number

Monitors and controls your SNMP connections.

When an SNMP manager requests information from a managed object, the desktop SNMP waits a set amount of time before attempting to cancel requests it has made against managed objects.

Syntax	asynchronous timeout *number*
Default	20

Example	For example, to have Desktop SNMP wait 35 system ticks (approximately 2 seconds) before attempting to cancel a request, you would place the following lines in your NET.CFG file:
	
	`desktop snmp`
	`  asynchronous timeout 35`

NOTE: The timeout number is in ticks (18.21 ticks per second on IBM* PCs and compatibles).

Community Types and Names

Use these parameters and values to identify types and names of access control groups.

Parameters and Values
MONITOR COMMUNITY [“name | public | private”] CONTROL COMMUNITY [“name | public | private”] TRAP COMMUNITY [“name | public | private”]

Desktop SNMP provides default community names for the monitor (read-only) and control (read/write) communities, as well as a default community types used for traps. Desktop SNMP and other SNMP entities use these types and names for access control.

The community name contained in a request message from an SNMP management station must match the name expected by Desktop SNMP.

For examples of complete files, see “Example of NET.CFG Files Using the Community Name and Types.”

NOTE: If Desktop SNMP receives a request protocol data unit (PDU) whose community name is not authorized, it does not respond to the request.

For example, suppose the control community name is “secret,” and Desktop SNMP receives a SETRequest PDU with a community name of “public.” Desktop SNMP discards the SETRequest UDP and does not respond to the UDP. However, Desktop SNMP does send an authentication trap to the trap targets if SNMPENABLEAUTHENTRAPS ON.

A community name can be any arbitrary, case-sensitive ASCII string up to 32 characters in length. It can include any characters except space, tab, open square bracket ([ ), equal sign (=), colon (:), semicolon (;), single quotation mark (“), or number sign (#).

Community name strings are case-sensitive. Therefore, always enclose the community name string in double quotation marks.

In the Desktop SNMP section of the NET.CFG file, you can define three communities, as described in the following table:

Table 2-2 Desktop SNMP Option Parameters for Community Names

Parameter	Explanation
control community	Describes the read/write community (the community that is allowed to do SET operations). Any community name established for read/write access is also valid for read-only access. The default value is “public.” When the control community is disabled, all write access is disabled.
monitor community	Describes the read-only community (the community that is allowed to do GET and GET NEXT operations). The default value is “public.” When the monitor community is disabled, all read access is disabled.
trap community	Describes the community name used for traps. The default value is “public.” When the trap community name is disabled, Desktop SNMP does not send traps.

MONITOR COMMUNITY [“name | public | private”]

Specifies the monitor community name.

Syntax	monitor community [“*name* | public | private”]
Default	public

Example	To specify the monitor community as “private,” you would place the following lines in your NET.CFG file:
	
	`desktop snmp`
	`  monitor community “private”`

CONTROL COMMUNITY [“name | public | private”]

Specifies the control community name.

Syntax	control community [“*name* | public | private”]
Default	public

Example	To specify the control community as “secret,” you would place the following lines in your NET.CFG file:
	
	`desktop snmp`
	`  control community “secret”`

TRAP COMMUNITY [“name | public | private”]

Specifies the trap community name.

Syntax	trap community [“*name* | public | private”]
Default	public

Example	To specify the trap community as “agenttrap,” you would place the following lines in your NET.CFG file:
	
	`desktop snmp`
	`  trap community “agenttrap”`

Community Access Management

Use these parameters and values to manage access control of SNMP agents and resources.

Parameters and Values
ENABLE MONITOR COMMUNITY [specified | any | off | omitted] ENABLE CONTROL COMMUNITY [specified | any | off | omitted] ENABLE TRAP COMMUNITY [specified | off | omitted]

Community types can also be disabled. When a community type is disabled, no management entity can access information for that community that you named.

For example, if you disable the control community, no one can use Desktop SNMP to do SET operations against the data it manages.

For examples of complete files, see "Example of NET.CFG Files Using the Community Name and Types."

Desktop SNMP reads the community name definition as follows, depending on the enable community_type community line:

Table 2-3 Desktop SNMP Option Values for Community Type Parameters

Value	Explanation
Any	If enable community_type community is set to "any" for a community type, any community string can be used to gain access. The line defining the community name can be omitted. If it is present, Desktop SNMP ignores it. For trap community strings, the "any" option is not applicable. If you specify enable trap community any, Desktop SNMP interprets the line as if the community type were specified.
Off	If enable community_type community is set to "off" for a community type, access for that community type is disabled. The line defining the community name can be omitted. If the line is present, Desktop SNMP ignores it.
Omitted	If enable community_type community is set to "omitted" for a specific community type, Desktop SNMP defaults to "specified" for that community type and looks for a line defining a community name for the community type. If no community name is specified, Desktop SNMP defaults to "public."
Specified	If enable community_type community is set to "specified" for a community type, Desktop SNMP uses only the specified community name for that community type. If none is specified, Desktop SNMP defaults to "public."

ENABLE MONITOR COMMUNITY [specified | any | off | omitted]

Enables the value settings for the monitor community.

Syntax	enable monitor community [specified | any | off | omitted]
Default	specified

Example	To enable the monitor community as "private" only, you would place the following lines in your NET.CFG file:
	
	`desktop snmp`
	`  enable monitor community specified`

ENABLE CONTROL COMMUNITY [specified | any | off | omitted]

Enables the value settings for the control community.

Syntax	enable control community [specified | any | off | omitted]
Default	specified

Example	To enable full access to the control community "secret," you would place the following lines in your NET.CFG file:
	
	`desktop snmp`
	`  enable control community any`

ENABLE TRAP COMMUNITY [specified | off | omitted]

Enables the value settings for the trap community.

Syntax	enable trap community [specified | off | omitted]
Default	specified

Example	To disable the trap community name "agenttrap," you would place the following lines in your NET.CFG file:
	
	`desktop snmp`
	`  enable trap community off`

Example of NET.CFG Files Using the Community Name and Types

To use the default trap community, allow any community name to be used for read access, but set the read/write community name to “secret,” you would place the following lines in your NET.CFG file:

```
desktop snmp
   enable monitor community any
   enable control community specified
   control community “secret”
```

To disable all traps and use the default values for the monitor and control communities, you would place the following lines in your NET.CFG file:

```
desktop snmp
   enable trap community off
```

To specify a NULL string for the control community and use the default values for the monitor and trap communities, you would place the following lines in your NET.CFG file:

```
desktop snmp
   enable control community specified
   control community “”
```

Setting a NULL string is the same as setting the enable community_type community line to “off.”

To allow only the “private” community to have access and set the community name for traps to “agenttrap,” you would place the following lines in your NET.CFG file:

```
desktop snmp
   enable monitor community specified
   monitor community “private”
   enable control community specified
   control community “private”
   enable trap community specified
   trap community “agenttrap”
```

To temporarily allow any community name to have read-only access in the previous example, you would modify the lines in your NET.CFG file as follows:

```
desktop snmp
   enable monitor community any
   monitor community “private”
   enable control community specified
   control community “private”
   enable trap community specified
   trap community “agenttrap”
```

Desktop SNMP ignores the monitor community “private” line until you reset the enable monitor community line to “specified.” This allows you to temporarily change the value to “any” or “off” without deleting the specific community name.

MIB-II (Management Information Base) Support

Desktop SNMP automatically supports three MIB-II groups:

- System and SNMP groups
- Interface group
- TCP/IP groups (all groups except interface, system, and SNMP, which are supported by other VLM™ files, and EGP and transmission, which are not supported)

System and SNMP Groups

Use these parameters and values to define information that can be retrieved by SNMP management stations or reported in SNMP traps, which are discussed on the indicated pages:

Parameters and Values
SNMPENABLEAUTHENTRAP [on | off] SYSCONTACT “contact” SYSLOCATION “location” SYSNAME “name”

Desktop SNMP automatically supports two MIB-II groups, system and SNMP. These groups provide SNMP management stations with information about your workstation and about Desktop SNMP.

You can define some parameters for your workstation environment in the NET.CFG file (or other defined configuration file). The following table explains these parameters.

Table 2-4 **Desktop SNMP Option Parameters for MIB-II Support**

Parameter	Explanation
snmpenableauthentrap	The snmpenableauthentraps parameter defaults to “off.” When set to “on,” it instructs Desktop SNMP to send a trap message if someone without proper access tries to use SNMP to get or change information that Desktop SNMP manages.
syscontact	Use the real name of the person who should be contacted if your workstation needs maintenance.
syslocation	Use the physical location of your workstation.
sysname	Use your user or login name—or your TCP/IP host name, if one is assigned.

Each of these parameters is optional and can be used in groups or separately.

The first parameter enables the workstation to notify a network supervisor that an unauthorized user is trying to access the workstation without proper authority.

The last three parameters define information that can be retrieved by SNMP management stations or reported in SNMP traps.

Following is an example of a NET.CFG entry that notifies Suzanne Morley that an unauthorized user is tying to access the network:

```
desktop snmp
   sysname “Suzanne Morley x893”
   syslocation “Building 2”
   syscontact “suzanne@acompany.com”
   snmpenableauthentraps on
```

NOTE: Always enclose the name, location, or contact information in quotation marks.

SNMPENABLEAUTHENTRAP [on | off]

Instructs the Desktop SNMP to send a trap message if someone without proper access tries to use SNMP to get or change information that Desktop SNMP manages.

Syntax	snmpenableauthentrap [on | off]
Default	off

Example	To increase security on your workstation, you would enable the Desktop SNMP to send a trap message to the manager by placing the following lines in your NET.CFG file:
	
	`desktop snmp`
	`  snmpenableauthentrap on`

SYSCONTACT “contact”

Informs the SNMP manager of your workstation’s system administrator.

Syntax	syscontact “contact”
Default	None

Example	To notify the SNMP manager that the name of your workstation’s system administrator is “Bob Jones” at “x324,” you would place the following lines in your NET.CFG file:
	
	`desktop snmp`
	`  syscontact “Bob Jones x324”`

SYSLOCATION “location”

Informs the SNMP manager of the physical location of your workstation.

Syntax	syslocation “*location*”
Default	None

Example	To notify the SNMP manager that your workstation is in “Building 2,” you would place the following lines in your NET.CFG file:
	
	`desktop snmp`
	`  syslocation “Building 2”`

SYSNAME “name”

Informs the SNMP manager of your username.

Syntax	sysname “*name*”
Default	None

Example	To notify the SNMP manager of your username “Suzanne,” you would place the following lines in your NET.CFG file:
	
	`desktop snmp`
	`  sysname “Suzanne”`

Interface Group

Desktop SNMP provides a separate VLM file, MIB2IF.VLM, that supports the interface group. This group provides information about your network connection, such as the number of packets transmitted and received and whether it is Ethernet or token ring.

Add the following lines to the NET.CFG file:

```
netware dos requester
  vlm=mib2if.vlm
```

NOTE: Always place the vlm=mib2if.vlm line after the lines that load the Desktop SNMP VLM suite of files.

TCP/IP Groups

Desktop SNMP also provides a separate VLM file, MIB2PROT.VLM, that supports the TCP/IP groups (all groups except interface, system, and SNMP, which are supported by other VLM files, and EGP and transmission, which are not supported).

Load this VLM when you are using the TCP/IP stack from LAN WorkPlace® software.

Add the following lines to the NET.CFG file:

```
netware dos requester
   vlm=mib2prot.vlm
```

NOTE: Always place the vlm=mib2prot.vlm line after the lines that load the Desktop SNMP VLM suite of files.

Example of NET.CFG File Including Each Group Support

On a workstation where the NetWare DOS Requester™ and comprehensive MIB-II support are both loaded, the entire Netware DOS Requester and Desktop SNMP sections of your NET.CFG file might read as follows:

```
netware dos requester
   vlm=wssnmp.vlm
   vlm=wstrap.vlm
   vlm=wsreg.vlm
   vlm=wsasn1.vlm
   vlm=mib2if.vlm
   vlm=mib2prot.vlm
desktop snmp
   sysname “Suzanne Morley x893”
   syslocation “Building 2”
   syscontact “Suzanne@acompany.com”
   snmpenableauthentraps on
```

Link Driver Option

Use this option to specify the hardware and software configurations of the LAN drivers for each network board in your workstation.

The value settings you specify for each parameter with this option should match the hardware and software settings for your network board.

This option also allows you to set up the proper frame type and protocols for NetWare supported LAN drivers.

Available Parameters and Values for the Link Driver Option

This option has the following parameters and values.

Parameters and Values
ALTERNATE BUS ID name number DMA [#1 | #2] channel_number FRAME frame_type_name [addressing_mode] IRQ [#1 | #2] interrupt_request_number MAX FRAME SIZE number MEM [#1 | #2] hex_starting_address [hex_length] NODE ADDRESS hex_address [mode] PORT [#1 | #2] hex_starting_address [hex_number_of_ports] PROTOCOL “name” hex_protocol_ID frame_type SAPS number SLOT number

LINK DRIVER *driver_name*

Specifies the LAN driver you are making configurations for.

Syntax	link driver driver_name parameter_name, value Replace driver_name with the name of the LAN driver. Replace parameter_name with the name of the parameter you want to use. Replace value with the value setting which corresponds with the parameter name. (See Table 2-6 “List of ODI LANB Driver Names and Supported Network Boards” for more information.)
Default	None

Example	To configure an NE2000™ driver, you could type
	`link driver ne2000`
	`  int 5`
	`  port 360`
	`  frame ethernet_802.2`

ALTERNATE

Specifies an alternate network board. Normally, the LANSUP and NTR2000 drivers use the primary board.

Syntax	alternate
Default	None

Example	To specify the LANSUP.COM driver to use an alternate network board, you would place the following lines in your NET.CFG file:
	`link driver lansup`
	`  alternate`

BUS ID name number

Specifies the bus that the network board is inserted into. Use this parameter with LAN drivers that support multiple bus types.

The currently defined bus ID types and numbers are as follows:

Bus Type	Identification Number
ISA	0
MCA	1
EISA	2
PCMCIA	3
PCI	4
VL (VESA Local Bus)	5

NOTE: This is not a conclusive list of bus types. You can obtain an updated list of identifiers from the Novell Labs™ facility.

The default value indicates that the LAN driver should search each of the machine's buses for a supported network board. The LAN driver should then initialize the first supported network board it finds. The LAN driver determines the order that the buses are searched in.

Syntax	`bus id` *`name number`* Replace *name* with the name of the bus type you want to use. Replace *number* with the identification number that corresponds with the bus type that you want to use.
Default	-1 0FFh

Example	To specify the bus type for an ISA adapter, you would place the following lines in your NET.CFG file:
	
	`link driver ntr2000`
	`  bus id isa 0`

DMA [#1 | #2] channel_number

Specifies the hardware setting of the network board used in the workstation. It allows DMA channels to be configured..

Syntax	`dma [#1 | #2] channel_number` Enter the *channel number* to be used. You can specify two DMA values.
Default	#1

Example	If the first configurable DMA channel on your NE2100™ network board uses DMA channel 3 and the second configurable channel uses channel 4, you would place the following lines in your NET.CFG file:
	
	`link driver ne2100`
	`  dma #1 3`
	`  dma #2 4`

FRAME frame_type_name [addressing_mode]

Enables the frame types used by the LAN driver.

When the LAN driver initializes the network board, it creates a logical board for the frame type name that is specified. You can create multiple frame types concurrently on each client workstation.

This parameter also allows for configuration of the addressing mode for some LAN drivers on the basis of frame types. The following keywords determine the address mode:

- LSB canonical addressing mode
- MSB non-canonical addressing mode

For example, the frame type Token-Ring can be used in LSB.

Syntax	`frame frame_type_name [addressing_mod]` Replace *frame_type_name* with the supported frame type of the network you are connecting to. Replace *addressing_mode* with the address mode for your frame type.

Default	Ethernet LAN drivers: Ethernet_802.2 Token-ring LAN drivers: Token-Ring TCP/IP LAN driver SLIP_PPP: SLIP See Table 2-4 “List of Frame Types, Protocols, and LAN Drivers” for more information.

Example	To enable the Token-Ring frame type for an NTR2000 LAN driver, you would place the following lines in your NET.CFG file:
	`link driver ntr2000`
	`  token-ring lsb`
	To enable the Ethernet_II and Ethernet 802.2 frame type for an NE2000 LAN driver, you would place the following lines in your NET.CFG file (for two logical networks):
	`link driver ne2000`
	`  frame ethernet_ii`
	`  frame ethernet_802.2`

NOTE: The NetWare Client binds to the first frame type listed under the LINK DRIVER heading.

Frame Types, Protocols, and LAN Drivers

The following table shows the frame types currently defined by Novell and their respective protocols. Included is a list of the supporting LAN drivers for each frame type that currently ship with the NetWare Client Kit for DOS and MS Windows.

Table 2-5 List of Frame Types, Protocols, and LAN Drivers

Frame Type and Description	Protocols	LAN Drivers
ETHERNET_802.2 Ethernet (802.3) using an 802.2 envelope	IPX/SPX	3C1100, 3C501, 3C501, 3C503, 3C505, 3C523, 3C5X9, CEODI, E20ODI, E2HODI, E30ODI, E31ODI, ES3210, EXP16ODI, HPMCAODI, IBMFDDIO, IBMODISH, ILANAT, INTEL593, INTEL595, INTEL596, LANSUP, NCRWL05, NE1000, NE1500T, NE2, NE2_32, NE2000, NE2100, NE3200, NI5210, NI6510, NI9210, ODINSUP, PCMDM, PE2ODI, SMC8000, TCE16ATW, TCE16MCW, TCE32MCW, TCNSW, UBODI
ETHERNET_802.3 IPX 802.3 raw encapsulation	IPX/SPX	3C1100, 3C501, 3C501, 3C503, 3C505, 3C523, 3C5X9, CEODI, E20ODI, E2HODI, E30ODI, E31ODI, ES3210, EXP16ODI, HPMCAODI, IBMFDDIO, IBMODISH, ILANAT, INTEL593, INTEL595, INTEL596, NCRWL05, NE1000, NE1500T, NE2, NE2_32, NE2000, NE2100, NE3200, NI5210, NI6510, NI9210, ODINSUP, PCMDM, PE2ODI, SMC8000, TCE16ATW, TCE16MCW, TCE32MCW, TCNSW, UBODI
ETHERNET_II Ethernet using a DEC Ethernet II envelope	IP IPX/SPX	3C1100, 3C501, 3C501, 3C503, 3C505, 3C523, 3C5X9, CEODI, E20ODI, E2HODI, E30ODI, E31ODI, ES3210, EXOS, EXP16ODI, HPMCAODI, IBMFDDIO, IBMODISH, ILANAT, INTEL593, INTEL595, INTEL596, NCRWL05, NE1000, NE1500T, NE2, NE2_32, NE2000, NE2100, NE3200, NI5210, NI6510, NI9210, ODINSUP, PCMDM, PE2ODI, SMC8000, TCE16ATW, TCE16MCW, TCE32MCW, TCNSW, UBODI

Table 2-5 List of Frame Types, Protocols, and LAN Drivers

Frame Type and Description	Protocols	LAN Drivers
FDDI_802.2 FDDI using an 802.2 envelope	IPX/SPX	†
IP IP Tunnel frame envelope	IP IPX/SPX	†
Token-Ring Token ring (802.5) using an 802.2 envelope	IPX/SPX RPL SNA NetBIOS	LANSUP, MADGEODI, NTR2000, ODINSUP, OSH391R, OSH392R, OSH89XR, OSH990R, SMC8100, T20ODI, T30ODI, TCTOKSH, NTR2000

NOTE: † Currently, no LAN drivers supporting this frame type are available in the NetWare Client for DOS and MS Windows Client kit. Contact third-party vendors for details.

You can specify more than one frame type statement for a single driver.

For example, you can specify that an Ethernet NE2000 board can use both Ethernet_802.2 and Ethernet_802.3 frame types.

802.2 is the type of communications sent on one network, and 802.3 is the type of communication sent on the other network.

Ethernet LAN Drivers

For Ethernet, enable Ethernet_802.3, Ethernet_II, and Ethernet_802.2 frame types.

You can use up to four frame types for a single Ethernet cable segment.

You can use either four network boards each with one frame type defined, or you can use one network board with four frames defined, or any similar combination.

Token-Ring LAN Drivers

For token ring, enable token ring frame types.

IRQ [#1 | #2] interrupt_request_number

Specifies which interrupt (IRQ) the network board is set to use.

Syntax	irq [#1 | #2] *interrupt_request_number* Use the same interrupt request number that is set on the network board.
Default	#1 See your LAN driver documentation for the specific IRQ default values.

Example	To specify interrupt 5 on an NE2100 network board, you would place the following lines in your NET.CFG file:
	
	`link driver ne2100`
	`  irq 5`
	
	To specify more than one interrupt request number, you could place the following lines in your NET.CFG file:
	
	`link driver ne2100`
	`  irq #1 5`
	`  irq #2 3`

NOTE: The syntax for this parameter has changed from INT to IRQ. The INT syntax is still supported, but it is recommended that you update client workstations to use IRQ.

Before changing the value of the interrupt request number for your network board, be sure you know what interrupt value settings are used on other hardware (such as monitors) that you are using.

For example, interrupts 2 and 9 through 15 are usually reserved, so don't use those numbers (especially 2) for your network board.

We recommend using 3, 5, or 7 for most network boards.

MAX FRAME SIZE number

Sets the maximum number of bytes that can be put on the network by the NTR2000.COM and LANSUP.COM LAN drivers.

Syntax	max frame size *number*
	If the line speed is 16 Mbps, the value for *number* must be between 638 and 17,954. If the line speed is 4 Mbps, the value must be between 638 and 4464. The value must include the number of bytes for the data packet (usually 1, 2, 4, or 8KB), and for the largest possible header (currently, 52 bytes LAN header + 74 bytes protocol header = 126 bytes). For example, to use 2KB data packets, you would calculate the value for *number* as 2048 + 52 + 74 = 2174
Default	The NTR2000.COM driver's default size is 4222 bytes. But if your network board has 8 KB of shared RAM available, the default size is 2174 bytes. The LANSUP.COM driver's default is 1150 bytes. But if you're using the IBM LAN Support Program with an Ethernet network board driver, the maximum size is 1496 bytes.

Example	To specify the maximum frame size of an NTR2000 LAN driver, you would place the following line in your NET.CFG file:
	
	`link driver ntr2000`
	`  max frame size 2174`

If you are running MS Windows in enhanced mode, the VIPX.386 device driver requires the maximum frame size to be less than 8000 bytes. If the frame size exceeds 8000 bytes, network communication fails.

MEM [#1 | #2] hex_starting_address [hex_length]

Specifies a memory range to be used by the network board.

Syntax	mem [#1 | #2] *hex_starting_address* [*hex_length*]

	Replace *hex_starting_address* with the hexadecimal physical (absolute) address of the memory used by the network board.
	Replace the *hex_length* with the memory address range (in hexadecimal paragraphs, with each paragraph being 16 bytes) used by the network board. Usually, the hex length is not needed.
Default	#1

Example	When the node address on a network board is set from D0000 to D4000 (bytes), the starting address is D0000 and the range is 400 hexadecimal paragraphs.
	
	Therefore, for an NE2000 network board, you would place the following lines in your NET.CFG file:
	
	`link driver ne2000`
	`  mem d0000 400`

Assign each board a unique memory range.

Be sure you don't assign a range that is already used by other hardware. (VGA monitors commonly use A0000-C6FFF and XVGA monitors commonly use A0000-CFFFF.)

NODE ADDRESS hex_address [mode]

Overrides the hard-coded node address in the LAN driver, if the hardware allows it. The new address is used to program the network board.

Changing the node address for a network board can create conflicts with other LAN drivers. Use the hard-coded node address whenever possible.

You can use either the canonical or non-canonical mode. Use the following keywords to determine the mode:

- LSB (canonical addressing mode)
- MSB (non-canonical addressing mode)

For example, node addresses for both modes could appear as follows:

- node address 0800005A656BL (canonical LSB)
- node address 1000005AA6D6M (non-canonical MSB)

NOTE: If the "M" or "L" is not specified, the default mode for the node address is the physical layer form of the address.

Syntax	node address *hex_address* [*mode*]
Default	None

Example	To change the node address of an NTR2000 LAN driver to "1000005aa6d6" for non-canonical mode, you would place the following lines in your NET.CFG file:
	
	`link driver ntr2000`
	`  node address 1000005aa6d6m`

NOTE: Even though FDDI is a non-canonical topology at the physical layer, all address are sent and received in canonical (LSB) mode.

A node labeled with an "M" for non-canonical mode address supports media setup for canonical mode, because the LAN driver simply swaps the provided address to obtain the appropriate canonical (LSB) address.

LANSUP

This parameter allows the LANSUP.COM file set to the frame size for a token-ring driver.

Syntax	open
Default	None

Example	To allow the RPL protocol to increase the frame size for a remote boot workstation running the LANSUP software above 1 KB, you would place the following lines in your NET.CFG file:
	
	`link driver lansup`
	`  open`

If the IBM LAN Support driver's DXMT0MOD.SYS file is to load in the CONFIG.SYS file, one of two actions must be performed:

1 Set "o=n" on the command line for loading the DXMT0MOD.SYS file in the CONFIG.SYS file and use this parameter in the NET.CFG file.

2 Set "o=y" on the command line to load the DXMT0MOD.SYS file and set the frame size for the NDIS driver. Do not use this parameter in the NET.CFG file.

PORT [#1 | #2] hex_starting_address [hex_number_of_ports]

Specifies the starting port (hex_starting_address) and number of ports in the range (hex_number_of_ports).

Syntax	port [#1 | #2] *hex_starting_address* [*hex_number_of_ports*] All values must be written in hexadecimal notation.
Default	#1

Example	If the starting I/O port on an NE2100 LAN driver is 300 and 16 ports are in the first range, you would place the following lines in your NET.CFG file:
	
	`link driver ne2100`
	`  port 300 16`
	
	To specify two ranges with 32 ports in each range, you could place the following lines in your NET.CFG file:
	
	`link driver ne2100`
	`  port #1 300 32`
	`  port #2 340 32`

PROTOCOL “name” hex_protocol_ID frame_type

Allows existing LAN drivers to handle new network protocols.

Syntax	protocol name hex_protocol_ID frame_type Replace name with the name of the new protocol. Replace hex_protocol_ID with the assigned hexadecimal protocol ID that the protocol is assigned. Replace frame_type with the frame type that the new protocol ID applies to.Table 5-6 for more information.
Default	None

Example	To use a new protocol named “XYZ” with an NE/2-32™ LAN driver using the Ethernet_II frame type, you would place the following lines in your NET.CFG file:
	
	`link driver ne2_32`
	`  frame ethernet_II`
	`  protocol xyz 904a ethernet_II`

Defined Protocols and Frame Types

The following table lists the protocols and frame types currently defined within the industry and their corresponding protocol ID and frame ID numbers.

Table 2-6 List and Description of Protocols and Frame Types with Their Identification Numbers

Frame ID	Frame Type	Protocol	Protocol ID Number	Description
2	ETHERNET_II	IPX/SPX	8137h	Ethernet using a DEC* Ethernet II envelope
2	ETHERNET_II	IP	800h	Ethernet using a DEC Ethernet II envelope
3	ETHERNET_802.2	IPX/SPX	E0h	Ethernet (802.3) using an 802.2 envelope
4	Token-Ring	IPX/SPX	E0h	Token ring (802.5) using an 802.2 envelope
5	ETHERNET_802.3	IPX/SPX	00h	IPX 802.3 raw encapsulation
19	IP	IPX/SPX	N/A	IP Tunnel frame envelope
20	FDDI_802.2	IPX/SPX	E0h	FDDI using an 802.2 envelope

SLOT number

In slot-based machines, the LAN driver usually locates the network board by scanning through the slots in order from lowest to highest.

This option directs the driver to locate the network board in the specified slot, instead of scanning for it.

Syntax	slot *number* Use the number of the slot you inserted the network board into. The slot number is usually found on the back of the computer.
Default	None

Example	If you were using two NE/2™ boards in the same workstation and you inserted one network board into slot 1 and the other into slot 2, you would place the following lines in your NET.CFG file:
	
	`link driver NE2`
	`  slot 1`
	`link driver NE2`
	`  slot 2`

Listing of Commonly Used ODI LAN Drivers

The following table lists some of the common ODI™ LAN drivers used with NetWare Client™ software for DOS and MS Windows.

This is not a comprehensive list of all LAN drivers certified by Novell. See the documentation included with your network board for information concerning the appropriate LAN driver to use with the NetWare operating system.

The table also includes a list of ODI LAN driver names and supported network boards. for information on the available parameters for each driver, see the .INS file accompanying each LAN driver.

Driver Name (.COM)	Network Board
3C501	3Com* 3C1100 3Station* Ethernet
3C501	3Com 3C501 EtherLink*
3C503	3Com 3C503 EtherLink II* 3Com 3C503 EtherLink II TP 3Com 3C503 EtherLink II/163C 3Com 3C503 EtherLink II/16 TP
3C505	3Com 3C505 EtherLink Plus*
3C523	3Com EtherLink/MC 3Com EtherLink/MC TP
3C5X9	3Com EtherLink III* Parallel Tasking Family

Driver Name (.COM)	Network Board
CEODI	Xircom Credit Card Ethernet Adapter
E20ODI	Cabletron Systems Ethernet E20
E2HODI	Cabletron Systems Ethernet E2HUB
E30ODI	Cabletron Systems Ethernet E3010 Cabletron Systems Ethernet E3010-X Cabletron Systems Ethernet E3020 Cabletron Systems Ethernet E3020-X Cabletron Systems Ethernet E3030 Cabletron Systems Ethernet E3030-X Cabletron Systems Ethernet E3040E Cabletron Systems Ethernet E3040-X
E31ODI	Cabletron Systems Ethernet E31 Cabletron Systems Ethernet E31-X
ES3210	Racal-Datacom* ES3210
EXP16ODI	Intel EtherExpress* ISA Family Intel EtherExpress MCA Family
HPMCAODI	Hewlett-Packard* MC Adapter 16
HPISAODI.COM	Hewlett-Packard PC Adapter 8/16/16+
HP32ODI.COM	HP* Ethertwist LAN Adapter/32
AM2100.COM	HP PC LAN Adapter NC/16 TP
IBMFDDIO	IBM FDDI
IBMODISH	IBM PS/2 Ethernet Adapter
ILANAT	Racal-Datacom InterLan AT/XT
INTEL593	Intel 593 Based Adapter Zenith Data Systems Z-NOTE
INTEL595	Intel 82595 Based Adapter
INTEL596	Intel 596 Based Adapter

Driver Name (.COM)	Network Board
LANSUP	ODI Module for the IBM LAN Support Program
MADGEODI	Madge Smart 16/4 PC Ringnode Madge Smart 16/4 AT Ringnode Madge Smart 16/4 MC Ringnode Madge Smart 16/4 MC32 Ringnode Madge Smart 16/4 EISA Ringnode Madge Smart 16/4 EISA Mk II Ringnode
NCRWL05	NCR* WaveLAN*
NE1000	Novell Ethernet NE1000
NE1500T	Novell Ethernet NE1500TN
NE2	Novell Ethernet NE/2
NE2_32	Novell Ethernet NE2-32
NE2000	Novell Ethernet NE2000 National Semiconductor NE2000 InfoMover
NE2100	Novell Ethernet NE2100
NE3200	Novell Ethernet NE3200
NE3300.COM	Microdyne NE3300 Ethernet
NI5210	Racal-Datacom NI5210
NI6510	Racal-Datacom NI6510
NI9210	Racal-Datacom NI9210
NTR2000	Novell NTR2000 Token-Ring Adapter IBM Token-Ring Network Adapter II IBM Token-Ring Network 16/4 Adapter IBM Token-Ring Network Adapter /A IBM Token-Ring Network 16/4 Adapter /A IBM Token-Ring 16/4 Credit Card Adapter compatible

Driver Name (.COM)	Network Board
NULLDRV	This is not a working LAN driver. The NetWare client installation program copies this file to the client directory during installation for dedicated (not ODI) IPX drivers. Contact the network board manufacturer for copies of the latest ODI drivers. This LAN driver can also be used to support the Personal NetWare™ client and server software running on a workstation without a network board installed.
OC32TR16.COM	Olicom EISA 32 Token-Ring 16/4 Adapter
OCTOK16.COM	Olicom Token-Ring MC 16/4 Adapter
OSH391R	Proteon p1391 RapiDriver
OSH89XR	Proteon p189X RapiDriver
OSH990R	Proteon p1990 RapiDriver
ODINSUP	ODI Module for the NDIS protocol stack
PCMDM	NSC Ethernet PCMCIA Card
PE2ODI	Xircom Credit Card Ethernet Adapter
SLIP_PPP	Novell LAN WorkPlace® Asynchronous SLIP & PPP Driver This driver runs on workstations serial hardware with either PPP or SLIP frame types. It supports both direct-connection line and indirect modem connections.
SMC8000	SMC* EtherCard* PLUS* Family Adapter
SMC8100	SMC Token Ring Elite Family Adapter
SMC9000.COM	Standard Microsystems Corporation SMC9000

Driver Name (.COM)	Network Board
SMCARCWS	SMC PC130/130E/270E SMC PC500WS/550WS (short or long card) SMC PC600WS/650WS SMC PS110//210/310 PS/2
T20ODI	Cabletron Systems Token-Ring T20
T30ODI	Cabletron Systems Token-Ring T30
TCCARC	Thomas-Conrad TC6x42 ARCnet 8-bit Adapter Thomas-Conrad TC6x45 ARCnet 16-bit Adapter Thomas-Conrad TC6x46 ARCnet MC Adapter
TCE16ATW	Thomas-Conrad TC5045 Ethernet Adapter
TCE16MCW	Thomas-Conrad TC5046 Ethernet Adapter - 16 bit
TCE32MCW	Thomas-Conrad TC5046 Ethernet Adapter - 32 bit
TCNSW	Thomas-Conrad TC3042 TCNS 8-bit Adapter Thomas-Conrad TC3045 TCNS 16-bit Adapter Thomas-Conrad TC3046 TCNS MC Adapter Thomas-Conrad TC3047 TCNS EISA Adapter
TCTOKSH	Thomas-Conrad TC4035/TC4045 Adapter Thomas-Conrad TC4046 Adapter
TRXNET	Novell RX-Net & RX-Net II® Novell RX-Net/2T™
UBODI	Ungermann-Bass* NIUpc/EOTP Ungermann-Bass NIUps Ungermann-Bass NIUpc Ungermann-Bass Personal NIU/ex Ungermann-Bass Personal NIU

Link Support Option

Use this option of the NET.CFG file to configure the number and size of the receive buffers, the size of the memory pool buffers, and the number of boards and stacks used by the Link Support Layer™ (LSL).

Available Parameters and Values for the Link Support Option

This option has the following parameters and values, which are discussed on the following pages.

Parameters and Values
BUFFERS communication_number [buffer_size] MAX BOARDS number MAX STACKS number MEMPOOL number [k]

LINK SUPPORT

Determines the number and size of the receive buffers, the size of the memory pool buffers, and the number of boards and stacks used by the LSL™ program.

Syntax	link support *parameter_name value* Replace *parameter_name* with the name of the parameter you want to use. Replace *value* with the value setting which corresponds with the parameter name.

Example	To configure for four network boards, you would place the following lines in your NET.CFG file:
	
	`link support`
	`  max boards 4`

BUFFERS communication_number [buffer_size]

Configures the number and size of receive buffers that the Link Support Layer (LSL) manages.

Communication Number The number of communication buffers must be large enough to hold all protocol headers and the maximum data size. If you make many connections, you should increase the number of buffers to increase performance.

If the protocol you are using builds the media header (MAC) for the LAN driver, then you need to set the buffer size large enough to manage the header also.

For example, IPX requires 512 bytes (LAN header) + 74 bytes (protocol header) + 52 bytes (MAC) = 638 bytes.

See the manufacturer's specifications for possible parameters and values required for third-party protocol stacks.

Buffer Size The total buffer size must share a 64KB segment with the mempool and resident program code from the LSL.COM file when loaded into memory (approximately 5 KB).

This means that the communication number multiplied by the buffer size (plus the code size and mempool) cannot be greater than 65,536 bytes.

For example, 20 communication buffers multiplied by a buffer size of 1514 bytes equals 30,280 bytes.

Syntax	buffers *communication_number* [*buffer_size*] Replace *communication_number* with a number of buffers greater than 1. Replace *buffer_size* with a number of bytes greater than 638. The buffer size is optional.

Default	For IPX: *communication_number:* The IPX protocol utilizes its own buffers and does not require the Link Support Layer software to provide buffers for it. *buffer_size:* 1500 bytes For TCP / IP: *communication_number:* 8 buffers *buffer_size:* 1500 bytes TCP / IP requires only 1500 bytes, because the LAN driver places the 14 bytes of media header on the frame, prior to transmission. For LSL: *buffer_size:* 1514 bytes This accommodates any protocol that utilizes raw sends on Ethernet.

Example	To ensure that the communication buffers are large enough to hold all media headers, you could place the following lines in your NET.CFG file:
	
	[For an Ethernet configuration]
	
	`link support`
	`  buffers 8 1514`
	
	[For a token-ring configuration]
	
	`link support`
	`  buffers 8 4222`

NOTE: For the most efficient communication, your link support buffer size should be the same size as the packets that your workstation receives over the network. You might want to set the link support buffer size equal to the largest buffer size that the network boards in your workstation supports.

MAX BOARDS number

Configures the maximum number of logical boards that the LSL.COM file can manage.

Each LAN driver can use more than one board resources.

If a LAN driver fails to load because of an out-of-resource condition, increase the value for this option.

The amount of resident memory the LSL.COM file consumes increases with larger MAX BOARD values and decreases with smaller values. Thus, you can conserve some memory by reducing the value to the actual number of boards used by a particular system.

For example, if you had the NE2000.COM file configured to load all possible Ethernet frame types (ETHERNET_II, ETHERNET_802.3, ETHERNET_802.2, and ETHERNET_SNAP), four network board resources would be used. Therefore, to load all four frame types, MAX BOARDS must be set to a value of 4 or higher.

Syntax	max boards *number*
Default	4
Range	1 to 16

Example	To specify that two boards can be loaded, you would place the following lines in your NET.CFG file:
	
	`link support`
	`  max boards 2`

NOTE: Each LAN driver resource or frame type requires approximately 200 bytes of memory.

MAX STACKS number

Configures the maximum number of logical protocol stack IDs the LSL.COM file can manage.

Each protocol stack can use more than one stack ID resources.

If a protocol stack fails to load because of an out-of-resource condition, increase the value for this option.

The amount of resident memory the LSL.COM file consumes increases with larger MAX STACKS values and decreases with smaller values. Thus, you can conserve some memory by reducing the value to the actual number of stack IDs used by a particular system.

Syntax	max stacks *number*
Default	4
Range	1 to 16

Example	To specify that three logical protocol stack IDs can be loaded, you would place the following lines in your NET.CFG file:
	
	`link support`
	`  max stacks 3`

MEMPOOL number [k]

Configures the size of the memory pool that the LSL program maintains for allocating buffers for some protocols.

Thus, the notation means to multiply by 1024.

Syntax	mempool *number* [k]
Default	0

Example	To configure the size of the memory pool, you could place the following lines in your NET.CFG file:
	
	`link support`
	`  mempool 1024`

NOTE: The IPXODI protocol stack does not use memory pool buffers.

NetWare DOS Requester Option

Use this option of the NET.CFG file to configure the NetWare Client software for various types of networking support and connectivity.

The NetWare DOS Requester software provides networking support for DOS and Microsoft (MS) Windows client workstations.

The NetWare DOS Requester is built on an architecture of Virtual Loadable Module (VLM) software. The VLM.EXE (VLM manager) file is responsible for loading the required modules.

Previous versions of the NetWare DOS and MS Windows client software relied on the single shell file NETX.EXE to provide networking support.

NOTE: The NetWare DOS Requester is not compatible with the NETX.COM or NETX.EXE file. Use the NETX.VLM file for compatibility with shell calls.

The following topics are covered in this section.

Topic
Current Core Virtual Loadable Module (VLM) Programs
Current Non-Core Virtual Loadable Module Programs
Compatibility with NetWare Shell Parameters
Managing the NetWare DOS Requester
Optimizing the NetWare DOS Requester
Available Parameters and Values for the NetWare DOS Requester Option

Current Core Virtual Loadable Module (VLM) Programs

The following table lists the current core VLM programs in their default load order.

The table also includes descriptions and flags indicating whether the module is required (R) or optional (O) for NetWare Directory Services™ (NDS) support, bindery services (BIND), and Personal NetWare™ (PNW) services.

Table 2-7 List of Current Core VLM Programs and the Respective Load Order

Module Name	Description	NDS	BIND	PNW
BIND.VLM	NetWare protocol implementation using the bindery	O	R	O
CONN.VLM	Connection table manager	R	R	R
FIO.VLM	File Input/Output	R	R	R
GENERAL.VLM	Miscellaneous functions for the NETX.VLM and REDIR.VLM files	R	R	R
IPXNCP.VLM	Transport protocol implementation using IPX	R	R	R
NDS.VLM	NetWare protocol implementation using NDS™ support	R	O	O
NETX.VLM	NetWare shell compatibility	O	O	O
NWP.VLM	NetWare protocol multiplexor	R	R	R
PNW.VLM	NetWare protocol implementation using Personal NetWare	O	O	R
PRINT.VLM	Printer redirector	O	O	O
REDIR.VLM	DOS redirector	R	R	R
SECURITY.VLM	NetWare enhanced security	O	O	O
TRAN.VLM	Transport protocol multiplexor	R	R	R

Current Non-Core Virtual Loadable Module Programs

The following table lists the current non-core VLM programs. Non-core VLM programs do not automatically load by default.

Use the VLM path_vlm parameter and value to load any non-core modules. for details on how to use this parameter and value.

The table also includes descriptions and flags indicating whether the module is used (U) or not used (N) for NetWare Directory Services (NDS), bindery services (BIND), and Personal NetWare (PNW) services.

Table 2-8 List of Current Non-Core VLM Programs

Module name	Description	NDS	BIND	PNW
AUTO.VLM	Auto-reconnect/auto-retry	U	U	U
MIB2IF.VLM	MIB-II interface groups support	U	U	U
MIB2PROT.VLM	MIB-II support for the TCP/IP groups	U	U	U
NMR.VLM	NetWare management responder	U	U	U
RSA.VLM	RSA encryption for NetWare Directory Services re-authentication	U	U	U
WSASN1.VLM	SNMP ASN.1 translation	U	U	U
WSDRVPRN.VLM	The print information collection file for gathering information about print mappings and captured printers	U	U	U
WSREG.VLM	SNMP MIB registration	U	U	U
WSSNMP.VLM	Desktop SNMP module, which includes support for MIB-II System and SNMP groups	U	U	U
WSTRAP.VLM	SNMP trap	U	U	U

Compatibility with NetWare Shell Parameters

† Parameter is supported under the NetWare DOS Requester heading in the NET.CFG file. See specific reference to the parameter for more information.

¤ No longer supported. The functionality of this parameter was either developed into the NetWare DOS Requester software or is no longer required because of the NetWare DOS Requester architecture. Existing lines in your NET.CFG file for these parameters will be ignored.

Parameters and Values	Status
all servers=[on | off]	¤
cache buffers=number	†
checksum=[on | off] (Now requires a number value from 0 to 3.)	¤
dos name=name	†
entry stack size=number	¤
environment pad=number	¤
eojs=[on | off]	†
file handles=number (Functionality is provided by the FILES parameter setting in the CONFIG.SYS file.)	¤
get local target stacks=number	¤
hold=[on | off]	¤
lipackets=[on | off] (Functionality is provided by LARGE INTERNET PACKETS.)	¤
local printers=number	†
lock delay=number	†
lock retries=number	†
long machine type=name	†
max cur dir length=number (Because of a limitation in DOS, the maximum character length is hard set to a value of 63 characters.)	¤
max path length=number	¤
max tasks=number	†

Parameters and Values	Status
patch=byte_offset_value	¤
pb buffers=number	†
preferred server=server_name	†
print header=number	†
print tail=number	†
read only compatibilitys=[on | off]	†
search mode=number	†
set station times=[on | off]	†
share=[on | off]	¤
short machine type=name	†
show dots=[on | off]	†
sign 386 mode	¤
signature level=number	†
special uppercases=[on | off]	¤
task mode=number	¤

Managing the NetWare DOS Requester

Use the following information to manage loading and using the VLM.EXE file and its modules under specific conditions.

Condition	Explanation
Loading the VLM.EXE file from a directory other than your current directory	The current directory is used for VLM software. To load the VLM software from another directory, use the “VLM =” command in the NET.CFG file. For example: VLM=C:\NWCLIENT\CONN.VLM

Condition	Explanation
Specifying a NET.CFG file outside the current directory	At the command line, enter a command similar to the following (or add the command to the AUTOEXEC.BAT file): VLM /C=C:\NWCLIENT\NET.CFG
Loading the NetWare DOS Requester memory managers under MS Windows 3.0	If you are running MS Windows 3.0, ensure that the VLM.EXE file is loaded after any upper- or high-memory managers are loaded.
Avoid Loading VLM software in expanded memory with MS Windows	Don't use the expanded memory option (/ME). Run MS Windows with the NetWare DOS Requester only if you use the extended memory option (/MX, preferred) or the conventional memory option (/MC).
Disabling the VLM software	The following are the preferred ways to disable specific VLM software: • Rename the module with an extension other than .VLM • Use the parameter and value. See "USE DEFAULTS=[on | off]" on page 165 • Use the parameter and value. See "EXCLUDE VLM=path_vlm"

Optimizing the NetWare DOS Requester

You can optimize the NetWare DOS Requester software for the following three conditions:

- Best Performance
- Best Conventional Memory Usage
- Best Compromise

Best Performance

For the best performance, use the following parameters under the NetWare DOS Requester option in the NET.CFG file.

AUTO LARGE TABLE=[on | off]	LOAD LOW CONN=[on | off]
AUTO RECONNECT=[on | off]	LOAD LOW IPXNCP=[on | off]
AVERAGE NAME LENGTH=number	MINIMUM TIME TO NET=number
CACHE BUFFER SIZE=number	NETWARE PROTOCOL=netware_protocol_list
CACHE BUFFERS=number	PB BUFFERS=number
CACHE WRITES=[on | off]	PRINT BUFFER SIZE=number
CHECKSUM=number	SIGNATURE LEVEL=number
LARGE INTERNET PACKETS=[on | off]	TRUE COMMIT=[on | off]

NOTE: Use the /MC command line parameter when loading the VLM.EXE file. /MC loads all of the VLM programs into conventional memory.

Loading the NetWare DOS Requester into conventional memory requires approximately 100 KB of memory.

Following is a sample NET.CFG file entry for a client workstation:

```
netware dos requester
   cache buffers=64
   cache buffer size=1540
   cache writes=on
   checksum=0
   exclude=nds.vlm
   large internet packet=on
   load low conn=on
   load low ipxncp=on
   minimum time to net=0
   netware protocol=bind,nds,pnw
   pb buffers=3
   print buffer size=256
   signature level=0
   true commit=off
```

This entry indicates the following:

- The CACHE BUFFER SIZE is set for an Ethernet network board
- The "DOS FILES=" command in the CONFIG.SYS file is set above 64 files
- The servers are running NetWare 3.12

Best Conventional Memory Usage

For the best memory usage, use the following parameters under the NetWare DOS Requester option in the NET.CFG file.

Table 2-9 Parameters and Values Used to Configure for Best Memory Usage

AUTO LARGE TABLE=[on | off]	LOAD LOW CONN=[on | off]
AUTO RECONNECT=[on | off]	LOAD LOW IPXNCP=[on | off]
AVERAGE NAME LENGTH=number	NETWORK PRINTERS=number
CACHE BUFFERS=number	PB BUFFERS=number
CONNECTIONS=number	PRINT HEADER=number
EXCLUDE VLM=path_vlm	PRINT TAIL=number
LARGE INTERNET PACKETS=[on | off]	SIGNATURE LEVEL=number

NOTE: Configuring for the smallest conventional memory usage will result in some performance degradation.

Following is a sample NET.CFG file entry for a client workstation:

```
netware dos requester
   auto large table=off
   auto reconnect=off
   average name length=7
   cache buffers=0
   connections=4
   exclude=pnw.vlm
   exclude=netx.vlm
   load low conn=off
   load low ipxncp=off
```

```
   network printers=0
   pb buffers=0
   print header=0
   print tail=0
   signature level=0
```

This entry indicates the following:

- The sum of character length of server names attached to is 24
- You are not running any NETX applications
- You print to a local printer only
- The servers are running NetWare 4 only (no bindery or Personal NetWare connections)

Best Compromise

The default settings for the NetWare DOS Requester provide the best compromise between performance and memory usage.

You should use the default settings until you can determine the optimal configuration for your particular network and client workstations.

Following is a sample working NET.CFG file for a client workstation. Wherever feasible, default settings are used. Otherwise, example variable replacements are used (for example, the preferred server name):

```
link driver ne2000
   port 300
   int 3
   frame ethernet_802.2
netware dos requester
   netware protocol=nds,bind
   first network drive=f
   preferred server=sales
   auto retry=10
   bind reconnect=on
   cache buffers=40
   checksum=0
   connections=16
```

```
large internet packet=on
load low conn=on
load low ipxncp=on
message timeout=183
pb buffers=8
signature level=0
pburst read window size=64
pburst write window size=64
```

Available Parameters and Values for the NetWare DOS Requester Option

Parameters and Values
AUTO LARGE TABLE=[on | off]
AUTO RECONNECT=[on | off]
AUTO RETRY=number
AVERAGE NAME LENGTH=number
BIND RECONNECT=[on | off]
BROADCAST RETRIES=number
BROADCAST SEND DELAY=number
BROADCAST TIMEOUT=number
CACHE BUFFER SIZE=number
CACHE BUFFERS=number
CACHE WRITES=[on | off]
CHECKSUM=number
CONFIRM CRITICAL ERROR ACTION=[on | off]
CONNECTIONS=number
DOS NAME=”name”
EOJ=[on | off]
EXCLUDE VLM=path_vim
FIRST NETWORK DRIVE=drive_letter
FORCE FIRST NETWORK DRIVE=[on | off]
HANDLE NET ERRORS=[on | off]
LARGE INTERNET PACKETS=[on | off]
LIP START SIZE=number
LOAD CONN TABLE LOW=[on | off]
LOAD LOW CONN=[on | off]
LOAD LOW IPXNCP=[on | off]

Parameters and Values
LOAD LOW REDIR=[on | off]
LOCAL PRINTERS=number
LOCK DELAY=number
LOCK RETRIES=number
LONG MACHINE TYPE="name"
MAX TASKS=number
MESSAGE LEVEL=number
MESSAGE TIMEOUT=number
MINIMUM TIME TO NET=number
NAME CONTEXT="name_context"
NETWARE PROTOCOL=netware_protocol_list
NETWORK PRINTERS=number
PB BUFFERS=number
PBURST READ WINDOWS SIZE=number
PBURST WRITE WINDOWS SIZE=number
PREFERRED SERVER="server_name"
PREFERRED TREE="tree_name"
PREFERRED WORKGROUP="workgroup_name"
PRINT BUFFER SIZE=number
PRINT HEADER=number
PRINT TAIL=number
READ ONLY COMPATIBILITY=[on | off]
RESPONDER=[on | off]
SEARCH MODE=number
SET STATION TIME=[on | off]
SHORT MACHINE TYPE="name"
SHOW DOTS=[on | off]
SIGNATURE LEVEL=number
TRUE COMMIT=[on | off]
USE DEFAULTS=[on | off]
VLM=path_VLM
WORKGROUP NET=workgroup_net_address

This option has the following parameters and values, which are discussed on the indicated pages:

NETWARE DOS REQUESTER

Controls network requests from your client workstation to a NetWare server or network.

Syntax	netware dos requester *parameter_name=value* Replace *parameter_name* with the name of the parameter you want to use. Replace *value* with the value setting which corresponds with the parameter name.

Example	To configure for four server connections, you would place these lines in your NET.CFG file:
	
	`netware dos requester`
	`  connections=4`

AUTO LARGE TABLE=[on | off]

Allocates a small table of 34 bytes per connection for bindery reconnects. When the parameter is enabled, the AUTO.VLM program allocates a large connection table of 178 bytes per connection for bindery reconnects.

Change the default value to “on” if the length of your username and password combined is more than 16 characters.

Syntax	auto large table=[on | off]
Default	off
Modules	AUTO.VLM

Example	To use a username and password longer than 16 characters combined, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  auto large table=on`

Use this parameter for binder connections if your username and password total more than 32 characters. For this parameter to work, also set BIND RECONNECT=ON.

AUTO RECONNECT=[on | off]

When this parameter is set to "on," the AUTO.VLM program reconnects a client workstation to a NetWare server and rebuilds the workstation's environment (excluding file-specific items) prior to connection loss.

When this parameter is set to "off," the AUTO.VLM program load fails at pre-initialization time.

Syntax	**`auto reconnect`**=[on | off]
Default	on
Modules	AUTO.VLM, NDS.VLM

Example	To make reconnecting to a NetWare server the user's responsibility, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  auto reconnect=off`

To enable the VLM.EXE program to load the AUTO.VLM file, you must include the VLM=AUTO.VLM parameter and value. You must also load RSA.VLM if you are re-establishing a NetWare Directory Services (NDS) connection, by using the VLM=RSA.VLM parameter.

AUTO RETRY=number

Sets the number of seconds the AUTO.VLM program waits before attempting a retry after receiving a network critical error.

Syntax	auto retry=*number*
Default	0
Range	0 to 3640
Modules	AUTO.VLM

Example	To make the AUTO.VLM program wait for one minute before retrying a connection, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  auto retry=60`

To enable the VLM.EXE program to load the AUTO.VLM file, you must include the VLM=AUTO.VLM parameter and value. You must also load RSA.VLM if you are re-establishing a NetWare Directory Services (NDS) connection, by using the VLM=RSA.VLM parameter.

NOTE: When the value for this parameter is set to 0, the AUTO.VLM program makes no retry attempts.

AVERAGE NAME LENGTH=number

Allows the NetWare DOS Requester to set aside space for a table of NetWare server names based on the values set for the AVERAGE NAME LENGTH and CONNECTIONS parameters.

For shorter NetWare server names, you can save some memory by setting the average length to a lower number.

Syntax	`average name length=`*`number`* Replace *number* with the sum of the average character length of your network server names, divided by the number set in the CONNECTIONS parameter plus 1.
Default	48
Range	2 to 48
Modules	CONN.VLM

Example	To save memory by shortening the average name length allowed to eight characters, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  average name length=9`

BIND RECONNECT=[on | off]

Automatically rebuilds bindery connections and restores drive mappings and printer connections.

Syntax	bind reconnect=[on | off]
Default	off
Modules	AUTO.VLM, BIND.VLM

Example	To make reconnecting to bindery services automatic, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  bind reconnect=on`

To enable the VLM.EXE program to load the AUTO.VLM file, you must include the VLM=AUTO.VLM parameter and value.

For this parameter to work, also set AUTO RECONNECT=ON.

BROADCAST RETRIES=number

Sets the number of times the NetWare DOS Requester software broadcasts a request.

Set this to a higher number if you are not seeing some resources in various utility lists.

NOTE: Increasing this number decreases the response time of some utilities.

Syntax	broadcast retries=*number*
Default	3
Range	1 to 255
Modules	PNW.VLM

Example	To enable a workgroup to display in the WGLIST utility that is not reachable by using the default value, you could place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  broadcast retries=10`

BROADCAST SEND DELAY=number

Sets the number of ticks the NetWare DOS Requester software waits between performing any function.

Set this to a higher number if you are using slow links such as modem or serial links on your network.

NOTE: Increasing this number decreases the response time of some utilities.

Syntax	broadcast send delay=*number*
Default	0
Range	0 to 255
Modules	PNW.VLM

Example	To ensure that utilities find resources connected by a serial cable, you could place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  broadcast send delay=10`

BROADCAST TIMEOUT=number

Sets the number of ticks the NetWare DOS Requester software waits between broadcast retries.

Set this to a higher number if you are trying to see resources physically located at a long distance from your workstation.

NOTE: Increasing this number decreases the response time of some utilities.

Syntax	broadcast timeout=*number*
Default	2
Range	1 to 255
Modules	PNW.VLM

Example	To enable a workgroup to display in the WGLIST utility that is unreachable by using the default value, you could place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  broadcast timeout=60`

CACHE BUFFER SIZE=number

Sets the buffer size (in bytes) for the cache buffers that the FIO module uses.

Increasing the value of the parameter allows you to cache larger amounts of data, thereby increasing performance and also memory use.

Syntax	cache buffers size=*number*
Default	Media maximum minus 64 bytes (For example, Ethernet: 1500 - 64 = 1436) See the manufacturer's documentation for the cache buffer size for your media type.
Range	64 to 4096 bytes
Modules	FIO.VLM

Example	To cache larger amounts of data than the default allows, you could place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  cache buffers size=1024`

WARNING: **When specifying the value for this parameter, don't exceed the maximum packet size of the network board. Doing so might cause system failure.**

CACHE BUFFERS=number

Sets the number of cache buffers the NetWare DOS Requester software uses for local caching of nonshared, nontransactionally tracked files.

Each buffer allocated allows one file to be cached.

You can increase the number of cache buffers to speed up the process of sequential reads/writes.

NOTE: Increasing the value of this parameter decreases available memory.

Syntax	cache buffers=*number* Replace *number* with the value from the DOS CONFIG.SYS command "FILES = number" minus 5, or with any number through 64.
Default	5
Range	0 to 64
Modules	FIO.VLM

Example	To speed up the process of sequential reads/writes to match a FILES=30 setting in the CONFIG.SYS file, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  cache buffers=25`

CACHE WRITES=[on | off]

Sets the cache write requests to "on" or "off." The default setting "on" increases performance of the NetWare Client software by waiting for a request from the server before writing data to disk.

Setting the value for this parameter to "off" increases data integrity but decreases performance.

Leaving the value for this parameter set to the “on” default can cause data loss if the NetWare server runs out of disk space between write requests.

Syntax	cache writes=[on | off]
Default	on
Modules	FIO.VLM

Example	To increase data integrity at the expense of performance, you could place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  cache writes=off`

CHECKSUM=number

Provides a higher level of data integrity by validating NetWare Core Protocol™, or NCP™, packets.

Setting the value for this parameter to 2 or 3 increases data integrity but decreases performance. The checksum values are the following:

0 = Disabled

1 = Enabled but not preferred

2 = Enabled and preferred

3 = Required

Syntax	checksum=*number*
Default	1
Modules	IPXNCP.VLM, NWP.VLM

Example	To increase data integrity at the expense of performance, you could place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  checksum=3`

NOTE: Ethernet frame type 802.3 doesn't support checksums.

CONFIRM CRITICAL ERROR ACTION=[on | off]

Determines the default method used for handling network critical error messages in MS Windows.

MS Windows often intercepts critical errors intended for the NetWare DOS Requester and displays the following error message

"Cannot read from device network"

or intercepts the error message, but does not display the message because MS Windows is running in Auto Fail mode.

If the error message response is to "Cancel," or MS Windows automatically fails, the network connection that issued the critical error is torn down and the NetWare DOS Requester proceeds to clean up the connection. Nevertheless, the client workstation is not informed of which connection that was lost. The two values are as follows:

On = A dialog box is displayed showing the server name of the connection that has gone down. You are prompted to "Cancel" or "Retry" before MS Windows responds. The dialog box provides additional information about the error and notifies the user in case of an Auto Fail.

Off = MS Windows often intercepts critical errors and automatically responds to error messages intended for the NetWare DOS Requester.

Syntax	confirm critical error action=[on | off]
Default	on
Modules	VLM.EXE

Example	To allow MS Windows to handle network critical errors, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  confirm critical error action=off`

For this parameter to function properly, use the NETWARE.DRV version 3.03 or later, and include the "network.drv=netware.drv" line under the [boot] heading in the MS Windows SYSTEM.INI file.

This parameter affects the performance of your MS Windows environment.

CONNECTIONS=number

Sets the maximum number of connections the NetWare DOS Requester software supports.

The NetWare DOS Requester supports up to 50 connections in its connection table. A value larger than needed uses memory unnecessarily.

Syntax	connections=*number*
Default	8
Range	2 to 50
Modules	AUTO.VLM, CONN.VLM, FIO.VLM, NDS.VLM, SECURITY.VLM

Example	To increase the number of server connections you can have to 12, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  connections=12`

NOTE: Setting the value to other than 8 might decrease NETX compatibility. Do not use this parameter if you are unsure of the maximum number of connections you are making to NetWare 3 or earlier servers.

If you lower the number of connections to fewer than 8, ensure that your client workstation does not exceed the number during operation. If the maximum number of connections is exceeded, you might experience some problems with all of your server connections.

DOS NAME="name"

Sets the name of the operating system used in the shell.

The %OS variable in the login or profile script uses this variable when mapping a search drive to the network DOS directory.

Syntax	dos name="*name*"
Default	MSDOS
Range	1 to 5 characters
Modules	GENERAL.VLM, NETX.VLM

Example	To set the operating system used for the shell to DR DOS®, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  dos name="drdos"`

NOTE: The NetWare DOS Requester software automatically recognizes DR DOS and sets this option. However, setting this option overrides the auto-detect feature.

EOJ=[on | off]

Specifies whether files, locks, semaphores, etc., are closed automatically at end of a job.

This supports End Of Job (EOJ) commands from a DOS INT 21 D6 function call.

Syntax	eoj=[on | off]
Default	on
Modules	NETX.VLM, REDIR.VLM

Example	To specify a client workstation to not send End Of Job commands for closing files to the server, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  eoj=off`

EXCLUDE VLM=path_vlm

Specifies a .VLM file that the VLM.EXE program should not load.

This parameter causes any .VLM file listed in the VLM.EXE program default load table or in the VLM vlm_path parameter to not load when the VLM.EXE program runs.

You must specify the complete filename, including the .VLM extension.

Syntax	exclude vlm=*path_vlm*
Range	1 to 50
Modules	VLM.EXE

Example	To exclude the loading of the PRINT.VLM, SECURITY.VLM, and NMR.VLM files with the core modules, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  exclude vlm=c:\nwclient\print.vlm`
	`  exclude vlm=c:\nwclient`
	`  \security.vlm`
	`  exclude vlm=c:\nwclient\nmr.vlm`

FIRST NETWORK DRIVE=drive_letter

Sets the first network drive to the letter of choice when the NetWare DOS Requester software makes a connection to the NetWare server.

This parameter accepts only the drive letter and not the colon.

Syntax	first network drive=*drive_letter*
Default	First available drive
Range	A to Z
Modules	GENERAL.VLM, NETX.VLM, REDIR.VLM

Example	To set the first drive to G:, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  first network drive=g`

FORCE FIRST NETWORK DRIVE=[on | off]

Specifies the network drive letter the SYS:LOGIN directory is mapped to after logging out of a server or network. Setting the value to "on" specifies that the drive letter must be the same as the one use in FIRST NETWORK DRIVE.

Previously, if you typed "LOGOUT" from any drive other than a local drive or the FIRST NETWORK DRIVE, the SYS:LOGIN drive would be mapped to the drive letter you logged out from.

Syntax	force first network drive=[on | off]
Default	off
Modules	GENERAL.VLM

Example	To specify that the first network drive letter be the same before and after you log out, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  force first network drive=on`

For this parameter to work, also set FIRST NETWORK DRIVE=drive.

The drive is map rooted to the first network drive and not to the \LOGIN directory as with the NetWare Shell software such as NETX. You should modify any batch files effected by this change.

HANDLE NET ERRORS=[on | off]

Determines the default method for handling network errors.

A network error is generated when the client workstation doesn't receive a response from the NetWare server. The two values are as follows:

On = interrupt 24 handles network errors

Off = return NET_RECV_ERROR (example: 8805h)

Syntax	handle net errors=[on | off]
Default	on
Modules	IPXNCP.VLM

Example	To receive an error return of "NET_RECV_ERROR," you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  handle net errors=on`

LARGE INTERNET PACKETS=[on | off]

Sets the Large Internet Packet (LIP) packet size above the default of 576 bytes.

In the past, NetWare communicated across routers and bridges with a 576-byte maximum packet size. However, Ethernet and token ring are capable of using larger packets for communication.

When the value for this parameter is set to "on," the maximum packet size negotiated between the NetWare server and the client workstation is used, even across routers and bridges.

Syntax	large internet packets=[on | off]
Default	on
Modules	IPXNCP.VLM, NWP.VLM

Example	To minimize the packet size used between the NetWare server and client workstation, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  large internet packet=off`

NOTE: Some routers and bridges have been hardcoded to use 576-byte packets. In this case, the NetWare DOS Requester can use only 576-byte packets, regardless of this parameter.

LIP START SIZE=number

Specifies the packet size of used by the NetWare DOS Requester when starting LIP negotiations.

Use this parameter to reduce the amount of traffic caused by the negotiation process across slow links.

Syntax	lip start size=*number*
Default	0 (0=off)
Range	576 to 655535
Modules	IPXNCP.VLM

Example	To cut down on network traffic from Large Internet Packets across a WAN link, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  lip start size=3658`

For this parameter to work, also set LARGE INTERNET PACKETS=ON.

LOAD CONN TABLE LOW=[on | off]

Specifies whether the connection table is loaded in high or conventional memory.

When using the initial release of NetWare 4.0 network operating system utilities, you must set the value of the LOAD CONN TABLE LOW parameter to “on.”

This loads the connection table low, which increases conventional memory requirements required by the initial release of NetWare 4.0 utilities.

If you are not using the initial release of the NetWare 4.0 utilities, the default value (off) gives you better memory performance. The default setting loads the connection table in an Upper Memory Block (UMB), if available.

Syntax	load conn table low=[on | off]
Default	off
Modules	CONN.VLM

Example	To increase conventional memory requirements required by the NetWare Client software for supporting the initial release of NetWare 4.0 utilities, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  load conn table low=on`

NOTE: Use the VLM /D command line parameter to determine how much conventional memory is used by this parameter setting.

LOAD LOW CONN=[on | off]

Specifies whether the connection manager is loaded in high or conventional memory.

By default, the connection manager, CONN.VLM, is loaded in conventional memory.

If the value for this parameter is set to “off,” CONN.VLM loads in upper memory, saving memory but sacrificing performance.

Syntax	load low conn=[on | off]
Default	on
Modules	CONN.VLM

Example	To load CONN.VLM in upper memory, thereby saving memory but sacrificing performance, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  load low conn=off`

NOTE: Use the VLM /D command line parameter to determine how much conventional memory is used by this parameter setting.

LOAD LOW IPXNCP=[on | off]

Specifies whether the transport manager for IPX is loaded in high (XMS or EMS) or conventional memory.

By default, the transport protocol implementation for IPX, IPXNCP.VLM, is loaded in conventional memory.

If the value for this parameter is set to "off," IPXNCP.VLM loads in upper memory, saving memory but sacrificing performance.

Syntax	load low ipxncp=[on | off]
Default	on
Modules	IPXNCP.VLM

Example	To load IPXNCP.VLM in upper memory, thereby saving memory but sacrificing performance, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  load low ipxncp=off`

NOTE: Use the VLM /D command line parameter to determine how much conventional memory is used by this parameter setting.

LOAD LOW REDIR=[on | off]

Specifies whether the redirector is loaded in high or conventional memory.

This parameter influences the speed of repeated small file read/writes to the network, such as database transactions.

By default, the redirector portion of the NetWare DOS Requester software, REDIR.VLM, is loaded in high memory.

If the value for this parameter is set to "on," REDIR.VLM loads in conventional memory, increasing performance but sacrificing memory.

Syntax	load low redir=[on | off]
Default	off
Modules	REDIR.VLM

Example	To load REDIR.VLM in conventional memory, thereby increasing performance but sacrificing memory, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  load low redir=on`

NOTE: Use the VLM /D command line parameter to determine how much conventional memory is used by this parameter setting.

LOCAL PRINTERS=number

Overrides the number of local printers connected to the client workstation, which is normally determined by the BIOS. The BIOS defaults to one local printer for each parallel port.

By setting the number of local printers to 0, you can prevent your workstation from hanging when you press <Shift>+<Print Screen> without any port capture or local printer connection.

Syntax	local printers=*number*
Default	3
Range	0 to 9
Modules	PRINT.VLM

Example	To prevent your workstation from hanging when you press <Shift> and <Print Screen> without any port capture or local printer connection, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  local printers=0`

LOCK DELAY=number

Determines the amount of time (in ticks) the NetWare DOS Requester software waits before trying to get a lock.

When many users access a file at the same time, the NetWare DOS Requester software might be unable to gain access before its allotted wait time.

Use this parameter if client workstations frequently receive error messages when a file is requested.

Syntax	lock delay=*number*
Default	1 tick
Range	0 to 255 (approximately six hours)
Modules	GENERAL.VLM

Example	To automatically clear messages from a user's screen after three hours, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  lock delay=50`

NOTE:

This number is used for lock types that do not have a wait ability. For locks that have a wait ability, the wait time is calculated by multiplying this parameter number by the LOCK RETRIES number and then multiplying by 2. The resulting number is the time, in ticks, the client workstation waits for a lock.

To determine the total time (in ticks) needed to broadcast a name resolution packet across the network, multiply the wait time value by the value used for the LOCK RETRIES parameter.

There are approximately 18.21 ticks per second on IBM PCs and compatibles.

LOCK RETRIES=number

Specifies the number of times the NetWare DOS Requester software attempts to get a lock on the network.

It is part of an equation that determines the total time the NetWare DOS Requester waits when attempting to access a locked file. If a client workstation frequently receives error messages when a file is requested, increase the value of this parameter.

Syntax	lock delay=*number*
Default	1 tick
Range	0 to 255 (approximately six hours)
Modules	GENERAL.VLM

Example	To automatically lock files, you could place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  lock retires=50`

NOTE: This number is used for lock types that do not have a wait ability. For locks that have a wait ability, the wait time is calculated by multiplying this parameter number by the LOCK DELAY number and then multiplying by 2. The resulting number is the time, in ticks, the client workstation waits for a lock.

To determine the total time (in ticks) needed to broadcast a name resolution packet across the network, multiply the wait time value by the value used for the LOCK DELAY parameter.

There are approximately 18.21 ticks per second on IBM PCs and compatibles.

LONG MACHINE TYPE="name"

Tells the NetWare DOS Requester software what type of machine is being used each time the %MACHINE variable is accessed.

Use this parameter and value to set the machine's search path to the correct version of DOS.

Syntax	long machine type="*name*"
Default	IBM-PC
Range	1 to 6
Modules	GENERAL.VLM, NETX.VLM

Example	To specify the machine type as COMPAQ, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  long machine type="compaq"`

MAX TASKS=number

Configures the maximum number of tasks that can be active simultaneously.

Certain multitasking applications, such as MS Windows and DESQview*, allow several applications to run simultaneously.

NOTE: If you have problems running a new application, increase the value of this parameter.

Syntax	max tasks=*number*
Default	31
Range	5 to 254
Modules	CONN.VLM

Example	To allow for 68 tasks to run simultaneously, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  max tasks=68`

MESSAGE LEVEL=number

Sets how load-time messages are displayed. Each message level implies the previous level's message (for example, 1 implies 0).

The values are as follows:

0 = Display copyright message and critical errors

1 = Display warning messages

2 = Display program load information for VLM programs

3 = Display configuration information

4 = Display diagnostic information

Syntax	message level=*number*
Default	1
Modules	VLM.EXE

Example	To display only configuration information, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  message level=3`

MESSAGE TIMEOUT=number

Defines how long (in ticks) before broadcast messages are cleared from the screen without user intervention.

Syntax	message timeout=*number*
Default	0
Range	0 (wait for user) to 10,000 (approximately nine minutes)
Modules	NWP.VLM

Example	To automatically clear messages from a user's screen after approximately five minutes, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  message timeout=5000`

MINIMUM TIME TO NET=number

Overrides the time-to-net value defined by the local router during connection.

This parameter is used for bridged WAN/Satellite links with time-to-net values set too low for workstations to make a connection under either of the following conditions:

- The server on the other side of the link is a NetWare 3™ or earlier server not running the Packet Burst™ protocol
- The transfer rate for the link is 2400 baud or less

Syntax	minimum time to net=*number* Replace *number* with the number of milliseconds (1000=one second) needed for maintaining a connection over bridges and routers.
Default	0
Range	1 to 65535
Modules	IPXNCP.VLM

Example	To increase the time-to-net value for a 2400-baud link to 10 seconds, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  minimum time to net=10000`

NAME CONTEXT="name_context"

Allows you to set your current position, or context, in the Directory tree structure.

This parameter and value apply only to client workstations connecting to a NetWare 4 network.

The default is the root, which might cause confusion if duplicate usernames exist.

Syntax	name context="*name_context*"
Default	Root
Range	Root to 257 (256 characters plus the NULL command)
Modules	NDS.VLM

Example	To set your name context to the marketing Organization, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  name context="ou=mngt.o=marketing"`

NETWARE PROTOCOL=netware_protocol_list

Allows you to list the order that the NetWare protocols (NDS, BIND, and PNW) are used in.

For example, you can give priority to a specific protocol for login, load order, or other functions performed by the NetWare DOS Requester software.

Each protocol is separated by a comma or a space in the list.

Syntax	netware protocol=*netware_protocol_list*
Default	nds bind pnw
Modules	VLM.EXE

Example	To speed up login to a Personal NetWare server, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  netware protocol=pnw nds bind`

If using this parameter, list it first under the NetWare DOS Requester option of the NET.CFG file.

NETWORK PRINTERS=number

Sets the number of LPT ports the NetWare DOS Requester software can capture. This parameter allows you to capture and redirect LPT1 through LPT9.

Increasing the number value increases memory use. Setting the value for this parameter to 0 specifies that PRINT.VLM does not load.

If you are setting more than three ports in MS Windows, edit the MS Windows WIN.INI file with an ASCII text editor and add as many lines as you need under the [port] section.

For example:

```
[port]
lp1:=
lp2:=
lp3:=
lp4:=
net1:=
record.prn=
```

Add a filename to a line under the [ports] option to have print files print directly to this file. The filename should have a .PRN file extension followed by an equal sign.

This causes the named file to appear in the Control Panel “Printer Configuration” dialog box. Any print jobs sent to this file direct their output to it.

Syntax	network printers=*number*
Default	3
Range	0 to 9
Modules	PRINT.VLM

Example	To allow for seven printer connections, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  network printers=7`

If you are using any application that uses the “netx=off” command, such as Microsoft Office, ensure that this parameter value is set to “on.”

PB BUFFERS=number

Controls the use of the Packet Burst protocol for file input/output.

Packet Burst is automatically enabled in the NetWare DOS Requester software.

The values are as follows:

0 = off

nonzero = on and increases the number of buffers.

Setting this option to 0 decreases memory use and, in some cases, performance.

Syntax	pb buffers=*number*
Default	3
Range	0 to 10
Modules	FIO.VLM, IPXNCP.VLM

Example	To disable Packet Burst, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  pb buffers=0`

PBURST READ WINDOWS SIZE=number

Sets the read buffer size (in bytes) for MS Windows.

Syntax	pburst read windows size=*number*
Default	16
Range	3 to 128
Modules	FIO.VLM

Example	To increase the read buffer size for better performance, you could place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  pburst read windows size=64`

WARNING: **If you are using this option on any client workstation not running Packet Burst, changing the default of this parameter might result in critical network errors.**

PBURST WRITE WINDOWS SIZE=number

Sets the write buffer size (in bytes) for MS Windows.

Syntax	pburst write windows size=*number*
Default	10
Range	3 to 128
Modules	FIO.VLM

Example	To increase write buffer size, you could place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  pburst write windows size=24`

WARNING: **If you are using this option on any client workstation not running Packet Burst, changing the default of this parameter might result in critical network errors.**

PREFERRED SERVER=“server_name”

Sets the NetWare server you attach to first and helps guarantee your connection to the network.

If the server specified has a connection available, the NetWare DOS Requester attaches to that server. Otherwise, it responds to the nearest broadcasting server.

Syntax	preferred server=“*server_name*”
Default	None
Modules	BIND.VLM

Example	To specify the MKT_9 server as your preferred server connection, thereby speeding up log in to bindery services, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  preferred server=mkt_9`

NOTE: If both PREFERRED TREE (for NetWare Directory Services) and PREFERRED SERVER (for bindery services) are specified, then the first protocol to successfully build an attachment is used.

PREFERRED TREE=“tree_name”

Sets the Directory tree you first want to connect to in a NetWare 4 network if you have multiple trees.

If the tree specified has a server with a free connection, the NetWare DOS Requester attaches to that tree. Otherwise, it attaches to the nearest tree that contains a User object for the user authenticating to the network server.

Syntax	preferred tree=“*tree_name*”
Default	None
Modules	NDS.VLM

Example	To specify the MARKETING tree as your preferred tree connection, thereby speeding up log in to NetWare Directory Services, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  preferred tree=“marketing”`

NOTE: If both PREFERRED TREE (for NetWare Directory Services) and PREFERRED SERVER (for bindery services) are specified, then the first protocol to successfully attach is used, such as NetWare Directory Services or Bindery services.

PREFERRED WORKGROUP="workgroup_name"

Sets the Personal NetWare workgroup you attach to first and helps guarantee your connection to the network.

If the workgroup specified has a connection available, the NetWare DOS Requester software attaches to that workgroup. Otherwise, it connects to the nearest workgroup.

Syntax	preferred workgroup="*workgroup_name*"
Default	None
Modules	PNW.VLM

Example	To specify the MKT_9 workgroup as your preferred workgroup connection, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  preferred workgroup="mkt_9"`

NOTE: If the PREFERRED TREE (for NetWare Directory Services), PREFERRED SERVER (for bindery services), and PREFERRED WORKGROUP (for Personal NetWare Services) parameters are all specified, then the first protocol to successfully build an attachment is used.

PRINT BUFFER SIZE=number

Determines the size (in bytes) for the print buffer. The print buffer acts as a cache for 1-byte print requests, which increases the size of some jobs.

Syntax	print buffer size=*number*
Default	64
Range	0 to 256

Modules	PRINT.VLM

Example	To set the buffer size for printing to 86, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  print buffer size=86`

NOTE: While increasing the speed in some of your printing output, this parameter also increases memory use.

PRINT HEADER=number

Sets the size of the buffer (in bytes) that holds the information used to initialize a printer for each print job.

If you send print jobs with many instructions in the header (such as initializing a printer for an emulated mode or changing defaults, font selections, page length, or orientation) and the printer is not delivering all the requested attributes, increase the print header size.

Syntax	print header=*number*
Default	64
Range	0 to 1024
Modules	PRINT.VLM

Example	To increase the size of the print header in order for all the printer attributes of a larger header to be delivered, you could place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  print header=960`

PRINT TAIL=number

Sets the size of the buffer (in bytes) that holds the information used to reset the printer after a print job.

If your printer is not clearing out the buffer completely or resetting after each print job, increase the print tail size.

Syntax	print tail=*number*
Default	16
Range	0 to 1024
Modules	PRINT.VLM

Example	To increase the size of the print header, you could place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  print header=64`

READ ONLY COMPATIBILITY=[on | off]

Determines whether a file marked Read Only can be opened with a read/write access call.

Certain applications require the value for this parameter to be set to "on." See the documentation provided with your application for details.

Prior to NetWare 2.1, a program could open a Read Only file with write access without getting an error, though any attempt to write to the file produced an error.

To be compatible with DOS, NetWare 2.1 or later does not allow a Read Only file to be opened for write access.

Setting this parameter to "on" causes the shell to revert to the old mode and allow the open request to succeed.

Syntax	read only compatibility=[on | off]
Default	off
Modules	REDIR.VLM

Example	To cause the shell to revert to the old mode and allow the open request to succeed, you would place the following in your NET.CFG file:
	
	`netware dos requester`
	`  read only compatibility=on`

NOTE: If you are using any application that uses the "netx=off" command, such as Microsoft Office, ensure that this parameter value is set to "on."

RESPONDER=[on | off]

Controls the communication and response of the client workstation. Also helps in reducing the NetWare DOS Requester software memory use on the client workstation.

Setting the value to "off" causes the client workstation to ignore broadcasts and diagnostic communication.

Syntax	responder=[on | off]
Default	on
Modules	PNW.VLM

Example	To reduce conventional memory use, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  responder=off`

SEARCH MODE=number

Alters the NetWare DOS Requester software's method for finding a file if it is not in the current directory.

In previous NetWare Client™ software versions, the default drive had to be a network drive for this parameter to function. But the NetWare DOS Requester is global and affects all .EXE and .COM files, regardless of the current drive.

When using Search Mode, select the search mode that works correctly with most of your .EXE and .COM files.

If you want to set a search mode for one particular .EXE or .COM file, use the "Search Mode" option in FLAG (see "FLAG" in NetWare 4.1 Utilities Reference or see "FLAG" in NetWare 3.12 Utilities Reference).

Valid search mode values are as follows:

Value	Explanation
0	No search instructions. Default value for executable files.
1	If a directory path is specified in the executable file, the executable file searches only that path. If a path is not specified, the executable file searches the default directory and network search drives.
2	The executable file searches only the default directory or the path specified.
3	If a directory path is specified in the executable file, the executable file searches only that path. If a path is not specified and the executable file opens data files flagged Read Only, the executable file searches the default directory and search drives.
4	Reserved.
5	The executable file searches the default directory and NetWare search drives whether or not the path is specified. If a search mode is set, the shell allows searches for any files with .XXX extension; otherwise the executable file searches only for .EXE, .COM, and .BAT files.
6	Reserved.
7	If the executable file opens data files flagged Read Only, the executable file searches the default directory and search drives whether or not the path is specified in the executable file.

Syntax	search mode=number
Default	1
Modules	GENERAL.VLM

Example	To cause the executable file to search only the default directory or the path specified, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  search mode=2`

SET STATION TIME=[on | off]

Synchronizes the client workstation date and time with that of the NetWare server that the client workstation initially attaches to.

Setting this option to “off” disables the synchronization feature.

Syntax	set station time=[on | off]
Default	on
Modules	VLM.EXE

Example	To disable the synchronization feature, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  set station time=off`

SHORT MACHINE TYPE=“name”

Specifies which overlay files to use with the specific machine type of your client workstations.

This parameter is similar to LONG MACHINE TYPE, except that it is used specifically with overlay files. Use it when the %SMACHINE variable is accessed.

Examples of files using this parameter and value include the IBM$RUN.OVL file for the windowing utilities and the CMPQ$RUN.OVL file that uses a default black-and-white color palette for NetWare menus.

Syntax	short machine type=“*name*”
Default	IBM

Range	1 to 4
Modules	GENERAL.VLM, NETX.VLM

Example	To specify the machine type as AST, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  short machine type="ast"`

SHOW DOTS=[on | off]

Specifies to display parent dots in list boxes for file and directory navigation.

NOTE: This parameter is supported only by NetWare 2.11 and later.

The NetWare server doesn't have directory entries for (“.” and “..”) as DOS does. To see “.” and “..” in directory listings, set the value for this parameter to “on.”

Syntax	show dots=[on | off]
Default	off
Modules	REDIR.EXE

Example	To see parent dots in a directory listings, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  show dots=on`

NOTE: You should set SHOW DOTS=ON when using MS Windows 3.x or DOS graphical user interface (GUI) with NetWare.

SIGNATURE LEVEL=number

Designates the level of enhanced security support.

Enhanced security includes the use of a message digest algorithm and a per connection/per request session state. The values are as follows:

0 = Disabled (the SECURITY.VLM file does not load)

1 = Enabled but not preferred

2 = Preferred

3 = Required

NOTE: Setting this option to 2 or 3 increases security but decreases performance.

Syntax	signature level=*number*
Default	1
Modules	NWP.VLM, SECURITY.VLM

Example	To designate the level of enhanced security support as "required," you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  signature level=3`

Do not set the signature level in the NET.CFG file to a value of 0 if you are using AUTO RECONNECT or BIND RECONNECT parameters when connecting to NetWare 3 or earlier servers.

If this value is set to 0 in this situation, the reconnect feature will not function.

TRUE COMMIT=[on | off]

Selects whether the commit NCP is sent on DOS commit requests.

Off = Opts for performance over integrity

On = Opts for integrity over performance

Set this option to "on" when processing critical data (for example, database applications) to guarantee data integrity.

Syntax	true commit=[on | off]
Default	off
Modules	FIO.VLM

Example	To select integrity over performance, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  true commit=on`

USE DEFAULTS=[on | off]

Overrides the default .VLM files that the VLM.EXE program loads. Without it, the VLM.EXE program attempts to load the following files in the given order:

CONN.VLM

IPXNCP.VLM

TRAN.VLM

SECURITY.VLM

NDS.VLM

BIND.VLM

NWP.VLM

FIO.VLM

GENERAL.VLM

REDIR.VLM

PRINT.VLM

NETX.VLM

AUTO.VLM

If you specify VLM programs that are normally loaded by default in the NET.CFG file and you don't set the value for this parameter to "off," any default VLM program you specify attempts to load twice, producing an error during load.

Syntax	use defaults=[on | off]
Default	on
Modules	VLM.EXE

Example	To load the AUTO.VLM file with the core modules and disable loading of the BIND.VLM file, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  use defaults=off`
	`  vlm=conn.vlm`
	`  vlm=ipxncp.vlm`
	`  vlm=tran.vlm`
	`  vlm=security.vlm`
	`  vlm=nds.vlm`
	`  ;vlm=bind.vlm`
	`  vlm=nwp.vlm`
	`  vlm=fio.vlm`
	`  vlm=general.vlm`
	`  vlm=redir.vlm`
	`  vlm=print.vlm`
	`  vlm=netx.vlm`
	`  vlm=auto.vlm`

NOTE: The EXCLUDE VLM parameter should be used instead of the USE DEFAULTS parameter for excluding specific core VLM programs.

VLM=path_VLM

Specifies a .VLM file that the VLM.EXE program should load.

This parameter and value allow VLM programs not listed in the default load table for the VLM.EXE program to be added to the table.

The following .VLM files are not listed in the default load table:

AUTO.VLM

MIB2IF.VLM

MIB2PROT.VLM

NMR.VLM

RSA.VLM

WSASN1.VLM

WSDRVPRN.VLM

WSREG.VLM

WSSNMP.VLM

WSTRAP.VLM

You must specify the complete filename, including the .VLM extension.

Syntax	vlm=*path_vlm*
Default	None
Range	1 to 50
Modules	VLM.EXE

Example	To load AUTO.VLM, RSA.VLM, and NMR.VLM with the core modules, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  vlm=c:\nwclient\auto.vlm`
	`  vlm=c:\nwclient\rsa.vlm`
	`  vlm=c:\nwclient\nmr.vlm`

WORKGROUP NET=workgroup_net_address

Provides a way for your client workstation to find workgroup information outside of your local network.

If your client workstation resides physically on a network segment other than the one that the workgroup you want to connect to is on, setting this address ensures that a connection is found.

This parameter and value should be modified only with the Personal NetWare utilities and should not be edited manually. Manually setting a value for this parameter might result in loss of connection.

Syntax	workgroup net=*workgroup_net_address*
Default	None
Modules	PNW.VLM

Example	To set a network address of 00123099:FFFFFFFFFFFF for your workgroup, you would place the following lines in your NET.CFG file:
	
	`netware dos requester`
	`  workgroup net=00123099:FFFFFFFFFFFF`

Protocol IPX Option

Use this option of the NET.CFG file to change the default value of parameters for the IPX™ protocol.

Available Parameters and Values for the Protocol IPX Option

This option has the following parameters and values, which are discussed on the following pages:

Parameters and Values
BIND LAN_driver_name [#number] INT64 [on | off] INT7A [on | off] IPATCH byte_offset, value IPX PACKET SIZE LIMIT number IPX RETRY COUNT number IPX SOCKETS number

PROTOCOL IPX

Specifies the protocol option you are making configurations for.

Syntax	protocol ipx *parameter_name value* Replace *parameter_name* with the name of the parameter you want to use. Replace *value* with the value setting which corresponds with the parameter name.

Example	To configure for the IPX protocol to not use interrupt 64h, you would place the following lines in your NET.CFG file:
	
	`protocol ipx`
	`  int64 off`

BIND LAN_driver_name [#number]

Usually a protocol binds to the first LAN driver it finds. This parameter forces the protocol to bind to the LAN driver you specify.

You are informed of the board number you are binding to when the LAN driver is loading.

Syntax	bind *LAN_driver_name* [*#number*] Replace *LAN_driver_name* with the name of the LAN driver you want the protocol to bind to. Replace *number* with the occurrence of the LAN driver you are loading. If you are binding only one LAN driver, you do not need to set this value. If you want to bind multiple LAN drivers, specify the BIND parameter multiple times, each on a different line in the NET.CFG file.
Default	None

Example	To bind protocol IPXODI to your NE2000 LAN driver, you would place the following lines in your NET.CFG file:
	
	`protocol ipx`
	`  bind ne2000`
	
	To bind protocol IPXODI to the second and third logical LAN drivers, you would place the following lines in your NET.CFG file:
	
	`protocol ipx`
	`  bind ne2000 #2, #3`

NOTE: Some protocols do not support binding to multiple LAN drivers. Refer to the protocol's documentation for binding information.

INT64 [on | off]

Allows applications to use interrupt 64h to access IPX services.

IPX now uses interrupt 64h to maintain compatibility with earlier versions of NetWare software.

If an application's documentation requests interrupt 64h, or if you have an application that works with earlier versions of NetWare but hangs with NetWare 3.1, set the value for this parameter to "off."

Syntax	int64 [on | off]
Default	on

Example	To allow an application use interrupt 64h, you would place the following lines in your NET.CFG file:
	
	`protocol ipx`
	`  int64 off`

INT7A [on | off]

Allows applications to use interrupt 7Ah to access IPX services.

IPX now uses interrupt 7Ah to maintain compatibility with NetWare 2.0a.

If an application's documentation requests interrupt 7Ah, or if an application works on earlier versions of NetWare software but hangs with NetWare 3.1, set the value for this parameter to "off."

Syntax	int7a [on | off]
Default	on

Example	To allow an application to use interrupt 7Ah, you would place the following lines in your NET.CFG file:
	
	`protocol ipx`
	`  int7a off`

IPATCH byte_offset, value

Allows any address in the IPXODI.COM file to be patched with any specified byte offset value.

Syntax	**ipatch *byte_offset, value***
Default	None

Example	To patch a byte offset value in the IPXODI.COM file, you could place the following lines in your NET.CFG file:
	
	`protocol ipx`
	`  ipatch 6775`

IPX PACKET SIZE LIMIT number

Reduces the maximum packet size (in bytes) set by each LAN driver.

Even though a LAN driver could send 16KB packets across the wire, the wasted memory for most operations might be unacceptable.

The optimum packet size for token ring drivers is 4160 bytes. For Ethernet, the optimum is 1500 bytes. Reduce the maximum packet size if you receive out-of-memory errors at the client workstation.

This parameter is a new feature, and not all LAN drivers support it. See the documentation provided with your LAN driver for details.

Syntax	ipx packet size limit *number* Replace *number* with the size set by the LAN driver.
Default	The lesser of either 4160 or the size specified by the LAN driver
Range	576 to 6500

Example	To reduce the maximum packet size for an Ethernet ODI LAN driver from the default, you could place the following lines in your NET.CFG file:
	
	`protocol ipx`
	`  ipx packet size limit 1200`

IPX RETRY COUNT number

Sets the number of times the client workstation resends a packet. On networks that lose many packets, this retry count might need to be increased.

The IPX protocol does not actually resend a packet. It uses this count to recommend the number of retries to the NetWare DOS Requester software and the SPX protocol.

Increasing this number causes a longer delay for some network functions, such as establishing a NetBIOS session or registering a NetBIOS name.

Syntax	ipx retry count *number* Replace *number* with the number of times the client workstation resends a packet.
Default	20

Example	To set the number of times the client workstation resends a packet to 30, you would place the following lines in your NET.CFG file:
	
	`protocol ipx`
	`  ipx retry count 30`

IPX SOCKETS number

Specifies the maximum number of sockets that IPX can have open at the client workstation.

A program developed to the IPX protocol standard, such as the LANSchool utility, might require more than the default number of sockets.

Syntax	ipx sockets *number* Replace *number* with the maximum number of sockets that IPX can have open at the client workstation.
Default	20

Example	To set the number of open sockets on a client workstation to 30, you would place the following lines in your NET.CFG file:
	
	`protocol ipx`
	`  ipx sockets 30`

Protocol SPX Option

Use this option of the NET.CFG file to change the default value of parameters for the SPX protocol.

Available Parameters and Values for the Protocol SPX Option

This option has the following parameters and values, which are discussed on the following pages:

Parameters and Values
MINIMUM SPX RETRIES number SPX ABORT TIMEOUT number SPX CONNECTIONS number SPX LISTEN TIMEOUT number SPX VERIFY TIMEOUT number

PROTOCOL SPX

Specifies the protocol option you are making configurations for. Use it to override the SPX retry value set by an application if SPX sessions are being destroyed because of a lack of acknowledgment between session hosts.

Syntax	protocol spx *parameter_name value* Replace *parameter_name* with the name of the parameter you want to use. Replace *value* with the value setting which corresponds with the parameter name.

Example	To set the number of retries to 30, you could place the following lines in your NET.CFG file:
	
	`protocol spx`
	`  minimum spx retries 30`

MINIMUM SPX RETRIES number

Determines how many unacknowledged transmit requests are allowed before assuming the connection has failed.

SPX applications have two methods of specifying a transmit retry count to SPX:

- The application can specify a retry value at the time of connection.
- If the application doesn't specify a retry value, SPX uses the configured value setting for the IPX RETRY COUNT parameter (default is 20).

This option is especially useful if an application that uses the SPX protocol is abnormally losing its connections. This might be due to a low value specified in the default retry count in the application or in the IPX RETRY COUNT parameter.

This parameter is supported by IPXODI.COM 2.00 and later.

Syntax	minimum spx retries *number* Replace *number* with the minimum number of times unacknowledged transmit requests are allowed before assuming the connection has failed.
Default	20
Range	0 to 255

Example	To set the number of retries to 30, you would place the following lines in your NET.CFG file:
	
	`protocol spx`
	`  minimum spx retries 30`

SPX ABORT TIMEOUT number

Specifies how long (in ticks) the SPX protocol waits without receiving any response from the other side of the connection before it terminates the session.

Syntax	spx abort timeout *number* Replace *number* with a number of ticks.
Default	540 (about 30 seconds)

Example	To set the amount of time (in ticks) that SPX waits before it terminates the session to 20 seconds, you would place the following lines in your NET.CFG file:
	
	`protocol spx`
	`  spx abort timeout 300`

NOTE: There are approximately 18.21 ticks per second on IBM PCs and compatibles.

SPX CONNECTIONS number

Specifies the maximum number of SPX connections a client workstation can use at one time.

The NPRINTER and RCONSOLE utilities use this setting. Increase the value of this setting if you are running multiple occurrences of these or other applications that use SPX.

Syntax	spx connections *number*
Default	15

Example	To set the number of concurrent SPX connections to 10, you would place the following lines in your NET.CFG file:
	
	`protocol spx`
	`  spx connections 10`

SPX LISTEN TIMEOUT number

Specifies how long (in ticks) the SPX protocol waits without receiving a packet from the other side of the connection before it requests the other side to send a packet to ascertain whether the connection is still valid.

If SPX has not heard from the other side of the connection within this time, it sends packets to the other side asking for verification that the connection still exists.

Syntax	spx listen timeout *number* Replace *number* with a number of ticks.
Default	108 (about 6 seconds)

Example	To set the amount of time (in ticks) that SPX waits before it requests a packet to 11 seconds, you would place the following lines in your NET.CFG file:
	
	`protocol spx`
	`    spx listen timeout 200`

NOTE: There are approximately 18.21 ticks per second on IBM PCs and compatibles.

SPX VERIFY TIMEOUT number

Specifies how often (in ticks) the SPX protocol sends a packet to the other side of a connection to indicate that it is still alive.

If no packets are being exchanged on the SPX connection by the software that established the session, SPX sends packets at regular intervals to verify its presence in the connection.

Syntax	spx verify timeout *number* Replace *number* with a number of ticks.
Default	54 (about 3 seconds)

Example	To specify that SPX send a packet to the other side of a connection every six seconds, you would place the following lines in your NET.CFG file:
	
	`protocol spx`
	`   spx verify timeout 108`

NOTE: There are approximately 18.21 ticks per second on IBM PCs and compatibles.

Protocol TCPIP Option

Use this option of the NET.CFG file to configure the TCP/IP protocol stack used by the TCPIP.EXE program. The TCPIP.EXE program reads the NET.CFG file for configuration information while loading.

Run the TCP/IP Transport for DOS installation program provided on the TCP/IP Transport for DOS diskette to obtain a copy of the TCPIP.EXE file. See Chapter 1, “Installing the Transport,” in TCP/IP Transport for DOS Installation Guide for instructions.

Available Parameters and Values for the Protocol TCPIP Option

This option has the following categories, parameters, and values, which are discussed on the following pages:

Category	Parameters and Values
LAN Drivers	BIND odi_driver [number frame_type network_name]
IP Addresses	IP_ADDRESS ip_address [network_name] IP_NETMASK net_mask_address [network_name] IP_ROUTER ip_address [network_name]
Connection Sockets	TCP_SOCKETS number UDP_SOCKETS number RAW_SOCKETS number
NetBIOS Support	NB_ADAPTER [0 | 1] NB_BRDCAST [0 | 1] NB_COMMANDS number NB_DOMAIN domain_name NB_SESSIONS number

Category	Parameters and Values
Additional Support	NO_BOOTP PATH TCP_CFG [[drive:]path [; ...]]

PROTOCOL TCPIP

Specifies the protocol option you are making configurat ions for.

Syntax	protocol tcpip *parameter_name value* Replace *parameter_name* with the name of the parameter you want to use. Replace *value* with the value setting which corresponds with the parameter name.

Example	To set the IP address for your client workstation and distinguish your connection to the network, you could place the following lines in your NET.CFG file:
	`protocol tcpip`
	`  bind ne2000 2 finance-net`
	`  ip_address 129.47.30.6 finance-net`

LAN Drivers

Use this parameter and value to bind the TCP/IP protocol stack to a LAN driver., which is discussed on the following page.

Parameter and Value
BIND odi_driver [number frame_type network_name]

The NetWare Client for DOS and MS Windows software provides ODI LAN drivers and COMn device drivers for the TCP/IP protocol stack.

You can set up client workstations to use either an Ethernet connection with a SLIP/PPP serial link or a network gateway machine with two network boards.

Protocol TCPIP supports one network board and LAN driver per client workstation.

BIND odi_driver [number frame_type network_name]

Binds the TCP/IP protocol stack to a LAN driver. When TCP/IP is bound to a LAN driver, the network board or COMn device for that LAN driver is the board or device used for transmissions to and from the network.

This parameter is needed only if you are running more than one ODI LAN driver to support TCP/IP.

Syntax	bind *odi_driver* [*number frame_type network_name*] Replace *odi_driver* with a token-ring or Ethernet ODI LAN driver name, or with the SLIP_PPP driver name. Replace *number* with the instance of the LAN driver number. Use of this number binds TCP/IP to a particular occurrence of a network board when you have two boards with the same name. This number reflects the load sequence number of the network board (not the same as the logical board number displayed by the LAN driver when it is loaded). Using board number 0 instructs TCP/IP to bind to the first board it finds that supports both TCP/IP and the specified frame type.

Syntax (*continued*)	Replace *frame_type* with the frame type used for your network connection. This value is the same as the frame value for the LINK DRIVER parameter. Replace *network_name* with a descriptive name for this network connection. The network name is used with the IP_ADDRESS, IP_ROUTER, and IP_NETMASK parameters to distinguish between the values for each network connection. If multiple LAN drivers are to be bound to, specify the BIND parameter multiple times, each on a different line in the NET.CFG file.
Default	1

Example	To bind TCP/IP to an NE2000 LAN driver for a single network board in your client workstation connected to the FINANCE-NET network, you would add the following lines to your NET.CFG file:
	
	`protocol TCPIP`
	`  bind ne2000 finance-net`
	
	To bind TCP/IP to an NE2000 LAN driver for both the first and the second network boards in your client workstation, you would add the following lines to your NET.CFG file:
	
	`Protocol TCPIP`
	`  bind ne2000 1`
	`  bind ne2000 2`

IP Addresses

Use these parameters and values to specify your client workstation's IP address, the subnetwork mask, and the default network router address, which are discussed on the following pages:

Parameters and Values
IP_ADDRESS ip_address [network_name] IP_NETMASK net_mask_address [network_name] IP_ROUTER ip_address [network_name]

IP_ADDRESS ip_address [network_name]

Specifies the IP address for your client workstation.

Syntax	ip_address *ip_address* [*network_name*] Replace *ip_address* with the correct address in dotted notation. If this parameter is missing or is 0.0.0.0, the protocol stack uses BOOTP or Reverse ARP to determine the IP address. Replace *network_name* with a descriptive name for this network connection. The network name is used with the BIND, IP_ROUTER, and IP_NETMASK parameters to distinguish between the values for each network connection. The network name is required only if you are configuring multiple ODI LAN drivers for TCP / IP.
Default	None

Example	To set the IP address for your client workstation and distinguish your connection to the network, you could place the following lines in your NET.CFG file:
	
	`protocol tcpip`
	`  bind ne2000 2 finance-net`
	`  ip_address 129.47.30.6 finance-net`
	`  ip_router 144.52.6.6`
	`  ip_netmask 255.255.0.0`

IP_NETMASK net_mask_address [network_name]

Specifies the default subnetwork mask if subnetworks are used.

Syntax	ip_netmask *net_mask_address* [*network_name*] Replace *net_mask_address* with the correct subnetwork mask address in dotted notation. If this parameter is missing or is 0.0.0.0, the protocol stack uses BOOTP or Reverse ARP to determine the IP address. Replace *network_name* with a descriptive name for this network connection. The network_name is used with the BIND, IP_ADDRESS, and IP_ROUTER parameters to distinguish between the values for each network connection. The network name is required only if you are configuring multiple ODI LAN drivers for TCP/IP.
Default	None

Example	To set the IP netmask for your client workstation and distinguish your connection to the network, you could place the following lines in your NET.CFG file:
	
	`protocol tcpip`
	`  bind ne2000 2 finance-net`
	`  ip_address 129.47.6.84`
	`  ip_router 144.52.6.6`
	`  ip_netmask 255.255.0.0 finance-net`

IP_ROUTER ip_address [network_name]

Specifies the default router address for all packets being sent to remote networks. All other gateways are dynamically discovered using the ICMP redirect mechanism.

Syntax	ip_router *ip_address* [*network_name*] Replace *ip_address* with the correct address in dotted notation. If this parameter is missing or is 0.0.0.0, the protocol stack uses BOOTP or Reverse ARP to determine the IP address. Replace *network_name* with a descriptive name for this network connection. The network name is used with the BIND, IP_ADDRESS, and IP_NETMASK parameters to distinguish between the values for each network connection. The network name is required only if you are configuring multiple ODI LAN drivers for TCP / IP.
Default	None

Example	To set the IP address for your client workstation and distinguish your connection to the network, you could place the following lines in your NET.CFG file:
	
	`protocol tcpip`
	`  bind ne2000 2 finance-net`
	`  ip_address 129.47.6.84`
	`  ip_router 144.52.6.6 finance-net`
	`  ip_netmask 255.255.0.0 finance-net`

Connection Sockets

Use these parameters and values to specify the maximum number of concurrent Transmission Control Protocol (TCP), User Datagram Protocol (UDP), and raw socket connections that your client workstation can support, which are discussed on the following pages:

Parameters and Values
TCP_SOCKETS number UDP_SOCKETS number RAW_SOCKETS number

Transmission Control Protocol (TCP) Sockets

Transmission Control Protocol (TCP) sockets are used to support concurrent TCP connections.

If you configure multiple ODI LAN drivers for TCP/IP and anticipate using applications that make heavy concurrent use of sockets, modify the number of TCP sockets by configuring the TCP_SOCKETS parameter value to a larger number.

TCP_SOCKETS number

Specifies the maximum number of concurrent TCP sockets (connections).

If you are configuring multiple ODI LAN drivers for TCP/IP, set the number of TCP sockets to the total shared by all drivers.

The value for your TCP_SOCKETS must be at least equal to the NB_SESSIONS value plus 1. See “NB_SESSIONS number” on page 215 for more information.

Syntax	tcp_sockets *number*
Default	8
Range	0 to 64

Example	To set a maximum of sixteen concurrent TCP connections, you could place the following lines in your NET.CFG file:
	
	`protocol tcpip`
	`  ip_address 129.47.6.84`
	`  ip_router 144.52.6.6`
	`  ip_netmask 255.255.0.0`
	`  tcp_sockets 16`

User Datagram Protocol (UDP) Sockets

User Datagram Protocol (UDP) sockets are used by the LWPCON utility and by all applications that query a name server using the domain name system (DNS). DNS name servers are the only type of name server that the TCP/IP Transport for DOS software supports.

The software accesses a DNS name server through the information provided in the RESOLV.CFG file. See Chapter 2, “Setting Up Domain Name Systems (DNS) Support,” in the TCP/IP Transport for DOS Configuration Guide for more information.

Because UDP sockets are used briefly to send and receive datagrams and are then released, you need more than the default eight UDP sockets only if you make very heavy use of DNS.

UDP_SOCKETS number

Specifies the maximum number concurrent UDP sockets (connections).

If you are using DNS, set at least one UDP socket for each concurrently run application.

If you are using NetBIOS, set at least two UDP sockets.

If you are configuring multiple ODI LAN drivers for TCP/IP, set the number of UDP sockets to the total shared by all LAN drivers.

Syntax	udp_sockets *number*
Default	8
Range	0 to 32

Example	To set a maximum of sixteen concurrent UDP connections, you could place the following lines in your NET.CFG file:
	
	`protocol tcpip`
	`  ip_address 129.47.6.84`
	`  ip_router 144.52.6.6`
	`  ip_netmask 255.255.0.0`
	`  udp_sockets 16`

Raw Sockets

One raw socket is required when using the PING utility, the LWPCON utility, or other applications that have ping functions.

The default NET.CFG file configured by the TCP/IP Transport for DOS installation program provides for eight TCP sockets, eight UDP sockets, and one raw socket. These default values are sufficient for most configurations.

Run the TCP/IP Transport for DOS installation program provided on the TCP/IP Transport for DOS diskette to obtain a copy of the PING.EXE and LWPCON.EXE files. See Chapter 1, “Installing the Transport,” in the TCP/IP Transport for DOS Installation Guide for more information.

RAW_SOCKETS number

Specifies the maximum number of raw IP sockets (connections).

If you are using the PING utility, the LWPCON utility, or other utilities that have ping functions, set one raw IP socket.

If you are not using utilities that have ping functions, set the number of raw IP sockets to 0.

If you are configuring multiple ODI LAN drivers for TCP/IP, set the number of raw sockets to the total shared by all LAN drivers.

Syntax	raw_sockets *number*
Default	1

Example	To decrease memory used by IP sockets, you could place the following lines in your NET.CFG file:
	
	`protocol tcpip`
	`  ip_address 129.47.6.84`
	`  ip_router 144.52.6.6`
	`  ip_netmask 255.255.0.0`
	`  raw_sockets 0`

Additional Support

Use these parameters and values to configure your client workstation's boot process and path to client workstation configuration files, as discussed on the following pages:

Parameters and Values
NO_BOOTP PATH TCP_CFG [[drive:]path [; ...]]

NO_BOOTP

Directs your client workstation to bypass any network BOOTP (a standard protocol that provides TCP/IP configuration information) server and uses Reverse Address Resolution Protocol (RARP) to identify a client workstation's IP address.

If your network has a BOOTP server that is configured to supply your client workstation with its IP address, router address, and subnetwork mask, you don't need to configure this parameter in your NET.CFG file. See Chapter 2, "Using a BOOTP Server," in the TCP/IP Transport for DOS Configuration Guide for more information.

If you want to bypass a BOOTP server, specify values for the IP_ADDRESS, IP_ROUTER, and IP_NETMASK parameters in your NET.CFG file.

Syntax	no_bootp
Default	None

Example	To bypass a BOOTP server, you could place the following lines in your NET.CFG file:
	
	`protocol tcpip`
	`  no_bootp`
	`  ip_address 129.47.6.84`
	`  ip_router 144.52.6.6`
	`  ip_netmask 255.255.0.0`

PATH TCP_CFG [[drive:]path [; ...]]

Specifies the directories that contain the database configuration files HOSTS, NETWORKS, PROTOCOL, SERVICES, and RESOLV.CFG. The syntax is the same as the DOS PATH command.

Syntax	path tcp_cfg [[*drive:*] *path* [; ...]]
	Replace *drive* with the letter of the drive where the database configuration files exist.
	Replace *path* with the directory path to the location of the database configuration files.
Default	drive: C path: \net\tcp

Example	To specify the directory \TCPNET\TCP for the database configuration files, you could place the following lines in your NET.CFG file:
	
	`protocol tcpip`
	`  path c:\tcpnet\tcp;%path%`
	`  ip_address 129.47.6.84`
	`  ip_router 144.52.6.6`
	`  ip_netmask 255.255.0.0`

Transport Provider IPX | UDP Option

The Desktop SNMP transport providers, STPIPX.COM and STPUDP.COM, read the configuration file to discover trap targets on NetWare networks.

Use this option of the NET.CFG file to specify the trap target address for your SNMP desktops. For more information, see “Desktop SNMP Option.”

Available Parameters and Values for the Transport Provider IPX | UDP Option

This option has the following parameter and value, which are discussed on the following pages:

Parameter and Value
TRAP TARGET ipxaddress | ipaddress

TRANSPORT PROVIDER IPX | UDP

Specifies the Transport Provider option IPX or UDP configurations.

Syntax	transport provider ipx | udp *parameter_name value* Replace *parameter_name* with the name of the parameter you want to use. Replace *value* with the value setting which corresponds with the parameter name. Select either IPX or UDP as the transport protocol used by the management station.

Example	To set your client workstation for a management station of “ab123456:0123456789ab,” you would place the following lines in your NET.CFG file:
	
	`transport provider ipx`
	`  trap target ab123456:0123456789ab`

TRAP TARGET ipxaddress | ipaddress

Trap targets define the addresses that the SNMP manager sends SNMP traps to.

To receive traps sent by Desktop SNMP, make sure your management workstation address is listed in the IPX (for IPX transport) or UDP (for UDP/IP transport) section of the NET.CFG file.

NOTE: If you do not configure at least one trap target, Desktop SNMP does not send traps.

Syntax	trap target *ipxaddress* | *ipaddress*
	Replace *ipxaddress* or *ipaddress* with the management station address for IPX transport or for UDP / IP transport, respectively.
Default	None

Example	To set your client workstation for a management station of "ab123456:0123456789ab," you would place the following lines in your NET.CFG file:
	
	`transport provider ipx`
	`  trap target ab123456:0123456789ab`
	
	To set your client workstation to use both IPX and UDP / IP transport you could place the following lines in your NET.CFG file:
	
	`transport provider ipx`
	`  trap target ab123456:0123456789ab`
	`  trap target cd654321:ba9876543210`
	
	`transport provider udp`
	`  trap target 999.88.77.66`
	`  trap target 888.11.22.33`

3

Command Line Parameters Reference

Overview

This chapter contains a listing of command line parameters for the NetWare® Client™ for DOS and MS Windows software.

Topic
Core NetWare Client Software
DOSNP Software

Introduction

The command line parameters for the NetWare Client software are used to specify nondefault value settings for NetWare Client software configuration options.

Command line parameters require a specific syntax format. The following example and table explain these syntax format conventions.

```
PROGRAM_NAME /parameter [option] <Enter>
```

Convention	Explanation
PROGRAM_NAME	Uppercase words. Type these words as shown. (However, you can use uppercase or lowercase.)
[ ]	Square brackets. The item enclosed in brackets is optional: you can use the command with or without the item.
option	Italicized words. Replace them with the value that you want to use.
<Enter>	Angle brackets. Press the key whose name is enclosed in them.

Core NetWare Client Software

The core NetWare Client software for the DOS and MS Windows environment are four terminate-and stay-resident (TSR) programs, listed and described in the following table.

Table 3-1 Explanation of the Core NetWare Client Software

Software	Explanation
IPXODI.COM (Internetwork Packet Exchange™ Open Data-Link Interface™)	Delivers requests and replies between a client workstation and the network. Handles packet sequencing and acknowledgment for the client-server connection. Takes requests that the NetWare DOS Requester™ software has determined are for the network, "packages" them with transmission information (such as their destination), and hands them to the LSL™ program. Supports mobile IPX client workstations. Allows client workstations to respond to the NetWare Event Service Layer (NESL) software (a DOS TSR program that provides a means by which programs running on client workstations can generate or receive event notifications). Allows IPXODI to detect changes in network connections, dynamically bind or unbind to new network boards, and autoswitch between a primary and secondary network board when problems are detected.
LSL.COM Link Support Layer™ (LSL)	Puts the packaged requests from the IPXODI driver into the proper format for transmission on the particular physical network that the client workstation is running on. Takes replies for the client workstation from the network (via the network LAN driver), removes the network-specific information it has added, and passes the reply to IPXODI.
ODI LAN driver.COM (example: NE2000™)	Takes requests from the LSL and sends them to the network; receives replies from the network and passes them to the LSL. Is specific to the network board installed in your client workstation.

Table 3-1 Explanation of the Core NetWare Client Software

Software	Explanation
VLM.EXE NetWare DOS Requester	DOS client software that provides the interface between DOS and the network. Is loaded when you run the STARTNET.BAT file. Consists of individual modules that provide various network services. Loads drivers that the NetWare DOS Requester needs in order to communicate with the network hardware.

Use the command line parameters discussed in the following sections in order to set nondefault settings for the core NetWare Client software.

IPXODI.COM

Use the following command line parameters to manage the IPXODI.COM software.

Parameter	Option	Explanation
Configuration File	/C	Indicates alternate filename for configuration information. /C = [path\]filename
Eliminate Diagnostic Responder and SPX	/A	Eliminates diagnostic responder and SPX, reducing memory size by 9 KB.
Eliminate Diagnostic Responder	/D	Eliminates diagnostic responder, reducing memory size by 3 KB.
Force Unload	/F	Forcibly unloads IPXODI.COM file from memory. Might cause system to hang.
Help	/?	Displays help screen.

Parameter	Option	Explanation
Mobile IPX	/M	Loads diagnostic responder and SPX. Disables /D and /A values. Requires NESL.COM be loaded before loading IPXODI.
Unload	/U	Unloads IPXODI.COM file from memory.

The syntax for using these parameters is as follows:

```
IPXODI [option]
```

Following is an example of an IPXODI command that you could execute at a client workstation:

```
IPXODI /D /C=C:\NWCLIENT\NET.CFG
```

This command indicates the following:

- The diagnostic responder portion of the code does not load.
- The NetWare Client configuration file is NET.CFG in the C:\NWCLIENT directory.

LSL.EXE

Use the following command line parameters to manage the Link Support Layer software.

Parameter	Option	Explanation
Configuration File	/C	Indicates alternate filename for configuration information. /C = [path\]filename
Help	/?	Displays help screen.
Unload	/U	Unloads LSL.EXE file from memory.

The syntax for using these parameters is as follows:

```
LSL [option]
```

Following is an example of an LSL command that you could execute at a client workstation:

```
LSL /C=C:\NWCLIENT\NET.CFG
```

This command indicates the following:

- The NetWare Client configuration file is NET.CFG in the C:\NWCLIENT directory.

ODI LAN driver.COM

Use the following command line parameters to manage the LAN driver software.

Parameter	Option	Explanation
Help	/?	Displays help screen.
Unload	/U	Unloads ODI LAN driver.COM file from memory.

The syntax for using these parameters is as follows:

```
ODI_LAN_driver [option]
```

Following is an example of an ODI LAN driver command that you could execute at a client workstation for an NE2000™ ODI™ LAN driver:

```
NE2000 /U
```

This command indicates the following:

- The NE2000 LAN driver is unloaded from memory.

VLM.EXE

Use the following command line parameters to manage the NetWare DOS Requester software.

Parameter	Option	Explanation
Configuration File	/C	Indicates alternate filename for configuration information. /C = [path\]filename
Conventional Memory	/MC	Uses conventional memory when loading VLM.EXE file.
Diagnostics	/D	Displays VLM.EXE file diagnostics.
Expanded Memory	/ME	Uses expanded memory when loading VLM.EXE file.
Extended Memory	/MX	Uses extended memory when loading VLM.EXE file.
Help	/?	Displays help screen.
Message Level (verbosity)	/Vx	0 = Displays copyright message and critical errors. 1 = Displays warning messages. 2 = Displays program load information for VLM programs. 3 = Displays configuration information. 4 = Displays diagnostic information.
Preferred Server	/PS=	Specifies the preferred sever connection.
Preferred Tree	/PT=	Specifies the preferred tree connection.
Unload	/U	Unloads VLM.EXE file from memory.

The syntax for using these parameters is as follows:

```
VLM [option]
```

Following is an example of a VLM™ (NetWare DOS Requester) command that you could execute at a client workstation:

```
VLM /D /C=C:\NWCLIENT\NET.CFG /PS=SALES /MC
```

This command indicates the following:

- Diagnostic information is displayed.
- The NetWare Client configuration file is NET.CFG in the C:\NWCLIENT directory.
- SALES is the preferred server connection.
- The VLM software is loaded into conventional memory.

DOSNP Software

Use the following command line parameters for managing the DOSNP software.

Parameter	Option	Explanation
Help	/?	Displays help screen.
Unload	/U	Unloads file from memory.
Version	/I	Displays version and load information.

The syntax for using these parameters is as follows:

```
DOSNP [option]
```

Following is an example of a DOSNP command that you could execute at a client workstation:

```
DOSNP /U
```

This command indicates the following:

- The DOSNP program will be unloaded from memory.

4

System Messages

A system message points to a software or hardware error that doesn’t allow further processing.

An explanation of the nature of the message and a recommended course of action follow each message.

AUTO-1.00-1: The message file <name> is invalid. The program cannot be loaded.

Explanation: The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

Action: Either update the .MSG file with a valid copy or delete the file. The NetWare® DOS Requester™ software uses the default messages that are bound to the binary files.

AUTO-1.00-5: <Module1>.VLM is not loaded. The <module2>.VLM file cannot be loaded before <module1>.VLM. Load the <module1>.VLM file first then try to load the <module2>.VLM file.

Explanation: The <module2>.VLM file requires that the <module1>.VLM file be loaded first. Either the current configuration has <module1>.VLM and <module2>.VLM loading out of order, or <module1>.VLM did not load successfully.

Action: Make sure that <module1>.VLM is configured to load before <module2>.VLM. To do this, change the load order of the VLM= parameter in the NET.CFG file.

AUTO-1.00-7: In order to load the <module3>.VLM, one of the following VLMs must be loaded: <module1>.VLM, <module2>.VLM.

Explanation: The <module3>.VLM file requires that the <module1>.VLM or <module2>.VLM file be loaded first. Either the current configuration has <module3>.VLM loading before <module1>.VLM or <module2>.VLM, or <module1>.VLM or <module2>.VLM did not load successfully.

Action: Make sure that <module1>.VLM or <module2>.VLM loads successfully before <module3>.VLM.

AUTO-1.00-42: There is a missing or invalid value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

AUTO-1.00-43: There is a missing or invalid ON/OFF value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

AUTO-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

Explanation: A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

Action: Check and correct the parameter specified in the NET.CFG file. See Chapter 2, "NET.CFG Options Reference," for more information.

BIND-1.00-1: The message file <name> is invalid. The program cannot be loaded.

Explanation: The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

Action: Either update the .MSG file with a valid copy or delete the file. The NetWare® DOS Requester™ software uses the default messages that are bound to the binary files.

BIND-1.00-5: <Module1>.VLM is not loaded. The <module2>.VLM file cannot be loaded before <module1>.VLM. Load the <module1>.VLM file first then try to load the <module2>.VLM file.

Explanation: The <module2>.VLM file requires that the <module1>.VLM file be loaded first. Either the current configuration has <module1>.VLM and <module2>.VLM loading out of order, or <module1>.VLM did not load successfully.

Action: Make sure that <module1>.VLM is configured to load before <module2>.VLM. To do this, change the load order of the VLM= parameter in the NET.CFG file.

BIND-1.00-7: In order to load the <module3>.VLM, one of the following VLMs must be loaded: <module1>.VLM, <module2>.VLM.

Explanation: The <module3>.VLM file requires that the <module1>.VLM or <module2>.VLM file be loaded first. Either the current configuration has <module3>.VLM loading before <module1>.VLM or <module2>.VLM, or <module1>.VLM or <module2>.VLM did not load successfully.

Action: Make sure that <module1>.VLM or <module2>.VLM loads successfully before <module3>.VLM.

BIND-1.00-42: There is a missing or invalid value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

BIND-1.00-43: There is a missing or invalid ON/OFF value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

BIND-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

Explanation: A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

Action: Check and correct the parameter specified in the NET.CFG file. See Chapter 2, “NET.CFG Options Reference,” for more information.

CONN-1.00-1: The message file <name> is invalid. The program cannot be loaded.

Explanation: The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

Action: Either update the .MSG file with a valid copy or delete the file. The NetWare® DOS Requester™ software uses the default messages that are bound to the binary files.

CONN-1.00-5: <Module1>.VLM is not loaded. The <module2>.VLM file cannot be loaded before <module1>.VLM. Load the <module1>.VLM file first then try to load the <module2>.VLM file.

Explanation: The <module2>.VLM file requires that the <module1>.VLM file be loaded first. Either the current configuration has <module1>.VLM and <module2>.VLM loading out of order, or <module1>.VLM did not load successfully.

Action: Make sure that <module1>.VLM is configured to load before <module2>.VLM. To do this, change the load order of the VLM= parameter in the NET.CFG file.

CONN-1.00-7: In order to load the <module3>.VLM, one of the following VLMs must be loaded: <module1>.VLM, <module2>.VLM.

Explanation: The <module3>.VLM file requires that the <module1>.VLM or <module2>.VLM file be loaded first. Either the current configuration has <module3>.VLM loading before <module1>.VLM or <module2>.VLM, or <module1>.VLM or <module2>.VLM did not load successfully.

Action: Make sure that <module1>.VLM or <module2>.VLM loads successfully before <module3>.VLM.

CONN-1.00-42: There is a missing or invalid value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

CONN-1.00-43: There is a missing or invalid ON/OFF value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

CONN-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

Explanation: A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

Action: Check and correct the parameter specified in the NET.CFG file. See Chapter 2, "NET.CFG Options Reference," for more information.

CONN-1.00-50: DOS version is not 3.1 or later. The NetWare Requester for DOS cannot be loaded. Reboot your computer with DOS 3.1 or later; then load the NetWare Requester for DOS files.

Explanation: The NetWare® DOS Requester™ software requires DOS 3.1 or later in order to operate. Because the current DOS version is not 3.1 or later, the NetWare DOS Requester cannot be loaded.

Action: Upgrade the DOS version on your machine to 3.1 or later.

CONN-1.00-51: An older version of the shell is loaded. The NetWare Requester for DOS cannot be loaded. Unload the shell; then load the NetWare Requester for DOS files.

Explanation: The NetWare® DOS Requester™ software cannot be loaded with the NetWare shell. The NetWare shell has been loaded in your machine. For NetWare shell compatibility.

Action: Type "NETX /U" to unload the NetWare shell. If the version of the NetWare shell in use does not support the /U parameter, reboot the machine without loading the NetWare shell. See Chapter 2, "NET.CFG Options Reference," for more information.

CONN-1.00-52: The NetWare DOS Named Pipes Extender is currently loaded. The NetWare Requester for DOS cannot be loaded. Unload the NetWare DOS Named Pipes Extender; then load the NetWare Requester for DOS files.

Explanation: The NetWare® DOS Requester™ software cannot be loaded after the NetWare DOS Named Pipes Extender.

To use both the NetWare DOS Requester and NetWare DOS Named Pipes Extender, load the NetWare DOS Requester first.

Action: Type "DOSNP /U" to unload the DOS Named Pipes Extender; then load the NetWare DOS Requester before reloading the DOS Named Pipes Extender. See "Named Pipes Options" for more information about Named Pipes parameters in the NET.CFG file.

DOSCLINST-1.0-1: <Filename> could not be installed.

Explanation: The indicated file could not be installed on the destination drive because either the file was not found on the master diskettes or the INSTALL utility was unable to write to the INSTALL directory.

Action: Make sure that the file is on the master diskettes. Also make sure that the destination drive is not write-protected or has some other condition that would prevent the indicated file from being copied.

DOSCLINST-1.0-2: This program requires more disk space. Installation could not be completed.

Explanation: Adequate disk space is not available on the destination drive. More space must be available for the installation to be completed.

Action: Delete unnecessary files from the hard disk.

DOSCLINST-1.0-3: The INSTALL.OVL file could not be found.

Explanation: The INSTALL.OVL file could not be found. This file is an integral piece of the INSTALL utility; it must be in the same directory as the INSTALL.EXE file.

Action: Make sure that INSTALL.OVL is in the same directory as the INSTALL.EXE file; then run INSTALL again.

DOSCLINST-1.0-4: A read error occurred while the program was reading INSTALL.OVL.

Explanation: The file INSTALL.OVL has been corrupted. The INSTALL utility requires INSTALL.OVL for successful installation.

Action: Copy INSTALL.OVL from the master diskettes to the same directory as the INSTALL.EXE file; then run INSTALL again.

FIO-1.00-1: The message file <name> is invalid. The program cannot be loaded.

Explanation: The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

Action: Either update the .MSG file with a valid copy or delete the file. The NetWare® DOS Requester™ software uses the default messages that are bound to the binary files.

FIO-1.00-5: <Module1>.VLM is not loaded. The <module2>.VLM file cannot be loaded before <module1>.VLM. Load the <module1>.VLM file first then try to load the <module2>.VLM file.

Explanation: The <module2>.VLM file requires that the <module1>.VLM file be loaded first. Either the current configuration has <module1>.VLM and <module2>.VLM loading out of order, or <module1>.VLM did not load successfully.

Action: Make sure that <module1>.VLM is configured to load before <module2>.VLM. To do this, change the load order of the VLM= parameter in the NET.CFG file.

FIO-1.00-7: In order to load the <module3>.VLM, one of the following VLMs must be loaded: <module1>.VLM, <module2>.VLM.

Explanation: The <module3>.VLM file requires that the <module1>.VLM or <module2>.VLM file be loaded first. Either the current configuration has <module3>.VLM loading before <module1>.VLM or <module2>.VLM, or <module1>.VLM or <module2>.VLM did not load successfully.

Action: Make sure that <module1>.VLM or <module2>.VLM loads successfully before <module3>.VLM.

FIO-1.00-42: There is a missing or invalid value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

FIO-1.00-43: There is a missing or invalid ON/OFF value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

FIO-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

Explanation: A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

Action: Check and correct the parameter specified in the NET.CFG file. See Chapter 2, "NET.CFG Options Reference," for more information.

FIO-1.00-60: Too much cache is configured. The FIO.VLM file will reduce the cache blocks by <number> blocks and load successfully. Check the CACHE BUFFERS and BUFFER SIZE parameters in the NET.CFG file; then load the FIO.VLM file.

Explanation: The amount of cache available in DOS is limited by conventional memory. Neither of the parameters determining the amount of memory to be used, CACHE BUFFERS nor BUFFER SIZE, can be configured to use more than 64 KB of memory.

Action: Reduce the CACHE BUFFERS or BUFFER SIZE parameter in the NET.CFG file. See Chapter 2, “NET.CFG Options Reference,” for more information.

FIO-1.00-61: The IPX interface is not loaded. The FIO.VLM file will load successfully without packet burst support. Load the IPXODI v2.00 or later to use packet burst or add PB BUFFERS=0 to the NET.CFG file; then load FIO.VLM.

Explanation: This message is only a warning. The NetWare® DOS Requester™ software functions properly without the use of the Packet Burst™ protocol. Packet Burst, which is a part of the FIO.VLM file, requires IPXODI.COM version 2.00 or later to function properly. No IPX™ interface was loaded in this machine.

Action: If you want Packet Burst, make sure that IPXODI.COM version 2.00 or later is loaded before the NetWare DOS Requester. If you do not want Packet Burst, add PB BUFFERS=0 to the NetWare DOS Requester section of the NET.CFG file. See See Chapter 2, “NET.CFG Options Reference,” for more information.

FIO-1.00-62: The LSL interface is not loaded. The FIO.VLM file will load successfully without packet burst support. Load the LSL module or add PB BUFFERS=0 to the NET.CFG file; then load FIO.VLM.

Explanation: This message is only a warning. The NetWare® DOS Requester™ software functions properly without the use of the Packet Burst™ protocol. Packet Burst, which is a part of the FIO.VLM file, requires the LSL.COM file to be loaded to function properly.

Action: If you want Packet Burst, make sure that the LSL.COM file is loaded before the NetWare DOS Requester. If you do not want Packet Burst, add PB BUFFERS=0 to the NetWare DOS Requester section of the NET.CFG file. See Chapter 2, “NET.CFG Options Reference,” for more information.

FIO-1.00-63: The IPX socket for packet burst could not be opened. The FIO.VLM file will load successfully without packet burst support. configure the IPXODI.COM file for enough sockets in the NET.CFG file or add PB BUFFERS=0 to the NET.CFG file; then load the FIO.VLM file.

Explanation: This message is only a warning. The NetWare® DOS Requester™ software functions properly without the use of the Packet Burst™ protocol. Packet Burst, which is a part of the FIO.VLM file, requires an IPX™ socket to function properly. The request to open a socket failed, indicating that not enough IPX sockets are available.

Action: If you want Packet Burst, make sure that the IPX interface is configured for enough sockets in the NET.CFG file before loading the NetWare DOS Requester. If you do not want Packet Burst, add PB BUFFERS=0 to the NetWare DOS Requester section of the NET.CFG file. See Chapter 2, “NET.CFG Options Reference,” for more information.

FIO-1.00-64: The IPX interface is not version 2.00 or later. The FIO.VLM file will load successfully without packet burst support. Load the IPXODI.COM file version 2.00 or later or add PB BUFFERS=0 to the NET.CFG file; then load the FIO.VLM file.

Explanation: This message is only a warning. The NetWare® DOS Requester™ software functions properly without the use of the Packet Burst™ protocol. Packet Burst, which is a part of the FIO.VLM file, requires IPXODI.COM version 2.00 or later to function properly. No IPX™ interface version 2.00 or later was loaded in this machine.

Action: If you want Packet Burst, make sure that IPXODI.COM version 2.00 or late is loaded before the NetWare DOS Requester. If you do not want Packet Burst, add PB BUFFERS=0 to the NetWare DOS Requester section of the NET.CFG file. See Chapter 2, “NET.CFG Options Reference,” for more information.

FIO-1.00-65: The NetWare DOS Requester is being loaded in a MS Windows DOS box. The FIO.VLM file will load successfully without packet burst support.

Explanation: This message is only a warning. The NetWare® DOS Requester™ software functions properly without the use of the Packet Burst™ protocol. Packet Burst, which is a part of the FIO.VLM file, cannot function when loaded in an MS Windows DOS box.

Action: You can use Packet Burst if it is loaded before starting MS Windows. To do this, exit MS Windows and load the VLM™ files; then restart MS Windows.

GENERAL-1.00-1: The message file <name> is invalid. The program cannot be loaded.

Explanation: The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

Action: Either update the .MSG file with a valid copy or delete the file. The NetWare® DOS Requester™ software uses the default messages that are bound to the binary files.

GENERAL-1.00-5: <Module1>.VLM is not loaded. The <module2>.VLM file cannot be loaded before <module1>.VLM. Load the <module1>.VLM file first then try to load the <module2>.VLM file.

Explanation: The <module2>.VLM file requires that the <module1>.VLM file be loaded first. Either the current configuration has <module1>.VLM and <module2>.VLM loading out of order, or <module1>.VLM did not load successfully.

Action: Make sure that <module1>.VLM is configured to load before <module2>.VLM. To do this, change the load order of the VLM= parameter in the NET.CFG file.

GENERAL-1.00-7: In order to load the <module3>.VLM, one of the following VLMs must be loaded: <module1>.VLM, <module2>.VLM.

Explanation: The <module3>.VLM file requires that the <module1>.VLM or <module2>.VLM file be loaded first. Either the current configuration has <module3>.VLM loading before <module1>.VLM or <module2>.VLM, or <module1>.VLM or <module2>.VLM did not load successfully.

Action: Make sure that <module1>.VLM or <module2>.VLM loads successfully before <module3>.VLM.

GENERAL-1.00-42: There is a missing or invalid value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

GENERAL-1.00-43: There is a missing or invalid ON/OFF value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time VLM.EXE is loaded, delete or correct the line specified by the error message.

GENERAL-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

Explanation: A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

Action: Check and correct the parameter in the NET.CFG file. See Chapter 2, "NET.CFG Options Reference," for more information.

IPXNCP-1.00-1: The message file <name> is invalid. The program cannot be loaded.

Explanation: The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

Action: Either update the .MSG file with a valid copy or delete the file. The NetWare® DOS Requester™ software uses the default messages that are bound to the binary files.

IPXNCP-1.00-5: <Module1>.VLM is not loaded. The <module2>.VLM file cannot be loaded before <module1>.VLM. Load the <module1>.VLM file first then try to load the <module2>.VLM file.

Explanation: The <module2>.VLM file requires that the <module1>.VLM file be loaded first. Either the current configuration has <module1>.VLM and <module2>.VLM loading out of order, or <module1>.VLM did not load successfully.

Action: Make sure that <module1>.VLM is configured to load before <module2>.VLM. To do this, change the load order of the VLM= parameter in the NET.CFG file.

IPXNCP-1.00-7: In order to load the <module3>.VLM, one of the following VLMs must be loaded: <module1>.VLM, <module2>.VLM.

Explanation: The <module3>.VLM file requires that the <module1>.VLM or <module2>.VLM file be loaded first. Either the current configuration has <module3>.VLM loading before <module1>.VLM or <module2>.VLM, or <module1>.VLM or <module2>.VLM did not load successfully.

Action: Make sure that <module1>.VLM or <module2>.VLM loads successfully before <module3>.VLM.

IPXNCP-1.00-42: There is a missing or invalid value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

IPXNCP-1.00-43: There is a missing or invalid ON/OFF value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

IPXNCP-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

Explanation: A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

Action: Check and correct the parameter specified in the NET.CFG file. See Chapter 2, “NET.CFG Options Reference,” for more information.

IPXNCP-1.00-53: The IPX interface is not loaded. The IPXNCP.VLM file cannot be loaded. Load the IPXODI.COM file first and then try loading the IPXNCP.VLM file.

Explanation: An attempt was made to load the IPXNCP.VLM file without having previously loaded the IPX™ interface.

Action: Load the IPXODI.COM file before attempting to load IPXNCP.VLM.

IPXNCP-1.00-54: The IPX sockets could not be opened. The IPXNCP.VLM file cannot be loaded. Configure the IPXODI.COM file for enough sockets in the NET.CFG file and then try to load the IPXNCP.VLM file.

Explanation: The IPXNCP.VLM file failed to open the IPX™ sockets needed in order to run. The IPXNCP.VLM file requires that four or more IPX sockets be available to run.

Action: Increase the number of IPX sockets available by using the IPX SOCKETS= parameter in the NET.CFG file; then reload the IPXODI.COM and IPXNCP.VLM files.

IPXNCP-1.00-55: The IPX interface does not support checksums. The IPXNCP.VLM file will load successfully without using checksums. Make sure the installed IPXODI. COM is version 2.01 or later and that it is not bound to a board configured to use the ETHERNET_802.3 frame format.

Explanation: The IPXNCP.VLM file loads without supporting checksums. For checksum to be supported in IPXNCP.VLM, the loaded IPXODI.COM file must support checksums. Either the loaded version of the IPXODI.COM file does not have checksum support or the protocol bound to the IPX™ software does not support checksums.

Action: No action required at this point. However, to avoid receiving this message the next time the IPXNCP.VLM file is loaded, either load the IPXODI.COM file with checksum support enabled or add CHECKSUM=OFF to the NetWare DOS Requester section of the NET.CFG file. See Chapter 2, “NET.CFG Options Reference,” for more information.

IPXNCP-1.00-56: The IPX socket for large internet packets could not be opened. The IPXNCP.VLM file will load successfully without using large internet packets. Configure the IPXODI.COM file for enough sockets in the NET.CFG file or add LARGE INTERNET PACKETS=OFF to the NET.CFG file; then load the IPXNCP.VLM file.

Explanation: This message is only a warning. The NetWar® DOS Requester™ software functions properly without the use of large internet packets. The large internet packet protocol, which is a part of the IPXNCP.VLM module, requires an IPX socket to function properly. The request to open a socket failed, indicating that not enough IPX sockets are available.

Action: If you want large internet packets, make sure that the IPX interface is configured for enough sockets in the NET.CFG file before loading the NetWare DOS Requester. If you do not want large internet packets, add LARGE INTERNET PACKETS=OFF to the NetWare DOS Requester section of the NET.CFG file.See Chapter 2, “NET.CFG Options Reference,” for more information.

NDS-1.00-1: The message file <name> is invalid. The program cannot be loaded.

Explanation: The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

Action: Either update the .MSG file with a valid copy or delete the file. The NetWare® DOS Requester™ software uses the default messages that are bound to the binary files.

NDS-1.00-5: <Module1>.VLM is not loaded. The <module2>.VLM file cannot be loaded before <module1>.VLM. Load the <module1>.VLM file first then try to load the <module2>.VLM file.

Explanation: The <module2>.VLM file requires that the <module1>.VLM file be loaded first. Either the current configuration has <module1>.VLM and <module2>.VLM loading out of order, or <module1>.VLM did not load successfully.

Action: Make sure that <module1>.VLM is configured to load before <module2>.VLM. To do this, change the load order of the VLM= parameter in the NET.CFG file.

NDS-1.00-7: In order to load the <module3>.VLM, one of the following VLMs must be loaded: <module1>.VLM, <module2>.VLM.

Explanation: The <module3>.VLM file requires that the <module1>.VLM or <module2>.VLM file be loaded first. Either the current configuration has <module3>.VLM loading before <module1>.VLM or <module2>.VLM, or <module1>.VLM or <module2>.VLM did not load successfully.

Action: Make sure that <module1>.VLM or <module2>.VLM loads successfully before <module3>.VLM.

NDS-1.00-42: There is a missing or invalid value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

NDS-1.00-43: There is a missing or invalid ON/OFF value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

NDS-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

Explanation: A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

Action: Check and correct the parameter specified in the NET.CFG file. See Chapter 2, "NET.CFG Options Reference," for more information.

NETX-1.00-1: The message file <name> is invalid. The program cannot be loaded.

Explanation: The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

Action: Either update the .MSG file with a valid copy or delete the file. The NetWare® DOS Requester™ software uses the default messages that are bound to the binary files.

NETX-1.00-5: <Module1>.VLM is not loaded. The <module2>.VLM file cannot be loaded before <module1>.VLM. Load the <module1>.VLM file first then try to load the <module2>.VLM file.

Explanation: The <module2>.VLM file requires that the <module1>.VLM file be loaded first. Either the current configuration has <module1>.VLM and <module2>.VLM loading out of order, or <module1>.VLM did not load successfully.

Action: Make sure that <module1>.VLM is configured to load before <module2>.VLM. To do this, change the load order of the VLM= parameter in the NET.CFG file.

NETX-1.00-7: In order to load the <module3>.VLM, one of the following VLMs must be loaded: <module1>.VLM, <module2>.VLM.

Explanation: The <module3>.VLM file requires that the <module1>.VLM or <module2>.VLM file be loaded before <module3>.VLM can run effectively. Either the current configuration has <module3>.VLM loading before <module1>.VLM or <module2>.VLM, or <module1>.VLM or <module2>.VLM did not load successfully.

Action: Make sure that <module1>.VLM or <module2>.VLM loads successfully before <module3>.VLM.

NETX-1.00-42: There is a missing or invalid value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

NETX-1.00-43: There is a missing or invalid ON/OFF value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

NETX-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

Explanation: A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

Action: Check and correct the parameter specified in the NET.CFG file. See Chapter 2, “NET.CFG Options Reference,” for more information.

NETX-1.00-57: DOS is only configured for <number> drives, NETX.VLM requires 26 drives for full functionality. The NETX.VLM file will load with partial support. Add LASTDRIVE=Z to the CONFIG.SYS file and reboot the workstation; then load the NETX.VLM file.

Explanation: This message is only a warning. However, if you want full compatibility with the NetWare® shell, you must set the DOS LASTDRIVE parameter in the CONFIG.SYS file to Z. Otherwise, problems occur when mapping drives, etc.

Action: Set LASTDRIVE=Z in the CONFIG.SYS file; then reboot the client workstation before loading the NetWare DOS Requester™ software.

NETX-1.00-58: The PRINT.VLM file has not been loaded. The NETX.VLM file will load successfully without print services. To enable printing services, load the PRINT.VLM file before loading the NETX.VLM file.

Explanation: This message is only a warning. For full NetWare® shell compatibility, the PRINT.VLM file must be loaded.

Action: If you want print functionality, load the PRINT.VLM file before loading the NETX.VLM file.

NMR-1.00-1: The message file <name> is invalid. The program cannot be loaded.

Explanation: The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

Action: Either update the .MSG file with a valid copy or delete the file. The NetWare® DOS Requester™ software uses the default messages that are bound to the binary files.

NMR-1.00-5: <Module1>.VLM is not loaded. The <module2>.VLM file cannot be loaded before <module1>.VLM. Load the <module1>.VLM file first then try to load the <module2>.VLM file.

Explanation: The <module2>.VLM file requires that the <module1>.VLM file be loaded first. Either the current configuration has <module1>.VLM and <module2>.VLM loading out of order, or <module1>.VLM did not load successfully.

Action: Make sure that <module1>.VLM is configured to load before <module2>.VLM. To do this, change the load order of the VLM= parameter in the NET.CFG file.

NMR-1.00-7: In order to load the <module3>.VLM, one of the following VLMs must be loaded: <module1>.VLM, <module2>.VLM.

Explanation: The <module3>.VLM file requires that the <module1>.VLM or <module2>.VLM file be loaded first. Either the current configuration has <module3>.VLM loading before <module1>.VLM or <module2>.VLM, or <module1>.VLM or <module2>.VLM did not load successfully.

Action: Make sure that <module1>.VLM or <module2>.VLM loads successfully before <module3>.VLM.

NMR-1.00-42: There is a missing or invalid value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

NMR-1.00-43: There is a missing or invalid ON/OFF value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

NMR-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

Explanation: A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

Action: Check and correct the parameter specified in the NET.CFG file. See Chapter 2, “NET.CFG Options Reference,” for more information.

NWDRV-3.00-00: The NetWare VLM is not loaded or is not configured correctly. All network functions will be disabled. Reconfigure the VLM and restart MS Windows.

Explanation: Without the NetWare® DOS Requester™ (VLM) software, the client workstation cannot communicate with NetWare servers. The NetWare MS Windows driver (NETWARE.DRV) depends on the VLM™ software in order to provide network support for MS Windows.

Action: Exit MS Windows. Go to the directory where the VLM.EXE file is located and type “VLM”. Then restart MS Windows.

NWDRV-3.00-10: The VNETWARE.386 file has not been loaded. All network functions will be disabled. To use NetWare, install VNETWARE.386 and restart MS Windows.

Explanation: The NetWare® MS Windows driver (NETWARE.DRV) depends on the VNETWARE.386 file to run MS Windows in enhanced mode.

Action: Exit MS Windows. Copy the VNETWARE.386 file to the MS Windows SYSTEM (for example, C:\WINDOWS\SYSTEM). VNETWARE.386 should be on the NetWare Client For DOS and MS Windows Disk 3 diskette. Also make sure that the “[386Enh]” section of the SYSTEM.INI file has the following line: NETWORK=*VNETBIOS, VNETWARE.386, VIPX.386. Then restart MS Windows.

NWDRV-3.00-50: The temporary drive could not be mapped. Contact the network supervisor.

Explanation: The NetWare® MS Windows driver (NETWARE.DRV) allocates temporary drives on the server to perform some functions.

Action: Contact the network supervisor to make sure that NetWare is running and the server is operating correctly.

NWP-1.00-1: The message file <name> is invalid. The program cannot be loaded.

Explanation: The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

Action: Either update the .MSG file with a valid copy or delete the file. The NetWare® DOS Requester™ software uses the default messages that are bound to the binary files.

NWP-1.00-5: <module1>.VLM is not loaded. The <module2>.VLM file cannot be loaded before <module1>.VLM. Load the <module1>.VLM file first then try to load the <module2>.VLM file.

Explanation: The <module2>.VLM file requires that the <module1>.VLM file be loaded. Either the current configuration has <module1>.VLM and <module2>.VLM loading out of order, or <module1>.VLM did not load successfully.

Action: Make sure that <module1>.VLM is configured to load before <module2>.VLM. To do this, change the load order of the VLM= parameter in the NET.CFG file.

NWP-1.00-7: In order to load the <module3>.VLM, one of the following VLMs must be loaded: <module1>.VLM, <module2>.VLM.

Explanation: The <module3>.VLM file requires that the <module1>.VLM or <module2>.VLM file be loaded. Either the current configuration has <module3>.VLM loading before <module1>.VLM or <module2>.VLM, or <module1>.VLM or <module2>.VLM did not load successfully.

Action: Make sure that <module1>.VLM or <module2>.VLM loads successfully before <module3>.VLM.

NWP-1.00-42: There is a missing or invalid value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

NWP-1.00-43: There is a missing or invalid ON/OFF value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

NWP-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

Explanation: A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

Action: Check and correct the parameter specified in the NET.CFG file. See Chapter 2, "NET.CFG Options Reference," for more information.

NWP-1.00-59: The SECURITY.VLM file has not been loaded. The NWP.VLM file will load successfully without NCP signature support. Load the SECURITY.VLM file before loading the NWP.VLM file or add SIGNATURE LEVEL=0 to the NET.CFG file; then load NWP.VLM.

Explanation: This message is only a warning. The NetWare® DOS Requester™ software functions properly without NetWare Core Protocol™ (NCP) authentication. For NCP™ authentication to operate properly, the SECURITY.VLM file must be loaded before the NWP.VLM file. Either NWP.VLM is being loaded before SECURITY.VLM or SECURITY.VLM did not load successfully.

Action: If you want NCP authentication, make sure that SECURITY.VLM is configured to load before NWP.VLM and that it loads successfully before attempting to load NWP.VLM. If you do not want NCP authentication, add SIGNATURE LEVEL=0 to the NetWare DOS Requester section of the NET.CFG file. See Chapter 2, "NET.CFG Options Reference," for more information.

REDIR-1.00-1: The message file <name> is invalid. The program cannot be loaded.

Explanation: The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

Action: Either update the .MSG file with a valid copy or delete the file. The NetWare® DOS Requester™ software uses the default messages that are bound to the binary files.

REDIR-1.00-5: <Module1>.VLM is not loaded. The <module2>.VLM file cannot be loaded before <module1>.VLM. Load the <module1>.VLM file first then try to load the <module2>.VLM file.

Explanation: The <module2>.VLM file requires that the <module1>.VLM file be loaded. Either the current configuration has <module1>.VLM and <module2>.VLM loading out of order, or <module1>.VLM did not load successfully.

Action: Make sure that <module1>.VLM is configured to load before <module2>.VLM. To do this, change the load order of the VLM= parameter in the NET.CFG file.

REDIR-1.00-7: In order to load the <module3>.VLM, one of the following VLMs must be loaded: <module1>.VLM, <module2>.VLM.

Explanation: The <module3>.VLM file requires that the <module1>.VLM or <module2>.VLM file be loaded. Either the current configuration has <module3>.VLM loading before <module1>.VLM or <module2>.VLM, or <module1>.VLM or <module2>.VLM did not load successfully.

Action: Make sure that <module1>.VLM or <module2>.VLM loads successfully before <module3>.VLM.

REDIR-1.00-42: There is a missing or invalid value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

REDIR-1.00-43: There is a missing or invalid ON/OFF value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

REDIR-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

Explanation: A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

Action: Check and correct the parameter specified in the NET.CFG file. See Chapter 2, "NET.CFG Options Reference," for more information.

RSA-1.00-1: The message file <name> is invalid. The program cannot be loaded.

Explanation: The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

Action: Either update the .MSG file with a valid copy or delete the file. The NetWare® DOS Requester™ software uses the default messages that are bound to the binary files.

RSA-1.00-5: <Module1>.VLM is not loaded. The <module2>.VLM file cannot be loaded before <module1>.VLM. Load the <module1>.VLM file first then try to load the <module2>.VLM file.

Explanation: The <module2>.VLM file requires that the <module1>.VLM file be loaded first. Either the current configuration has <module1>.VLM and <module2>.VLM loading out of order, or <module1>.VLM did not load successfully.

Action: Make sure that <module1>.VLM is configured to load before <module2>.VLM. To do this, change the load order of the VLM= parameter in the NET.CFG file.

RSA-1.00-7: In order to load the <module3>.VLM, one of the following VLMs must be loaded: <module1>.VLM, <module2>.VLM.

Explanation: The <module3>.VLM file requires that the <module1>.VLM or <module2>.VLM file be loaded first. Either the current configuration has <module3>.VLM loading before <module1>.VLM or <module2>.VLM, or <module1>.VLM or <module2>.VLM did not load successfully.

Action: Make sure that <module1>.VLM or <module2>.VLM loads successfully before <module3>.VLM.

RSA-1.00-42: There is a missing or invalid value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

RSA-1.00-43: There is a missing or invalid ON/OFF value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

RSA-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

Explanation: A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

Action: Check and correct the parameter specified in the NET.CFG file. See Chapter 2, “NET.CFG Options Reference,” for more information.

SECURITY-1.00-1: The message file <name> is invalid. The program cannot be loaded.

Explanation: The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

Action: Either update the .MSG file with a valid copy or delete the file. The NetWare® DOS Requester™ software uses the default messages that are bound to the binary files.

SECURITY-1.00-5: <Module1>.VLM is not loaded. The <module2>.VLM file cannot be loaded before <module1>.VLM. Load the <module1>.VLM file first then try to load the <module2>.VLM file.

Explanation: The <module2>.VLM file requires that the <module1>.VLM file be loaded first. Either the current configuration has <module1>.VLM and <module2>.VLM loading out of order, or <module1>.VLM did not load successfully.

Action: Make sure that <module1>.VLM is configured to load before <module2>.VLM. To do this, change the load order of the VLM= parameter in the NET.CFG file.

SECURITY-1.00-7: In order to load the <module3>.VLM, one of the following VLMs must be loaded: <module1>.VLM, <module2>.VLM.

Explanation: The <module3>.VLM file requires that the <module1>.VLM or <module2>.VLM file be loaded first. Either the current configuration has <module3>.VLM loading before <module1>.VLM or <module2>.VLM, or <module1>.VLM or <module2>.VLM did not load successfully.

Action: Make sure that <module1>.VLM or <module2>.VLM loads successfully before <module3>.VLM.

SECURITY-1.00-42: There is a missing or invalid value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

SECURITY-1.00-43: There is a missing or invalid ON/OFF value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time VLM.EXE is loaded, delete or correct the line specified by the error message.

SECURITY-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

Explanation: A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

Action: Check and correct the parameter specified in the NET.CFG file. See Chapter 2, "NET.CFG Options Reference," for more information.

SECURITY-1.00-67: The SECURITY.VLM file must be loaded before any other NetWare Protocol Module. The SECURITY.VLM file will not be loaded. Load the SECURITY.VLM file before loading any NetWare Protocol Modules.

Explanation: This message is only a warning. The NetWare® DOS Requester™ software functions properly without the NetWare Core Protocol™ (NCP) authentication. The SECURITY.VLM file must be loaded before any NetWare protocol modules (BIND.VLM or NDS.VLM) to function properly.

Action: If you want NCP™ authentication, load the SECURITY.VLM after TRAN.VLM and before any NetWare protocol module (BIND.VLM or NDS.VLM); then set the SIGNATURE LEVEL= parameter in the NET.CFG file. If you do not want NCP authentication, set SIGNATURE LEVEL=0 in the NetWare DOS Requester section of NET.CFG file. See Chapter 2, "NET.CFG Options Reference," for more information.

TRAN-1.00-1: The message file <name> is invalid. The program cannot be loaded.

Explanation: The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

Action: Either update the .MSG file with a valid copy or delete the file. The NetWare® DOS Requester™ software uses the default messages that are bound to the binary files.

TRAN-1.00-5: <Module1>.VLM is not loaded. The <module2>.VLM file cannot be loaded before <module1>.VLM. Load the <module1>.VLM file first then try to load the <module2>.VLM file.

Explanation: The <module2>.VLM file requires that the <module1>.VLM file be loaded first. Either the current configuration has <module1>.VLM and <module2>.VLM loading out of order, or <module1>.VLM did not load successfully.

Action: Make sure that <module1>.VLM is configured to load before <module2>.VLM. To do this, change the load order of the VLM= parameter in the NET.CFG file.

TRAN-1.00-7: In order to load the <module3>.VLM, one of the following VLMs must be loaded: <module1>.VLM, <module2>.VLM.

Explanation: The <module3>.VLM file requires that the <module1>.VLM or <module2>.VLM file be loaded first. Either the current configuration has <module3>.VLM loading before <module1>.VLM or <module2>.VLM, or <module1>.VLM or <module2>.VLM did not load successfully.

Action: Make sure that <module1>.VLM or <module2>.VLM loads successfully before <module3>.VLM.

TRAN-1.00-42: There is a missing or invalid value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

TRAN-1.00-43: There is a missing or invalid ON/OFF value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

TRAN-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

Explanation: A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

Action: Check and correct the parameter specified in the NET.CFG file. See Chapter 2, "NET.CFG Options Reference," for more information.

VLM-1.00-1: The message file <name> is invalid. The program cannot be loaded.

Explanation: The .MSG file is invalid. This problem could be the result of a corrupted file, a bad translation, or an outdated file version.

Action: Either update the .MSG file with a valid copy or delete the file. The NetWare® DOS Requester™ software uses the default messages that are bound to the binary files.

VLM-1.00-8: The VLM.EXE file is already loaded. VLM.EXE cannot be reloaded. If you want to load VLM.EXE with a different configuration, unload VLM.EXE first with the /U parameter and then try loading the VLM.EXE file.

Explanation: The VLM.EXE file has already been loaded into memory. There cannot be two copies of the VLM.EXE file loaded in memory at one time.

Action: If the VLM.EXE file is being reloaded in an attempt to reconfigure the VLM™ program already loaded, you must first unload VLM.EXE using the /U parameter; then reload VLM.EXE.

VLM-1.00-9: The VLM.EXE file is testing the VLMs.

Explanation: This message is used to show the progress the VLM.EXE file is making in configuring the VLM™ programs to be loaded. Because this process can take a few seconds, the progress is displayed using a period (.) to indicate each VLM being tested.

Action: No action is required; this is simply an information message.

VLM-1.00-10: The conventional memory block for the EMS stack cannot be resized. The VLM.EXE file cannot be loaded. DOS memory is probably corrupt. Reboot your computer and then try to load the VLM.EXE again.

Explanation: This error indicates that the DOS memory chain has been corrupted.

Action: It is unsafe to continue operating the computer. The best solution is to reboot the computer; then try to load the VLM.EXE file.

VLM-1.00-11: There is insufficient memory to load the VLMs. The VLM.EXE file cannot be loaded. Reconfigure the VLMs to be loaded in the NET.CFG file and then try to load the VLM.EXE file.

Explanation: The VLM.EXE file can use expanded memory (EMS), extended memory (XMS), or conventional memory. The VLM.EXE file has determined that there is not enough of any one of these types of memory to support loading the VLM™ files.

Action: Check the amount of all memory available. Some memory managers provide dynamic memory pools by converting extended memory to expanded memory or vice versa. Other memory managers require you to configure the amount and type of memory you want. In either case, do one of the following:

- Make sure there is enough memory of any one type to support the VLM files.
- Reduce the number of VLM files to be loaded, either by using the EXCLUDE= parameter in the NET.CFG file or by renaming the files so the VLM.EXE file does not recognize the .VLM file that you do not want loaded.

VLM-1.00-15: An invalid command line parameter was specified.

Explanation: The VLM.EXE file was loaded with an invalid command line parameter.

Action: Run VLM.EXE with the /? parameter (for example, type "VLM /?") to display the valid parameters; then load the VLM.EXE file with a valid command line parameter.

VLM-1.00-16: The VLM.EXE file <version> is currently loaded.

Explanation: This message is displayed during the VLM.EXE file diagnostics display. (The diagnostics display is accessed by running the VLM.EXE file with the /D parameter.)

Action: No action is required; this is simply an information message.

VLM-1.00-17: The VLM.EXE file is not currently loaded.

Explanation: This message is displayed during the VLM.EXE file diagnostics display. (The diagnostics display is accessed by running the VLM.EXE file with the /D parameter.)

Action: No action is required; this is simply an information message.

VLM-1.00-30: A task switcher has been detected in memory. The VLM.EXE file cannot be loaded under a task switcher. Exit the task switcher; then try again.

Explanation: The NetWare® DOS Requester™ software cannot function properly when loaded after a task switcher. A task switcher has been previously loaded. (Task switchers include DR DOS® TaskMax, MS-DOS* 5.0 DOSSHELL, and MS Windows 3.1 in standard mode.)

Action: Unload the task switcher before attempting to load the NetWare DOS Requester.

VLM-1.00-31: Network error on server <name>. Check network cabling or server status.

Explanation: This is a critical communications error between the client workstation and the NetWare® server. This error may also appear as "General failure on device NETWORK." A communications error can be caused by either a hardware or software failure.

Action: Check the hardware and make sure that the network cable is connected properly and that the NetWare server is up and running properly. If the condition persists, report the problem to your Novell Authorized ResellerCLM representative.

VLM-1.00-32: The VLM.EXE file is not loaded. VLM.EXE cannot be unloaded.

Explanation: An attempt was made to unload the VLM.EXE file using the /U parameter.

Action: The condition that caused this message to occur is invalid, so no action is required.

VLM-1.00-33: The loaded VLM.EXE file has a different version. VLM.EXE cannot be unloaded. Make sure the VLM.EXE file you are using has the same version number and then try to unload VLM.EXE.

Explanation: An attempt was made to unload the VLM.EXE file using the /U parameter.

Action: Use the same version of the VLM.EXE file to try the unload as was used to load; otherwise, the VLM.EXE file that is loaded in memory can be removed only by rebooting the computer.

VLM-1.00-34: The loaded VLM.EXE file indicates it is unsafe to execute an unload for VLM number <number>. VLM.EXE will not be unloaded. Unload all memory resident programs (TSRs) that were loaded after the VLM.EXE file and then try to unload VLM.EXE.

Explanation: An attempt was made to unload the VLM.EXE file using the /U parameter. The loaded VLM.EXE file refused to unload. This usually is caused by interrupts being used after the VLM.EXE file was loaded. Another possible cause is that a VLM™ program is in an unsafe state to unload.

Action: Unload all the TSR (terminate-and-stay-resident) programs that were loaded after the VLM.EXE file; then try to unload VLM.EXE. Make sure all loaded VLM programs are in use before unloading.

VLM-1.00-36: The VLM.EXE file cannot use expanded memory (EMS). VLM.EXE will use an alternate memory scheme.

Explanation: An attempt was made to load the VLM.EXE file in expanded memory (EMS) using the /ME parameter. Either no LIM EMS 4.0 expanded memory manager is present or insufficient expanded memory is available to load the VLM™ program. The VLM.EXE file attempts to use an alternate type of memory, either extended (XMS) or conventional, before failing to load.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, either make sure an expanded memory manager is loaded with sufficient memory, or do not specify the /ME parameter when loading VLM.EXE.

VLM-1.00-37: The VLM.EXE file is using expanded memory (EMS).

Explanation: This message indicates that the VLM.EXE file is loading the VLM™ programs into expanded memory (EMS).

Action: No action is required; this is simply an information message.

VLM-1.00-38: The VLM.EXE file cannot use extended memory (XMS). VLM.EXE will use an alternate memory scheme.

Explanation: An attempt was made to load the VLM.EXE file in extended memory (XMS) using the /MX parameter. Either no XMS 2.0 extended memory manager is present or insufficient extended memory is available to

load the VLM™ programs. The VLM.EXE file attempts to use an alternate type of memory, either expanded (EMS) or conventional, before failing to load.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, either make sure an extended memory manager is loaded with sufficient memory, or do not specify the /MX parameter when loading VLM.EXE.

VLM-1.00-39: The VLM.EXE file is using extended memory (XMS).

Explanation: This message indicates that the VLM.EXE file is loading the VLM™ programs into extended memory (XMS).

Action: No action is required; this is simply an information message.

VLM-1.00-40: The VLM.EXE file cannot use conventional memory. VLM.EXE will use an alternate memory scheme.

Explanation: An attempt was made to load the VLM.EXE file in conventional memory using the /MC parameter. Insufficient conventional memory is available to load the VLM™ programs. The VLM.EXE file attempts to use an alternate type of memory, either expanded (EMS) or extended (XMS), before failing to load.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, either make sure there is sufficient conventional memory, or do not specify the /MC parameter when loading VLM.EXE.

VLM-1.00-41: The VLM.EXE file is using conventional memory.

Explanation: This message indicates that the VLM.EXE file is loading the VLM™ programs into conventional memory.

Action: No action is required; this is simply an information message.

VLM-1.00-42: There is a missing or invalid value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that has an invalid value or that does not specify a value. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

VLM-1.00-43: There is a missing or invalid ON/OFF value for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that is defined to use either an ON or OFF value, but a different value or no value has been specified. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

VLM-1.00-44: There is an invalid string length specified for '<parameter>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter that is defined to use a string of a specified minimum or maximum length, but a string has been entered that does not fit the specified limits. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

VLM-1.00-45: The parameter specified for the following option was out of range and has been adjusted.

Explanation: A configurable parameter has been configured too high or too low to be valid. The parameter is specified on the line following the message.

Action: Check and correct the parameter specified in the NET.CFG file. See Chapter 2, "NET.CFG Options Reference," for more information.

VLM-1.00-46: A duplicate VLM ID was found during VLM load test. The file will not be loaded. check the VLM= and USE DEFAULTS= parameters specified in the NET.CFG file before continuing.

Explanation: The VLM™ programs for the NetWare® DOS Requester™ software have been improperly configured to include two programs with the same VLM ID. This can be caused by either a VLM being included twice in the VLM= parameter list, or by a VLM attempting to reuse a VLM ID assigned to a different VLM.

Action: Check the NET.CFG file for duplicates in the VLM list. Make sure that the USE DEFAULTS= parameter is set to OFF. See Chapter 2, "NET.CFG Options Reference," for more information.

VLM-1.00-47: A file server could not be found. Check the network cabling and the server's status before continuing.

Explanation: No NetWare® server was found when attempting to build an initial connection. This could be caused by any of the following:

- An improperly configured network board or LAN driver
- Improperly configured IPX™ software
- A disconnected network cable
- An attempt to load the NetWare DOS Requester™ software before a NetWare server has been initialized.

Try one or more of the following:

- Ensure that the network board and LAN driver are properly configured by checking the network board settings and the NET.CFG file settings.

  See Chapter 2, "NET.CFG Options Reference," for more information.
- Ensure that IPX is configured to be bound properly to the right LAN driver.
- Check the network cabling.
- Ensure that a NetWare server is up and running before attempting to load the NetWare DOS Requester.

VLM-1.00-60: There is an unrecognized entry '<entry>' on line <number> of the configuration file. This entry will be ignored. Correct the line specified in the configuration file before continuing.

Explanation: The configuration file contains a parameter under a header where it does not belong. This invalid line is ignored by the configuration process.

Action: No action is required at this point. However, to avoid this message the next time the VLM.EXE file is loaded, delete or correct the line specified by the error message.

Index

A

ALTERNATE Link driver option parameter, explained, 2-28
Asynchronous Timeout Connections (Desktop SNMP NET.CFG option category) parameters, listed and explained 2-14. See also specific parameter name
ASYNCHRONOUS TIMEOUT Desktop SNMP option parameter, explained, 2-14
AUTO LARGE TABLE NetWare DOS Requester option parameter, explained, 2-63
AUTO RECONNECT NetWare DOS Requester option parameter, explained, 2-64
AUTO RETRY NetWare DOS Requester option parameter, explained, 2-64
AVERAGE NAME LENGTH NetWare DOS Requester option parameter, explained, 2-65

B

BIND Protocol IPX option parameter, explained, 2-104
BIND Protocol TCPIP option parameter, explained, 2-116
BIND RECONNECT NetWare DOS Requester option parameter, explained, 2-66
BROADCAST RETRIES NetWare DOS Requester option parameter, explained, 2-66
BROADCAST SEND DELAY NetWare DOS Requester option parameter, explained, 2-67
BROADCAST TIMEOUT NetWare DOS Requester option parameter, explained, 2-67
BUFFERS Link support option parameter, explained, 2-47
BUS ID Link driver option parameter, explained, 2-28, 2-29

C

CACHE BUFFER SIZE NetWare DOS Requester option parameter, explained, 2-68
CACHE BUFFERS NetWare DOS Requester option parameter, explained, 2-69
CACHE WRITES NetWare DOS Requester option parameter, explained, 2-69
Categories, listed and explained. See also specific category names
  Desktop SNMP NET.CFG option, 2-13
  Protocol TCPIP NET.CFG option, 2-114
CHECKSUM NetWare DOS Requester option parameter, explained, 2-70
Client workstation
  configuring, with NET.CFG, 2-3
  improving security for, 1-8
  improving speed of, 1-4, 1-5
  security guidelines for, 1-15
  using NCP packet signature on, for security, 1-11
Command conventions, explained
  DOS, v
  syntax, v
  typographic, v
  UNIX, vii
Command line parameters, using
  explained, 3-3
  syntax conventions, 3-3
  with core NetWare Client software, 3-4
  with DOSNP software, 3-10
Community access management (Desktop SNMP NET.CFG option category)
  example NET.CFG using, 2-21
  parameters, listed and explained 2-18 (see also specific parameter name)
Community names (Desktop SNMP NET.CFG option category)
  example NET.CFG using, 2-21
  parameters, listed and explained 2-15 (see also specific parameter name)
Community types (Desktop SNMP NET.CFG option category)
  example NET.CFG file using, 2-21
  parameters, listed and explained 2-18 (see also specific parameter name)
Configuration
  file, NET.CFG (see NET.CFG file)
  parameters, default NET.CFG (see Default configuration (NET.CFG) settings)
  specifying network board driver, 2-27
Configuring client workstations with NET.CFG, explained, 2-3
CONFIRM CRITICAL ERROR ACTION NetWare DOS Requester option parameter, explained, 2-71
Connection sockets, explained
  raw, 2-123
  Transmission Control Protocol (TCP), 2-121
  User Datagram Protocol (UDP), 2-122
CONNECTIONS NetWare DOS Requester option parameter, explained, 2-72
CONTROL COMMUNITY Desktop SNMP option parameter, explained, 2-18
Conventions, explained
  command line parameter syntax, 3-3
  documentation, used in this manual, v
  NET.CFG file format, vi, 2-4
Core VLM programs in NetWare DOS Requester, listed and explained, 2-52
Creating NET.CFG file 2-4. See also NET.CFG file, modifying

D

Default configuration (NET.CFG) settings
  example file, 2-60
  listed, 2-7
Desktop SNMP NET.CFG option
  explained, 2-13
  parameters available, listed and explained 2-13 (see also specific parameter name)
  syntax example, 2-13
Disabling
  Large Internet Packets (LIP), 1-7
  NCP packet signature, 1-12
  Packet Burst, 1-5
DNS. See Domain Name System

Index

Domain Name System (DNS) UDP socket requirements, 2-122
DOS conventions, explained
  command, v
  file, path, and directory names, v
DOS NAME NetWare DOS Requester option parameter, explained, 2-72
DOS Requester. See NetWare DOS Requester
DOSNP software
  command example, 3-10
  command line options, explained, 3-10
Drivers, specifying configurations for network board, 2-27
Drivers, using
  instead of TOKEN, 2-43
  with IBM LAN Support Programs, 2-43
  with NDIS protocol stacks, 2-44
  with serial hardware, SLIP_PPP, 2-44

E
ENABLE CONTROL COMMUNITY Desktop SNMP option parameter, explained, 2-20
ENABLE MONITOR COMMUNITY Desktop SNMP option parameter, explained, 2-20
ENABLE TRAP COMMUNITY Desktop SNMP option parameter, explained, 2-20
EOJ NetWare DOS Requester option parameter, explained, 2-73
Error messages. See System messages
EXCLUDE VLM NetWare DOS Requester option parameter, explained, 2-74

F
Filename conventions, explained
  DOS, v
  UNIX, vii
FIRST NETWORK DRIVE NetWare DOS Requester option parameter, explained, 2-74
FORCE FIRST NETWORK DRIVE NetWare DOS Requester option parameter, explained, 2-75
FRAME Link driver option parameter, explained, 2-30
Frame size, setting token-ring (with LANSUP), 2-34

G
Groups, Desktop SNMP NET.CFG option MIB-II support (listed) 2-22. See also specific group name

H
HANDLE NET ERRORS NetWare DOS Requester option parameter, explained, 2-75

I
Installing NCP packet signature, explained, 1-11
INT64 Protocol IPX option parameter, explained, 2-104
INT7A Protocol IPX option parameter, explained, 2-105
Interface group, MIB-II support (Desktop SNMP NET.CFG option)
  example NET.CFG using, 2-26
  explained, 2-25
IP_ADDRESS Protocol TCPIP option parameter, explained, 2-118
IP_NETMASK Protocol TCPIP option parameter, explained, 2-119
IP_ROUTER Protocol TCPIP option parameter, explained, 2-120
IPATCH Protocol IPX option parameter, explained, 2-106
IPX | UDP. See Transport IPX | UDP NET.CFG option
IPX PACKET SIZE LIMIT Protocol IPX option parameter, explained, 2-106
IPX RETRY COUNT Protocol IPX option parameter, explained, 2-107
IPX SOCKETS Protocol IPX option parameter, explained, 2-107
IPXODI software
  command example, 3-6
  explained, 3-4
  parameters and values, listed and explained, 3-5
IRQ Link driver option parameter, explained, 2-34

K
Key name conventions, explained, vi

L
LAN Driver NET.CFG parameters, listed and explained, 2-41
Large Internet Packet (LIP) functionality
  disabling, 1-7
  explained, 1-5
  when to use, 1-6
LARGE INTERNET PACKETS NetWare DOS Requester option parameter, explained, 2-76
Link Driver NET.CFG option
  explained, 2-27
  parameters available, listed 2-27 (see also specific parameter name)
  syntax example, 2-27
Link Support Layer (LSL) software
  command example, 3-7
  explained, 3-4
  parameters and values, listed and explained., 3-6
Link Support NET.CFG option
  explained, 2-46
  parameters available, listed 2-46 (see also specific parameter name)
  syntax example, 2-46
LIP START SIZE NetWare DOS Requester option parameter, explained, 2-77
LIP. See Large Internet Packet
LOAD CONN TABLE LOW NetWare DOS Requester option parameter, explained, 2-77
LOAD LOW CONN NetWare DOS Requester option parameter, explained, 2-78
LOAD LOW IPXNCP NetWare DOS Requester option parameter, explained, 2-79
LOCAL PRINTERS NetWare DOS Requester option parameter, explained, 2-80

Index

LOCK DELAY NetWare DOS Requester option parameter, explained, 2-81
LOCK RETRIES NetWare DOS Requester option parameter, explained, 2-81
LONG MACHINE TYPE NetWare DOS Requester option parameter, explained, 2-82
LSL. See Link Support Layer

M
MAX BOARDS Link support option parameter, explained, 2-49
MAX FRAME SIZE Link driver option parameter, explained, 2-34
MAX STACKS Link support option parameter, explained, 2-50
MAX TASKS NetWare DOS Requester option parameter, explained, 2-83
MEM Link driver option parameter, explained, 2-35
Memory requirements, Packet Burst, 1-4
MEMPOOL Link support option parameter, explained, 2-50
MESSAGE LEVEL NetWare DOS Requester option parameter, explained, 2-83
MESSAGE TIMEOUT NetWare DOS Requester option parameter, explained, 2-84
Messages. See System messages
MIB-II support (Desktop SNMP NET.CFG option category)
  example NET.CFG using, 2-26
  explained, 2-22
  interface group, explained, 2-25
  parameters, listed and explained 2-22 (see also specific parameter name)
  SNMP group, explained, 2-22
  system group, explained, 2-22
  TCP/IP groups, explained, 2-26
MINIMUM SPX RETRIES Protocol SPX option parameter, explained, 2-110
MINIMUM TIME TO NET NetWare DOS Requester option parameter, explained, 2-85
Modifying NET.CFG file. See NET.CFG file, modifying
MONITOR COMMUNITY Desktop SNMP option parameter, explained, 2-17

N
NAME CONTEXT NetWare DOS Requester option parameter, explained, 2-85
Name servers supported by TCP/IP Transport for DOS, explained, 2-122
NCP packet signature
  disabling, 1-12
  effective signature levels, explained, 1-10
  examples, 1-10
  explained, 1-8
  improving security with, 1-8
  installing, 1-11
  levels, explained, 1-9
  options, explained, 1-9
  troubleshooting, 1-13
  when to use, 1-9
NET.CFG file
  client workstation, configuring with, 2-3
  conventions, explained, vi, 2-4
  creating, 2-4
  example, 2-5, 2-60
  explained, 2-3
  format of, 2-4
  Large Internet Packets, disabling with, 1-7
  NCP packet signature, disabling with, 1-12
  Packet Burst, disabling with, 1-5
  parameters, entering in, 2-4
  parameters, listed and explained, 2-7
NET.CFG file, modifying
  explained, 2-4
  for Desktop SNMP, 2-13, 2-126
  for IPX protocol, 2-103
  for LAN driver, 2-27
  for Link Driver support, 2-46
  for NetWare client workstation software, 2-52
  for SPX protocol, 2-109
  for TCP/IP protocol stack, 2-114
  for transport providers, 2-126
NET.CFG options listed, with parameters and defaults 2-7. See also specific option name; specific parameter name
NetBIOS UDP socket requirements, 2-122
NetWare DOS Requester NET.CFG option
  explained, 2-52
  parameters available, listed 2-62 (see also specific parameter name)
  syntax example, 2-63
NetWare DOS Requester software
  command (VLM) example, 3-8
  core VLM programs, listed and explained, 2-52
  default settings, example NET.CFG file, 2-60
  explained, 2-52, 3-4
  managing, 2-56
  NETX parameters, compatibility with, 2-54
  optimizing, for best performance, 2-57
  optimizing, for smallest conventional memory usage, 2-59
  parameter and values, listed and explained, 3-5
NetWare Loadable Module (NLM), using third-party with NCP packet signature, 1-14
NetWare Protocol NetWare DOS Requester option parameter, explained, 2-86
Network
  increasing speed on, 1-4
  security (see Security)
NETWORK PRINTERS NetWare DOS Requester option, explained, 2-87
NLM. See NetWare Loadable Module
NO_BOOTP Protocol TCPIP option parameter, explained, 2-124
NODE ADDRESS Link driver option parameter, explained, 2-36, 2-37
Non-Core VLM programs in NetWare DOS Requester, listed and explained, 2-53

O
ODI LAN Driver software

Index

command example, 3-7
explained, 3-4
parameters and values, listed and explained, 3-7
Optimizing NetWare Client software
memory usage, 2-59
security, 1-8, 1-15
speed, 1-4, 2-57
Options. See NET.CFG options

P
Packet
forgery, preventing, 1-8
sending large, over network, 1-5
signature (see NCP packet signature)
Packet Burst
disabling, 1-5
memory requirements, 1-4
using, to increase client speed, 1-4
when to use, 1-5
Parameters. See BIND parameters; Command line parameters; NET.CFG parameters; specific parameter name
PATH TCP_CFG Protocol TCPIP option parameter, explained, 2-125
PB BUFFERS NetWare DOS Requester option parameter, explained, 2-88
PBURST READ WINDOWS SIZE NetWare DOS Requester option parameter, explained, 2-88
PBURST WRITE WINDOWS SIZE NetWare DOS Requester option parameter, explained, 2-89
PORT Link driver option parameter, explained, 2-38
PREFERRED SERVER NetWare DOS Requester option parameter, explained, 2-89
PREFERRED TREE NetWare DOS Requester option parameter, explained, 2-90
PRINT BUFFER SIZE NetWare DOS Requester option parameter, explained, 2-91
PRINT HEADER NetWare DOS Requester option parameter, explained, 2-92
PRINT TAIL NetWare DOS Requester option parameter, explained, 2-92
Problems. See Troubleshooting
Protocol IPX NET.CFG option
explained, 2-103
parameters available, listed 2-103 (see also specific parameter name)
syntax example, 2-103
PROTOCOL Link driver option parameter, explained, 2-39
Protocol SPX NET.CFG option
explained, 2-109
parameters available, listed 2-109 (see also specific parameter name)
syntax example, 2-109
Protocol TCPIP NET.CFG option
explained, 2-114
parameters available, listed 2-114 (see also specific parameter name)
syntax example, 2-115

R
RAW_SOCKETS Protocol TCPIP option parameter, explained, 2-123
READ ONLY COMPATIBILITY NetWare DOS Requester option parameter, explained, 2-93
Requester. See NetWare DOS Requester
Requirements, Packet Burst memory, 1-4
RESPONDER NetWare DOS Requester option parameter, explained, 2-94
ROUTE.COM file. See also IBM Token-Ring Source Routing
reducing traffic on IBM bridges with, 1-10

S
SEARCH MODE NetWare DOS Requester option parameter, explained, 2-94
Security
guidelines for client workstations, 1-15
improving, with NCP packet signature, 1-8
NCP packet signature, risks (explained), 1-9
Server, explained
name, supported by TCP/IP Transport for DOS, 2-122
security, 1-10
signature level, 1-10
SET STATION TIME NetWare DOS Requester option parameter, explained, 2-96
SHORT MACHINE TYPE NetWare DOS Requester option parameter, explained, 2-96
SHOW DOTS NetWare DOS Requester option parameter, explained, 2-97
SIGNATURE LEVEL NetWare DOS Requester option parameter, explained, 2-97
SLOT Link driver option parameter, explained, 2-40
SNMP group, MIB-II support (Desktop SNMP NET.CFG option)
example NET.CFG using, 2-26
explained, 2-22
SNMPENABLEAUTHENTRAP Desktop SNMP option parameter, explained, 2-24
Sockets. See Connection sockets
SPX ABORT TIMEOUT Protocol SPX option parameter, explained, 2-110
SPX CONNECTIONS Protocol SPX option parameter, explained, 2-111
SPX LISTEN TIMEOUT Protocol SPX option parameter, explained, 2-111
SPX VERIFY TIMEOUT Protocol SPX option parameter, explained, 2-112
STPIPX.COM transport provider file, explained, 2-126
STPUDP.COM transport provider file, explained, 2-126
SYSCONTACT Desktop SNMP option parameter, explained, 2-24
SYSLOCATION Desktop SNMP option parameter, explained, 2-25
SYSNAME Desktop SNMP option parameter, explained, 2-25
System group, MIB-II support (Desktop SNMP NET.CFG option)
example NET.CFG using, 2-26
explained, 2-22
System messages, listed and explained, 4-1

Index

T

TCP/IP groups, MIB-II support (Desktop SNMP NET.CFG option)
  example NET.CFG file using, 2-26
  explained, 2-26
TCP/IP Transport for DOS software
  explained, 2-114
  name servers supported by, 2-122
TCP_SOCKETS Protocol TCPIP option parameter, explained, 2-121
TCPIP.EXE program, specifying NET.CFG file location to, 2-114
Third-party NLM problems, troubleshooting, 1-14
Transport Provider IPX | UDP NET.CFG option
  explained, 2-126
  parameters available, listed 2-126 (see also specific parameter name)
  syntax example, 2-126
TRAP COMMUNITY Desktop SNMP option parameter, explained, 2-18
TRAP TARGET Transport Provider IPX | UDP NET.CFG option parameter, explained, 2-127
Troubleshooting. See also Help
  client login, 1-13
  NCP packet signature, 1-13
  security, 1-13
  third-party NLM, 1-14
TRUE COMMIT NetWare DOS Requester option parameter, explained, 2-98

U

UDP_SOCKETS Protocol TCPIP option parameter, explained, 2-122
UNIX conventions, explained
  commands, vii
  file, path, and directory names, vii
USE DEFAULTS NetWare DOS Requester option parameter, explained, 2-99

V

Virtual Loadable Module (VLM). See NetWare DOS Requester software
VLM NetWare DOS Requester option parameter, explained 2-100. See also NetWare DOS Requester software
VLM. See Virtual Loadable Module

W

WORKGROUP NAME NetWare DOS Requester option parameter, explained, 2-91
WORKGROUP NET NetWare DOS Requester option parameter, explained, 2-101

Index