# IxChariot™ Performance Endpoints

IXIA®

Release 6.0

909-0327 Rev. A

March 2005

**Corporate Headquarters**

26601 W. Agoura Rd.
Calabasas, CA 91302
USA
+1 (877) FOR IXIA (877-367-4942)
+1 (818) 871 1800 (International)
(FAX) +1 (818) 871-1805
sales@ixiacom.com

**Web site:** www.ixiacom.com
**General:** info@ixiacom.com
**Investor Relations:** ir@ixiacom.com
**Support:** support@ixiacom.com
**Training:** training@ixiacom.com

**EMEA, India**

Ixia Europe Limited
Avonbury Business Park, Unit 5
Howes Lane
Bicester, Oxfordshire, OX26 2UB
United Kingdom
+44 1869 356370
(FAX) +44 1869 356371
ixiaeurope@ixiacom.com

**Asia Pacific**

Asia Pacific Representative Office
New Shangai International Tower, Suite 26E
360 Pudong Nan Rd
Shanghai 200120
China
+86 21 50543439
ixiachina@ixiacom.com

**Japan**

Ixia KK
Aioi Sampo Shinjuku Building, 16th Floor
3-25-3 Yoyogi Shibuya-Ku
Tokyo 151-0053
+81-3-5365-4690
(FAX) +81-3-3299-6263

Part No. 913-0013 Rev. A
March 14, 2005

# Table of Contents

## Chapter 1 Introduction

Endpoint Requirements and Capabilities . . . . . . . . . . . . . . 1-1

Operating System and Protocol Stack Support . . . . . . . . . . 1-1

Microsoft Windows 3.1 TCP/IP Stacks . . . . . . . . . . . . . . . . . . . . . . 1-4

MVS TCP/IP Stacks. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 1-4

Endpoint Capabilities. . . . . . . . . . . . . . . . . . . . . . . . . . . . 1-4

Performance Endpoint Support for IxChariot Functions . . . 1-6

IPv6 Test Module Support . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 1-7

Endpoint Computer Resource Guidelines . . . . . . . . . . . . . . 1-8

Generating Maximum Throughput. . . . . . . . . . . . . . . . . . . . 1-8

Calculating Memory Requirements . . . . . . . . . . . . . . . . . . . 1-9

Endpoint Pair Capacity . . . . . . . . . . . . . . . . . . . . . . . . . . . 1-10

Endpoint Versions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 1-11

What’s New in Version 6.0? . . . . . . . . . . . . . . . . . . . . . . . 1-11

## Chapter 2 Endpoint Initialization File

ALLOW . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2-2

SECURITY_AUDITING . . . . . . . . . . . . . . . . . . . . . . . . . . 2-3

AUDIT_FILENAME . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2-4

ENABLE_PROTOCOL. . . . . . . . . . . . . . . . . . . . . . . . . . . 2-5

Configuring Endpoints for Large-Scale Customization . . . . 2-5

## Chapter 3 Ixia

Theory of Operation. . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3-3
- Single Network Operation. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3-3
- Two Network Configuration. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3-4
- Setting up Ixia Ports . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3-4

Installing Ixia Endpoints . . . . . . . . . . . . . . . . . . . . . . . . . . 3-6
- IxChariot Chassis Installation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3-6
- IxChariot Management Station Installation . . . . . . . . . . . . . . . . . . . . .3-6
- Using Different PCs for IxApplifier and IxChariot Console Software.3-6

Configuring Ixia Endpoints . . . . . . . . . . . . . . . . . . . . . . . . 3-8
- Planning . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3-8
- Step by Step Usage . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3-9

Logging and Messages . . . . . . . . . . . . . . . . . . . . . . . . . 3-27
- Message CHR0181. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3-27

Starting and Stopping Ixia Endpoints . . . . . . . . . . . . . . . . . 3-27
- Stopping the Endpoint. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3-28
- Restarting the Endpoint . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3-28
- Additional Notes . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3-28

Getting the Latest Fixes and Service Updates. . . . . . . . . . 3-29

## Chapter 4 Distributing Endpoints using SMS

Installing Endpoints Using SMS . . . . . . . . . . . . . . . . . . . . . . 4-1

Uninstalling Endpoints Using SMS . . . . . . . . . . . . . . . . . . . 4-3

## Chapter 5 HP-UX

Installation Requirements for HP-UX Endpoints . . . . . . . . . 5-1

Endpoint Installation for HP-UX. . . . . . . . . . . . . . . . . . . . . . 5-2
Unattended Installation for HP-UX . . . . . . . . . . . . . . . . . . . . . . . . 5-4
What Happens During Installation . . . . . . . . . . . . . . . . . . . . . . . . . 5-4
Removing the Endpoint Package (Uninstall) . . . . . . . . . . . . . . . . . 5-5

Configuring HP-UX Endpoints. . . . . . . . . . . . . . . . . . . . . . . 5-6
Configuration for TCP/IP . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5-6
Determining Your IP Network Address . . . . . . . . . . . . . . . . . . . . . 5-6
Testing the TCP/IP Connection . . . . . . . . . . . . . . . . . . . . . . . . . . . 5-6
Sockets Port Number . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5-7

Running HP-UX Endpoints . . . . . . . . . . . . . . . . . . . . . . . . 5-7
Starting an HP-UX Endpoint . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5-7
Stopping an HP-UX Endpoint . . . . . . . . . . . . . . . . . . . . . . . . . . . 5-8
Cleanup after Unexpected Errors . . . . . . . . . . . . . . . . . . . . . . . . . 5-8
How to Tell If an HP-UX Endpoint Is Active . . . . . . . . . . . . . . . . . 5-8
Disabling Automatic Startup . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5-8
Messages CHR0174, CHR0204, CHR0210, or CHR0245 . . . . . . . 5-8

Logging and Messages . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5-9
CORE and CMA_DUMP.LOG Files . . . . . . . . . . . . . . . . . . . . . . . . 5-9
Message CHR0181 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5-9

Updates for HP-UX . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5-10

## Chapter 6 IBM AIX

Installation Requirements for AIX Endpoints . . . . . . . . . . . . . 6-1

Endpoint Installation for AIX. . . . . . . . . . . . . . . . . . . . . . . . . . 6-1
Unattended Installation for AIX . . . . . . . . . . . . . . . . . . . . . . . . . . . 6-4
What Happens During Installation . . . . . . . . . . . . . . . . . . . . . . . . 6-4
Removing the Endpoint Package (Uninstall). . . . . . . . . . . . . . . . . . . 6-5

Configuring AIX Endpoints . . . . . . . . . . . . . . . . . . . . . . . . . . . 6-5
Configuration for TCP/IP . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6-5
Determining Your IP Network Address . . . . . . . . . . . . . . . . . . . . . 6-5
Testing the TCP/IP Connection. . . . . . . . . . . . . . . . . . . . . . . . . . . 6-6
Sockets Port Number . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6-6

Running AIX Endpoints . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6-6
Starting an AIX Endpoint. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6-6
Stopping an AIX Endpoint . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6-7
Cleanup after Unexpected Errors . . . . . . . . . . . . . . . . . . . . . . . . . 6-7
How to Tell If an AIX Endpoint Is Active . . . . . . . . . . . . . . . . . . . . 6-7
Disabling Automatic Startup . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6-8

Logging and Messages . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6-8
Message CHR0181. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6-8
Updates for AIX. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6-8

## Chapter 7 Linux

Installation Requirements for Linux Endpoints. . . . . . . . . . . . 7-2
RPM-Based Endpoint Installation for Linux . . . . . . . . . . . . . . . . . . 7-3
Removing the RPM Endpoint Package (Uninstall) . . . . . . . . . . . . . 7-4
Cobalt-Based Endpoint Installation for Linux . . . . . . . . . . . . . . . . . 7-4
Removing the Cobalt Endpoint Package (Uninstall) . . . . . . . . . . . . 7-7
TAR-Based Endpoint Installation for Linux . . . . . . . . . . . . . . . . . . . 7-8
Removing the TAR-Based Endpoint Package (Uninstall) . . . . . . . 7-10
Unattended Installation for TAR-Based Linux . . . . . . . . . . . . . . . . 7-10
What Happens During Installation . . . . . . . . . . . . . . . . . . . . . . . 7-10

Configuring Linux Endpoints . . . . . . . . . . . . . . . . . . . . . . . 7-11
Configuration for TCP/IP . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7-11
Determining Your IP Network Address for TAR and RPM Linux . . 7-11
Determining Your IP Network Address for Cobalt . . . . . . . . . . . . . 7-11
Testing the TCP Connection . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7-12
Sockets Port Number . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7-12
Autostarting the Endpoint . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7-12

Running Linux Endpoints . . . . . . . . . . . . . . . . . . . . . . . . 7-13
Starting a Linux Endpoint. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7-13
Stopping a Linux Endpoint. . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7-14
Cleanup after Unexpected Errors . . . . . . . . . . . . . . . . . . . . . . . . 7-14
How to Tell If a Linux Endpoint Is Active . . . . . . . . . . . . . . . . . . . 7-15
Disabling Automatic Startup . . . . . . . . . . . . . . . . . . . . . . . . . . . 7-15

Logging and Messages . . . . . . . . . . . . . . . . . . . . . . . . . 7-15
Message CHR0181 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7-15
Increasing the Number of Concurrent Connections . . . . . . . . . . . 7-16

Updates for Linux . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7-16

## Chapter 8 Linux IA-64

Installing Linux IA-64 Endpoints . . . . . . . . . . . . . . . . . . . . . 8-1
TAR-Based Endpoint Installation for Linux IA-64. . . . . . . . . . . . . . . 8-2
What We Do During Installation. . . . . . . . . . . . . . . . . . . . . . . . . . 8-3
Removing the TAR-Based Endpoint Package (Uninstall) . . . . . . . . 8-4
Unattended Installation for TAR-Based Linux IA-64 . . . . . . . . . . . . 8-4

Configuring Linux IA-64 Endpoints . . . . . . . . . . . . . . . . . . . 8-5
Configuration for TCP/IP . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8-5
Determining Your IP Network Address . . . . . . . . . . . . . . . . . . . . . 8-5
Sockets Port Number . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8-5
Testing the TCP Connection . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8-5

Running Linux IA-64 Endpoints. . . . . . . . . . . . . . . . . . . . . . 8-6
Autostarting the Endpoint . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8-6
Starting a Linux IA-64 Endpoint . . . . . . . . . . . . . . . . . . . . . . . . . . 8-6

Stopping a Linux IA-64 Endpoint. . . . . . . . . . . . . . . . . . . . . . . . . . 8-7
Cleanup after Unexpected Errors . . . . . . . . . . . . . . . . . . . . . . . . . 8-7
How to Tell If an IA-64 Linux Endpoint Is Active . . . . . . . . . . . . . . . 8-7
Disabling Automatic Startup . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8-7
Increasing the Number of Concurrent Connections . . . . . . . . . . . . 8-8

Logging and Messages . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8-8
Message CHR0181. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8-8

Getting the Latest Fixes and Service Updates. . . . . . . . . . . . 8-8
Updates for Linux IA-64 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8-8

## Chapter 9 Mac OS X

Installation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9-1
Installation Procedure . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .9-1
Removing the Endpoint (Uninstall) . . . . . . . . . . . . . . . . . . . . . . . . 9-2
What Happens During Installation . . . . . . . . . . . . . . . . . . . . . . . . 9-2

Configuring Mac OS X Endpoints . . . . . . . . . . . . . . . . . . . . 9-2
Configuration for TCP/IP. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9-2
Determining Your IP Network Address . . . . . . . . . . . . . . . . . . . . . 9-3
Testing the TCP Connection . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9-3
Sockets Port Number . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9-3

Running Mac OS X Endpoints . . . . . . . . . . . . . . . . . . . . . . . 9-3
Stopping a Mac OS X Endpoint . . . . . . . . . . . . . . . . . . . . . . . . . . 9-4
Cleanup after Unexpected Errors . . . . . . . . . . . . . . . . . . . . . . . . . 9-4
How to Tell If a Mac OS X Endpoint Is Active . . . . . . . . . . . . . . . . 9-4
Disabling Automatic Startup . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9-5

Logging and Messages . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9-5

Updates for Mac OS X. . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9-5

## Chapter 10 Microsoft Windows CE

Installation Requirements . . . . . . . . . . . . . . . . . . . . . . . . . 10-1

Network Protocol Stacks . . . . . . . . . . . . . . . . . . . . . . . . . . 10-1

Endpoint Installation for Windows CE . . . . . . . . . . . . . . . . . 10-2

Installation on Intel Strong Arm and XScale Processors . . . . . . . . 10-2

Installation on Intel x86 Processors . . . . . . . . . . . . . . . . . . . . . . . 10-2

Removing the Endpoint Package (Uninstall) . . . . . . . . . . . 10-3

Windows CE Configuration for TCP/IP . . . . . . . . . . . . . . . 10-3

Determining Your IP Network Address . . . . . . . . . . . . . . . . . . . . 10-3

Testing the TCP Connection . . . . . . . . . . . . . . . . . . . . . . . . . . . 10-3

Sockets Port Number . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 10-4

Running Windows CE Endpoints . . . . . . . . . . . . . . . . . . . 10-4

Limitations of the Windows CE Endpoint . . . . . . . . . . . . . . . . . . . 10-4

Intel Strong Arm and XScale Processor Based Operation . . . . . . 10-5

Intel x86 Processor Based Operation . . . . . . . . . . . . . . . . . . . . . 10-5

Checking the Endpoint Version . . . . . . . . . . . . . . . . . . . . . . . . . 10-5

Error Messages . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 10-5

## Chapter 11 Microsoft Windows NT/2000/2003/XP

Installation Requirements for Windows Endpoints . . . . . . 11-2

Endpoint Installation . . . . . . . . . . . . . . . . . . . . . . . . . . . . 11-3

What Happens During Installation . . . . . . . . . . . . . . . . . . . . . . . 11-7

Unattended Installation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 11-7

Installing the Windows Endpoint with SMS . . . . . . . . . . . . . . . . . 11-8

Removing the Endpoint Package (Uninstall) . . . . . . . . . . . . . . . . 11-8

Removing the Endpoint Manually . . . . . . . . . . . . . . . . . . . . . . . 11-9

Configuring Windows Endpoints . . . . . . . . . . . . . . . . . . . . 11-9

Windows NT and Windows 2000 Configuration for APPC . . . . . . 11-9

Windows NT, Windows 2000, or Windows XP Configuration for IPX and SPX . . . . . 11-10
Windows NT, Windows 2000, or Windows XP Configuration for TCP/IP . . . . . 11-12

Running Windows Endpoints . . . . . 11-13
Starting the Endpoint. . . . . 11-13
Stopping a Windows Endpoint . . . . . 11-13
Disable Your Screen Saver . . . . . 11-14
Disabling Automatic Startup . . . . . 11-15
How to Tell If a Windows Endpoint Is Active . . . . . 11-16

Logging and Messages . . . . . 11-16
Application Monitoring Support with CheckPoint VPN Software. 11-16

Getting the Latest Fixes and Service Updates. . . . . 11-17
Updates and Information for Windows . . . . . 11-17
Updates for Microsoft SNA Server . . . . . 11-17
Updates for IBM SNA Software for Windows . . . . . 11-17

Chapter 12 Microsoft Windows XP/2003 64-bit Edition

Installation Requirements for Windows XP/2003 64-Bit Edition Endpoints. . . . . 12-1

Endpoint Installation for Windows XP/2003 64-bit Edition . 12-2
Unattended Installation for 64-bit Windows XP/2003 . . . . . 12-4
What We Do During Installation . . . . . 12-5
Removing the Endpoint Package (Uninstall). . . . . 12-5

Configuring 64-bit Windows XP/2003 Endpoints . . . . . 12-5
Windows XP/2003 Configuration for TCP/IP . . . . . 12-6
Determining Your IP Network Address . . . . . 12-6
Trying Out the TCP/IP Connection . . . . . 12-7
Sockets Port Number . . . . . 12-7

Running 64-bit Windows XP/2003 Endpoints . . . . . 12-7
Starting a Windows XP/2003 Endpoint . . . . . 12-7

Stopping a Windows XP/2003 Endpoint . . . . . . . . . . . . . . . . . . . . 12-8
Disable Your Screen Saver . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 12-8
Disable NIC Power Save Mode . . . . . . . . . . . . . . . . . . . . . . . . . . . 12-8
Disabling Automatic Startup . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 12-8
How to Tell If a Windows XP/2003 Endpoint Is Active . . . . . . . . . . 12-8
The SetAddr Utility for 64-bit Windows . . . . . . . . . . . . . . . . . . . . . 12-8

Logging and Messages . . . . . . . . . . . . . . . . . . . . . . . . . . 12-10

Getting the Latest Fixes and Service Updates . . . . . . . . 12-10
Updates and Information for Windows XP/2003 . . . . . . . . . . . . . 12-10

## Chapter 13 Sun Solaris

Installation Requirements for Solaris Endpoints . . . . . . . . 13-1

Endpoint Installation for Sun Solaris . . . . . . . . . . . . . . . . . 13-2
Installation Defaults File for Solaris . . . . . . . . . . . . . . . . . . . . . . . 13-4
Unattended Installation for Solaris. . . . . . . . . . . . . . . . . . . . . . . . . 13-5
What Happens During Installation . . . . . . . . . . . . . . . . . . . . . . . . 13-5
Removing the Endpoint Package (Uninstall) . . . . . . . . . . . . . . . . 13-6

Configuring Solaris Endpoints . . . . . . . . . . . . . . . . . . . . . . 13-6
Configuration for TCP/IP . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 13-8
Determining Your IP Network Address . . . . . . . . . . . . . . . . . . . . 13-8
Testing the TCP/IP Connection . . . . . . . . . . . . . . . . . . . . . . . . . . 13-8
Sockets Port Number . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 13-8

Running Solaris Endpoints . . . . . . . . . . . . . . . . . . . . . . . . 13-9
Starting a Solaris Endpoint . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 13-9
Stopping a Solaris Endpoint . . . . . . . . . . . . . . . . . . . . . . . . . . . . 13-9
Cleanup after Unexpected Errors . . . . . . . . . . . . . . . . . . . . . . . 13-10
How to Tell If a Solaris Endpoint Is Active . . . . . . . . . . . . . . . . . 13-10
Disabling Automatic Startup . . . . . . . . . . . . . . . . . . . . . . . . . . . 13-10

Logging and Messages . . . . . . . . . . . . . . . . . . . . . . . . . . 13-10
Known Problems . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 13-10
Message CHR0181 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 13-11

Updates for Sun Solaris . . . . . . . . . . . . . . . . . . . . . . . . . . 13-11

## Chapter 14 Web-Based Performance Endpoint

Running the Web-Based Endpoint . . . . . . . . . . . . . . . . . . . . 14-2

Error Handling . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 14-2

Compatibility with Other Endpoints . . . . . . . . . . . . . . . . . . . 14-3

Stopping the Web-Based Endpoint . . . . . . . . . . . . . . . . . . . 14-3

# List of Tables

Table 1-1. Endpoint - Operating System Compatibility . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .1-2
Table 1-2. Endpoint Compatibility . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .1-4
Table 1-3. Performance Endpoint Capabilities per OS . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .1-6
Table 1-4. Guidelines for Selecting CPUs . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .1-8
Table 1-5. Calculating Memory Requirements . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .1-9
Table 1-6. Endpoint Pair Capacity . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .1-10
Table 2-1. Endpoint Initialization File Defaults . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .2-1
Table 2-2. Security Auditing . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .2-3
Table 2-3. Audit File Contents . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .2-4
Table 3-1. IxApplifier Menus . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3-10
Table 3-2. Chassis Chain Column Interpretation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3-12
Table 3-3. Chassis Properties . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3-13
Table 3-4. Add Interfaces Dialog Fields . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3-17
Table 3-5. Filters Tab Fields . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3-22
Table 3-6. Network Settings . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3-24
Table 3-7. Misc dialog choices for LM1000STXS4 ports . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3-26
Table 3-8. Ixia port - reserved port usage. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3-28
Table 4-1. SETUP Command Parameters . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .4-2
Table 14-1. Supported/Unsupported Features . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .14-1

# 1 Introduction

This guide contains information about the Performance Endpoints, which are available for more than 20 different operating systems.

All the information you need to install, configure, and run the endpoints in your network is included here and in the printed version of the *Performance Endpoints* guide. In addition to topics discussing issues common to all the endpoints, these guides also contain information about each operating system, organized in separate chapters.

## Endpoint Requirements and Capabilities

The following topics describe the software and hardware requirements and the supported functions of the Performance Endpoints, version 5.0 through 5.20.

The latest version of the endpoint software can always be downloaded free from the Internet. A single installable file is available for each operating system. Endpoints are available for downloading at http://www.ixiacom.com/support/ixchariot.

You cannot run endpoint software from a CD-ROM; you must install it on a computer.

## Operating System and Protocol Stack Support

**Related Topics**

*Endpoint Capabilities* on page 1-4

The following tables list the software with which we have tested the Performance Endpoints for each operating system.

**Note**: Versions listed are the **earliest**, not necessarily the only, versions supported.

Table 1-1.Endpoint - Operating System Compatibility

| Endpoint | OS version | TCP, UDP, RTP | IP Multicast version | IPX/SPS stack | APPC stack version |
|---|---|---|---|---|---|
| Cobalt RaQ/RaQ2 (MIPS) | Linux v. 2.0 for MIPS | included | kernel 2.0.32 | no | no |
| Cobalt RaQ3 (x86) | kernel 2.0.32 | included | kernel 2.0.32 | no | no |
| Compaq Tru64 UNIX | Digital UNIX 4.0B or Compaq Tru64 Unix for Alpha | included | v4.0B | no | no |
| FreeBSD UNIX | BSD v3.1 | included | v3.1 | no | no |
| HP-UX | HP-UX v10.10 | included | v10.10 | no | no |
| IBM AIX | AIX v4.1.4 | included | v4.1.4 | no | no |
| IBM MVS | MVS/ESA SP v4R2.2 | See "*MVS TCP/IP Stacks* on page 1-4" | no | no | IBM ACF/ VTAM for MVS/ESA v3R4.2 |
| IBM OS/2 | OS/2 Warp 4, Warp Connect 3 | Download TCP 4.1 | Download TCP 4.1 | Download Novell Netware Client v2.12 | IBM CommServer for OS/2 v4.1 |
| Ixia | Linux - automatically downloaded | included | included | included | included |
| Linux (x86 and MIPS) | kernel 2.0.32 | included | kernel 2.0.32 | no | no |
| Linux IA-64 | kernel 2.4.0test7-42 | included | kernel 2.4.0test7-42 | no | no |
| Microsoft Windows 3.1 | Windows 3.1 or Windows for Workgroups 3.11 | see "*Microsoft Windows 3.1 TCP/IP Stacks* on page 1-4" | Chameleon 7.0, as E2 | no | no |
| Microsoft Windows 95 | Windows 95 | included | no | Download Novell Netware Client v3.21 | IBM PComm v4.3 for Windows 95 |
| Microsoft Windows 95 with WinSock 2 | Windows 95 with WinSock 2 installed | Download WinSock 2 | included | included | IBM PComm v4.3 for Windows 95 |

Table 1-1.Endpoint - Operating System Compatibility (Continued)

| **Endpoint** | **OS version** | **TCP, UDP, RTP** | **IP Multicast version** | **IPX/SPS stack** | **APPC stack version** |
|---|---|---|---|---|---|
| Microsoft Windows 98 | Windows 98 | included | included | included | IBM PComm v4.3 for Windows 98 |
| Microsoft Windows Millennium Edition (Me) | Windows Me | included | included | included | IBM PComm v4.3 for Windows 98 |
| Microsoft Windows NT 4 | Windows NT SP 4 | included | SP3 (IGMPv1)<br>SP4 (IGMPv2) | included | IBM PComm v4.3, or IBM CommServer v5.0 (for Windows NT), or Microsoft SNA Server v4.0s for Windows NT |
| Microsoft Windows NT 4 for Alpha | Windows NT4 SP 3 | included | SP3 (IGMPv1)<br>SP4 (IGMPv2) | included | Microsoft SNA Server for Alpha v4.0 with SP1 or v3.0 with SP2 |
| Microsoft Windows 2000 | Windows 2000 | included | included | included | IBM PCOMM version 5.0, or IBM CommServer v6.0 |
| Microsoft Windows XP | Windows XP (32-bit) | included | included | included | IBM PCOMM version 5.0, or IBM CommServer v6.0 |
| Microsoft Windows XP (64-bit) | Windows XP (64-bit) | included | included | no | no |
| Novell NetWare | v3.12 | included | v4.0 | included | no |
| SCO UnixWare | UnixWare v2.1 | included | v7.0 | no | no |
| SGI IRIX | IRIX v6.2 with patches | included | v6.2 | no | no |
| Sun Solaris for SPARC | Solaris v2.4 | included | v2.4 | no | no |
| Sun Solaris for x86 | Solaris v2.4 | included | v2.4 | no | no |

## Microsoft Windows 3.1 TCP/IP Stacks

The Microsoft Windows 3.1 Performance Endpoint software supports the following TCP/IP stacks:

- Microsoft 32-bit stack, shipped on the Windows NT 4.0 Server CD-ROM
- Frontier Technologies *SuperTCP* v2.2
- FTP Software *OnNet for Windows* v2.1
- NetManage *Chameleon NFS* v4.6.3 (IP Multicast support requires version 7.0 or later)
- Novell Client 3.1 for DOS and Windows 3.x v2.71
- Novell *Client for DOS/Win* (VLMs) v1.21
- WRQ TCP Connection for Windows v5.1

Because Windows 3.x lacks thread support, you cannot use the Windows 3.1 endpoint as Endpoint 1 in an IP Multicast test.

## MVS TCP/IP Stacks

The MVS Performance Endpoint software supports the following TCP/IP stacks:

- TCP/IP versions 3.2 through 3.8, from IBM. Version 2.6 of OS/390 (TCP/IP version 3.5) and higher includes support for IP Multicast testing with IxChariot.
- *SOLVE:TCPaccess* versions 4.1 and 5.2 stack from Sterling Software. A set of PTFs is required for operation with version 4.1.

# Endpoint Capabilities

### Related Topics

*Performance Endpoint Support for IxChariot Functions* on page 1-6
*Operating System and Protocol Stack Support* on page 1-1

The following table indicates which endpoints have been tested with and are supported by Ixia products. Shaded rows indicate endpoints that have been archived at previous versions. For more details on specific product capabilities, see the topics below.

Table 1-2.Endpoint Compatibility

| **Ixia Product** | **Qcheck** | **IxChariot** |
|---|---|---|
| **Endpoint** | | |
| Compaq Tru64 UNIX | Yes | Yes |
| FreeBSD UNIX | Yes | Yes |
| HP-UX | Yes | Yes |
| IBM AIX | Yes | Yes |
| IBM MVS, Windows install | Yes | Yes |
| IBM OS/2 | Yes | Yes |

Table 1-2.Endpoint Compatibility (Continued)

| **Ixia Product** | **Qcheck** | **IxChariot** |
|---|---|---|
| **Endpoint** | | |
| Ixia | Yes | Yes |
| Linux for Cobalt RaQ/RaQ2 (MIPS) | Yes | Yes |
| Linux for Cobalt RaQ3 (x86) | Yes | Yes |
| Linux x86 (TAR) | Yes | Yes |
| Linux x86 (RPM) | Yes | Yes |
| Linux IA-64 (TurboLinux) | Yes | Yes |
| Microsoft Windows 95 | Yes | Yes |
| Microsoft Windows 98 | Yes | Yes |
| Microsoft Windows NT/2000/ XP | Yes | Yes |
| Microsoft Windows 98 (Web-Based) | Yes | Yes |
| Microsoft Windows NT/2000/ XP (Web-Based) | Yes | Yes |
| Microsoft Windows XP (64-Bit) | Yes | Yes |
| Microsoft Windows 3.1 | Yes | Yes |
| Novell NetWare | Yes | Yes |
| SCO UnixWare | Yes | Yes |
| SGI IRIX | Yes | Yes |
| Sun Solaris (SPARC) | Yes | Yes |
| Sun Solaris Endpoint (x86) | Yes | Yes |

Endpoints for Windows 2000 and Windows XP also support testing with IPv6. Refer to the following topic, *Performance Endpoint Support for IxChariot Functions* on page 1-6 for more information.

## Performance Endpoint Support for IxChariot Functions

The following table describes the Performance Endpoint capabilities for the supported operating systems. Shaded rows indicate endpoints that have been archived at previous versions. These endpoints may not support functionality new in the latest versions of IxChariot.

Table 1-3.Performance Endpoint Capabilities per OS

| Endpoint OS | IP QoS (DiffServ, GQOS, TOS) | Trace-route | CPU Utilitization | VoIP Tests | IPv6 Tests | 802.11 |
|---|---|---|---|---|---|---|
| Cobalt RaQ or RaQ2 (MIPS) | TOS | No | Yes | No | No | No |
| Cobalt RaQ3 (x86) | TOS | Yes | Yes | Yes | No | No |
| Compaq Tru64 UNIX | TOS | No | Yes | No | No | No |
| FreeBSD UNIX | TOS | No | Yes | No | No | No |
| HP-UX | TOS | Yes | Yes | No | No | No |
| IBM AIX | TOS | Yes | Yes | No | No | No |
| IBM MVS | No | No | No | No | No | No |
| IBM OS/2 | TOS | No | Yes | No | No | No |
| Ixia | TOS, DiffServ | Yes | Yes | Yes | Yes | No |
| Linux | TOS | Yes | Yes | Yes | Yes. See “*IPv6 Test Module Support* on page 1-7” | No |
| Linux IA-64 | TOS | Yes | Yes | No | No | No |
| Microsoft Windows 3.1 | No | No | No | No | No | No |
| Microsoft Windows 95 | No | No | Yes | No | No | No |
| Microsoft Windows 95 with WinSock 2 | TOS (UDP, RTP) | Yes | Yes | No | No | No |
| Microsoft Windows 98 | GQOS (RSVP), TOS (UDP, RTP) | Yes | Yes | Yes | No | No |
| Microsoft Windows Me | GQOS (RSVP) | Yes | Yes | Yes | No | No |
| Microsoft Windows NT 4 | TOS (UDP, RTP) | Yes | Yes | Yes | No | No |

Table 1-3.Performance Endpoint Capabilities per OS (Continued)

| **Endpoint OS** | **IP QoS (DiffServ, GQOS, TOS)** | **Trace-route** | **CPU Utilitization** | **VoIP Tests** | **IPv6 Tests** | **802.11** |
|---|---|---|---|---|---|---|
| Microsoft Windows NT 4 for Alpha | No | Yes | Yes | No | No | No |
| Microsoft Windows 2000 | DiffServ, GQOS, TOS (via Registry) | Yes | Yes | Yes | Yes. See "*IPv6 Test Module Support* on page 1-7" | Yes |
| Microsoft Windows 2003 | DiffServ, GQOS, TOS (via Registry) | Yes | Yes | Yes | No. | Yes |
| Microsoft Windows CE 4.20 | DiffServ, GQOS, TOS | Yes | No | Yes | No. | Yes (version 4.20 or later) |
| Microsoft Windows 98 (Web-Based) | Yes | No | Yes | Yes | No | No |
| Microsoft Windows NT/2000/ XP (Web-Based) | Yes | No | Yes | Yes | No | No |
| Microsoft Windows XP | DiffServ, GQOS, TOS (via Registry) | No | Yes | Yes | Yes. See "*IPv6 Test Module Support* on page 1-7" | Yes |
| Microsoft Windows XP (64-bit) | DiffServ, GQoS, TOS | No | No | Yes | No | No |
| Novell NetWare | No | No | No, v3.12;<br>Yes, v4.0 | No | No | No |
| SCO UnixWare | TOS (bits 3-5) | No | No | No | No | No |
| SGI IRIX | TOS | No | Yes | No | No | No |
| Sun Solaris for SPARC | TOS | Yes | Yes | Yes | No | No |
| Sun Solaris for x86 | TOS | Yes | Yes | Yes | No | No |

## IPv6 Test Module Support

Currently, testing with version 6 of the Internet Protocol (IPv6) is only supported on endpoints for Ixia Performance Endpoints, Windows 2003, Windows XP (32-bit only), Red Hat Linux, versions 8.0 or higher. You must first install IPv6 support on these endpoints before you begin testing.

In addition, Windows 2000 provides unofficial support for IPv6, but it requires a patch called the "Microsoft IPv6 Technology Preview for Windows 2000 Network Protocol Stack," which you can download from http://msdn.microsoft.com/Downloads/sdks/platform/tpipv6/readme.asp.

## Endpoint Computer Resource Guidelines

### Related Topics

*Generating Maximum Throughput* on page 1-8
*Calculating Memory Requirements* on page 1-9
*Endpoint Pair Capacity* on page 1-10

Determining the computer requirements for a given endpoint can be challenging. There are many variables involved, such as processor speed, operating system, protocol stack, memory, disk space, and the underlying network.

To determine your computer requirements, you must first define how you plan to use IxChariot. The type of information you need depends upon your usage. The following topics provide recommended endpoint computer specifications according to different testing scenarios.

## Generating Maximum Throughput

The main factors in getting the most throughput from a computer are CPU speed and memory. You need a CPU that is fast enough to match your network capacity, and with enough memory to hold the code and data used for the test. For best throughput, we recommend using a 32-bit (or better) operating system. The memory you need is based on your operating system. Make sure that you have enough memory at the endpoints so that no swapping takes place while running a test. The following table shows some guidelines in determining the best CPU for different network speeds.

Table 1-4. Guidelines for Selecting CPUs

| Throughput | Recommended computer |
|---|---|
| less than 100 Mbps | PCI-based computer with a 32-bit operating system |
| 100 to 200 Mbps | Pentium 166 or greater (consider multiple concurrent pairs) |
| 200 to 500 Mbps | Pentium II or greater (consider multiprocessors) |
| over 500 Mbps | Pentium III or greater, with the latest NICs (consider multiprocessors) |

The following observations may help guide your throughput testing.

- Windows NT, Windows 2000/2003, Windows XP, and Linux yield the highest throughput. If you test on one of the Windows OSs with the IxChariot benchmark script called `High_Performance_Throughput`, the endpoints can make use of Microsoft's WinSock 2 overlapped I/O to achieve much greater throughput on high-speed networks (100 MB and faster). In a test of Gigabit Ethernet throughput using Windows 2000 Server and two Pentium III computers, each having two 933-MHz processors, 1 Gigabyte of RAM, and a single Gigabit NIC, we generated 943 Mbps with six pairs.

- We have also observed some improvements in throughput measurements after changing the `TcpWindowSize` setting in the Windows NT Registry to `65536`. You can set this parameter in the following Registry key:

```
HKEY_LOCAL_MACHINES\SYSTEM\CurrentControlSet\Services\T
cpip
\Parameters
```

Set the key as a type `REG_DWORD`.

Refer to the Windows NT Resource Kit for more information.

## Calculating Memory Requirements

Endpoints are designed to run in any computer that has sufficient memory to run the operating system well. If you plan to use multiple pairs on a single computer, you may want to calculate the number of pairs that will run without causing the operating system to swap either code or data.

The following table can be used to plan for multiple pairs. The Base RAM column indicates the amount of memory that is allocated by the endpoint before running any pairs. If the endpoint is not being used, this amount may go toward zero if the operating system supports swapping. The protocol columns indicate the amount of memory required for a pair of that protocol.

Table 1-5.Calculating Memory Requirements

| Operating System | Base RAM (in KB) | TCP KB/ pair | RTP or UDP KB/ pair | SPX KB/ pair | IPX KB/ pair | APPC KB/ pair |
|---|---|---|---|---|---|---|
| MVS | 666 | 25-48 | 24-52 | n/a | n/a | n/a |
| NetWare | 1100 | 80-110 | 320-340 | 70-100 | 260-280 | n/a |
| OS/2 | 1096 | 50-65 | 150-170 | 315-340 | 150-170 | 65-90 |
| UNIX (AIX) | 1176 | 132-284 | 146-296 | n/a | n/a | n/a |
| Windows NT/2000/XP | 2076 | 35-60 | 160-180 | 35-60 | 160-180 | n/a |

These RAM usage numbers represent sending with the variable `send_datatype` set to `ZEROS`. Other `send_datatypes` require memory buffers roughly equivalent to the disk space of the `.cmp` file being used. Add 2 KBytes when using `send_datatype = NOCOMPRESS`. See the *Application Scripts* guide for more information on script variables.

# Endpoint Pair Capacity

The following table shows some example pair capacities we have tested on various computers. These pairs ran on a 10 Mbps Ethernet LAN. The values in the pairs columns represent the number of pairs this computer supported as Endpoint 2 for a single test. We used the default values for all tests, with two exceptions: for datagram testing, we lengthened the timeout values, as well as the `initial_delay` in test scripts.

This table does not represent the full capacities of these operating systems and stacks, just some representative tests we have run in our test lab.

Table 1-6.Endpoint Pair Capacity

| Operating System | Installed RAM | TCP pairs | RTP or UDP pairs | SPX pairs | IPX pairs | APPC pairs |
|---|---|---|---|---|---|---|
| Ixia - ALM1000T8 | 512MB | 500 | 500 | n/a | n/a | n/a |
| Ixia - TXS family | 256MB | 500 | 500 | n/a | n/a | n/a |
| Ixia - LM100TXS8 | 128MB | 150 | 150 | n/a | n/a | n/a |
| AIX 4.1 | 64 MB | 200 | 180 | n/a | n/a | n/a |
| NetWare 4.12 | 64 MB | 500 | 200 | 100 | 100 | n/a |
| OS/2 4.0 | 32 MB | 500 | 200 | 20 | 20 | 500 |
| Windows NT/2000/XP | 32 MB | 500 | 100 | 300 | 100 | 200 |
| Windows XP | 768 MB | 500 | 100 | 300 | 100 | n/a |
| MAC OSX 10.3 | 512 MB | 50 | 50 | n/a | n/a | n/a |
| Windows CE | 64 MB | 20 | 20 | n/a | n/a | n/a |
| Linux | 768 MB | 500 | 200 | n/a | n/a | n/a |
| Solaris X86 | 768 MB | 500 | 100 | n/a | n/a | n/a |
| Solaris | 768 MB | 500 | 50 | n/a | n/a | n/a |
| HP-UX | 768 MB | 50 | 40 | n/a | n/a | n/a |

**Notes**:

- Ixia - TXS family includes the following load modules: LM1000TXS4, LM1000STXS2, LM1000STXS4, OLM1000STXS24 and LM1000SFPS4.
- On Windows NT and Windows 2000, APPC pairs were run using Microsoft SNA Server.
- On Windows 95, Windows 98, and Windows Me, SPX and IPX pairs were run using Novell Client32 for SPX and IPX.
- On OS/2 4.0, IPX and SPX pairs were run using Novell Client for OS/2.

IxChariot now supports larger tests under certain conditions using TCP. See “Internet-Scale Testing” in the *User Guide* for IxChariot for more information about requirements for large tests.

# Endpoint Versions

With each new release of IxChariot, the endpoints are updated to support new functionality. However, because some endpoint operating systems are rarely used or provide limited support for IxChariot features, endpoints for a few operating systems have been archived. These endpoints are still made available on the Performance Endpoints CD-ROM and on the Ixia Web site; however, they may not support the latest capabilities of IxChariot. The Endpoint README file, included in the root directory of the endpoint CD-ROM, provides a list of all available endpoints and indicates their versions if they are different from the current endpoint level.

# What's New in Version 6.0?

The following endpoints or features are new to this release:

- **Ixia endpoint reserved ports –** the ports reserved by the Ixia Endpoint are listed. Refer to the following for details:
  - *Reserved TCP/UDP Ports* on page 3-28
- **New MAC OS X endpoint.** Refer to the following for details:
  - *Mac OS X* on page 9-1
- **New Windows XP/2003 64-bit Edition endpoint.** Refer to the following for details:
  - *Microsoft Windows XP/2003 64-bit Edition* on page 12-1

## Internal Items

The following items will not be visible to the user. They are for our internal use.

- **Chassis warnings about Tcl Server and use of more than two chassis**
  - *Ixia* on page 3-1
- **New endpoint.log file location for Linux endpoints**
  - *Logging and Messages* on page 3-27
  - *Logging and Messages* on page 7-15
  - *Logging and Messages* on page 8-8

# 2 Endpoint Initialization File

**Related Topics**

*SECURITY_AUDITING* on page 2-3
*ALLOW* on page 2-2
*ENABLE_PROTOCOL* on page 2-5
*AUDIT_FILENAME* on page 2-4
*Configuring Endpoints for Large-Scale Customization* on page 2-5

An endpoint initialization file is installed with each Performance Endpoint. With this file, you can do the following:

- Restrict the use of this endpoint to specific IxChariot or Qcheck Consoles.
- Control which access attempts are logged in an audit file.
- Change the filename of the audit file.
- Enable only particular protocols on this endpoint for setup connections.
- Change the location of the endpoint software used for automatic updating.

On most operating systems, this file is named `endpoint.ini` (on MVS, see data set `HLQ.SLQ.JCL(ENDPTINI)`, where “`HLQ`” and “`SLQ`” are the high-level and second-level qualifiers entered during MVS endpoint installation). This file has the same format and structure on all the operating systems.

Here are the default contents of the endpoint initialization file. You can change these keywords and their parameters to tailor individual endpoints for your needs.

Table 2-1. Endpoint Initialization File Defaults

| Keyword | Parameters |
|---|---|
| ALLOW | ALL |
| SECURITY_AUDITING | NONE |
| AUDIT_FILENAME | ENDPOINT.AUD |
| ENABLE_PROTOCOL | ALL |

Table 2-1. Endpoint Initialization File Defaults (Continued)

| Keyword | Parameters |
|---|---|
| SAFESTORE_DIRECTORY | (the directory where the endpoint is installed) |
| UPDATE_SERVER | endpointupdate.ganymede.com |
| END2END_SERVER | (no default) |

**Note**: For the MVS endpoint, the default filename of ENDPOINT.AUD is ENDPTAUD.

This file is an editable text file. There is a separate copy for each operating system. You might want to make changes to it once, before endpoint installation, which are then incorporated into all the installs for different sets of computers. You can modify this text file before installation by copying the endpoint installation directory for an operating system to a hard drive (preferably a LAN drive), and then modifying the file before running the install from that drive.

We strongly recommend that you make any changes to your endpoint.ini files once, before you install any endpoints, as opposed to installing the endpoints and then going back to each of them and separately modifying each one. If you're using Windows (32-bit or 64-bit) endpoints, we've included a utility to help you edit the endpoint.ini files before installing the endpoints, should you wish to prepare the endpoints for future automatic upgrades. See *Configuring Endpoints for Large-Scale Customization* on page 2-5 for more information.

# ALLOW

### Related Topics

*ENABLE_PROTOCOL* on page 2-5
*Endpoint Initialization File* on page 2-1

This keyword determines which IxChariot or Qcheck Consoles can run tests using this endpoint.

To allow any user to run tests on this endpoint, use the ALL parameter, which is the installation default:

```
ALLOW ALL
```

However, **the default "ALLOW ALL" is NOT RECOMMENDED**. Although "ALLOW ALL" makes it easy to install an endpoint and see that it's running, it also lets any user who can reach the endpoint potentially use that endpoint as a traffic generator.

To allow only specific users to run tests with this endpoint, remove the "ALLOW ALL" line and identify one or more specific IxChariot or Qcheck Consoles by their network addresses. You can specify more than one address per protocol. For example,

```
ALLOW TCP 192.86.77.120
ALLOW TCP 192.86.77.121
ALLOW APPC ixia.johnq
```

Specify a connection-oriented protocol (that is, APPC, TCP, or SPX) as the first parameter and provide its corresponding network address as the second parameter. Endpoints only listen for incoming tests on connection-oriented protocols, like TCP. Datagram tests are set up and results are returned using their "sister" connection-oriented protocol; thus, UDP tests are set up using TCP, and IPX tests are set up using SPX.

The network address cannot be an alias or hostname; that is, in APPC it must be a fully qualified LU name, in TCP/IP it must be an IP address in dotted notation, and in IPX/SPX it must be an IPX address with hex network address and node address.

You cannot use the `ALLOW` parameter to restrict access from one endpoint to another endpoint. The `ALLOW` parameter can only be used to permit (or prevent) access from specific IxChariot or Qcheck Consoles to the endpoint at which the parameter is defined.

If, for some reason, you need to restrict your endpoint to access only your own computer, specify your own IP network address rather than `127.0.0.1`. Specifying `127.0.0.1` (the equivalent of `localhost`) allows any other user who specifies "`localhost`" as Endpoint 1 to access your computer as Endpoint 2.

# SECURITY_AUDITING

This keyword determines which access attempts the endpoint keeps track of in its audit file. Here are the possible parameters:

Table 2-2. Security Auditing

| Parameter | Comment |
|---|---|
| NONE | Nothing is written to the audit file. |
| PASSED | Only access attempts that passed the `ALLOW` address check are logged. |
| REJECTED | Only access attempts that failed the `ALLOW` address check are logged. |
| ALL | Both passed and rejected access attempts are logged. |

If a test initialization fails for a reason other than address checking, no entry is made in the audit file.

# AUDIT_FILENAME

This keyword specifies the filespec for the audit file. See *SECURITY_AUDITING* on page 2-3 to understand the types of events logged in its audit file. The default filename, in `endpoint.ini`, is `endpoint.aud`. If no drive or path is specified, the audit file uses the drive and path of the endpoint program.

This file contains at most two lines for each endpoint pair that is started on this endpoint. These two lines represent the start of an endpoint instance and the end of that instance.

Each line written to the audit file consists of a set of information about the endpoint instance and what it has been asked to do. The information is written in comma-delimited form, so you can load the audit file into a spreadsheet or database. When the audit file is created, an initial header line explains the contents of the subsequent entries.

The following table shows the fields of each entry in the audit file:

Table 2-3. Audit File Contents

| Field | Comment |
|---|---|
| Time | The date and time when the entry was created, in the local time zone. |
| Action | Whether this entry indicates that an endpoint instance was “Started” or “Ended.” |
| Endpoint | Whether the endpoint is in the role of Endpoint 1 or Endpoint 2. |
| Protocol of IxChariot Console | The network protocol used to contact Endpoint 1. |
| Network Address of IxChariot Console | The network address as seen by Endpoint 1. If you encounter problems setting up your `ALLOW` entries, this is the value to use for the protocol address. |
| Security Result | Whether this `SECURITY_AUDITING` “passed” or was “rejected.” If this is an entry for an “Ended” action, this field is reported as “n/a.” |
| Endpoint Partner Protocol | The network protocol used to run the test with our partner endpoint. |
| Endpoint Partner Address | The network address of our partner endpoint. |

# ENABLE_PROTOCOL

This keyword lets you control which connection-oriented protocols this endpoint uses to listen for setup connections. This does not affect the network protocols, which can be used to run tests. Here are the possible parameters:

```
ALL
APPC
SPX
TCP
```

In general, you should use the `ALL` setting (the default). Specify protocols explicitly to reduce the overhead of listening on the other protocols or if you're encountering errors when listening on the other protocols.

See the discussion of the ALLOW keyword (refer to *ALLOW* on page 2-2) for information about support of the datagram protocols, IPX, RTP, and UDP.

# Configuring Endpoints for Large-Scale Customization

To customize features such as automatic upgrades, you must edit the `endpoint.ini` file for each endpoint. For obvious reasons, you may not want to undertake such a potentially lengthy procedure. You can extract the files located in `gsendw32.exe` if you need to perform a large-scale customization of `endpoint.ini`. In addition to WinZip 7.0, you'll need the WinZip command-line support add-on and WinZip Self-Extractor. Here's how to use it:

1. Open the file `gsendw32.exe` using WinZip. See Using WinZip on page 17-5 for more information.
2. Extract the files to a temporary directory.
3. Edit or replace the `endpoint.ini` that is now in the temporary directory.
4. Using WinZip, create a new archive that contains all the files in the temporary directory.
5. Using the WinZip Self-Extractor, create a self-extracting executable; for the command line to run, enter the following:

   ```
   SETUP.EXE replace_ini
   ```

Now, anyone who executes the new executable you've created will automatically have the endpoint installed using the `endpoint.ini` file that you've customized.

To create a file that silently self-installs with a custom `endpoint.ini`, take the following steps:

1. Open the file `gsendw32.exe` using WinZip. See Using WinZip on page 17-5 for more information.
2. Extract the files to a temporary directory.

3. Edit or replace the `endpoint.ini` that is now in the temporary directory.
4. Create a custom response file (say, `customer.iss`); enter

```
SETUP -noinst -r -f1.\customer.iss
```

5. Using WinZip, create a new archive that contains all the files in the temporary directory.
6. Using the WinZip Self-Extractor, create a self-extracting executable; for the command line to run, enter the following:

```
SETUP.EXE replace_ini -s -f1.\CUSTOMER.ISS
```

Now, anyone who executes the file you've created will automatically have the endpoint installed using `customer.iss` as the response file, and the `endpoint.ini` file installed will also be the customized version you created.

# 3 Ixia

The Ixia Performance Endpoint allows Ixia load module ports to be used in much the same manner as other Performance Endpoints. Ixia Performance Endpoints, however, allow for:

- High-density of ports
- High-speed operation
- Ease of installation and maintenance

The Ixia Performance Endpoints package includes three distinct components:

- **Ixia Performance Endpoint Software** – this includes all of the software normally loaded on an IxChariot endpoint machine. This software is automatically downloaded to the Ixia ports as needed and no per-port installation or maintenance is required.
- **Ixia Application Preparation Utility (IxApplifier)** – this utility provides the means by which IP addresses and other endpoint characteristics are assigned to Ixia Performance Endpoints. It also is the mechanism by which the Ixia Performance Endpoint Software is downloaded to each Ixia port.

In order to use Ixia Performance Endpoint software, you must use the following Ixia equipment:

- IXIA 400T, 1600T, 250 or optixia chassis, running:
  - Windows 2000
  - IxOS software version 3.70 or later. This includes any later major version, such as 3.80.

**Note:** You must install the Tcl Server component of IxOS on each chassis that will be used by IxChariot.

- With one or more of the following Ixia load modules:[1]
    - LM1000TXS4 – 4-port 10/100/1000 Mbps Ethernet
    - LM1000STXS4 – 4-port 10/100/1000 Mbps Ethernet with dual copper/fiber interfaces
    - LM1000STXS24 – 24-port 10/100/1000 Mbps Ethernet with dual copper/fiber interfaces for use with optixia chassis
    - LM1000SFPS4 – 4-port 1000 Mbps Ethernet
    - LM100TXS8 – 8-port 10/100 Mbps Ethernet
    - ALM1000T8 – 8 port 10/100/1000 Mbps Ethernet
    - LM622MR – 2 port ATM and Packet Over Sonet
    - LM10GE700F1B-P - 10Gigabit Ethernet port with Xenpak interface and advanced processor

  For all load modules, refer to the *Ixia Hardware Guide* for full load module specifications.

- A remote IxChariot Console, running
  - Any of the following Windows Operating Systems:
    - Windows 2000 with Service Pack 2
    - Windows XP with Service Pack 1a
    - Windows NT with Service Pack 6a.

**Note:** In prior versions of this software it was necessary to install Ixia software on the IxChariot Management Station. This is no longer necessary; the version of IxApplifier included with this package will support all Ixia chassis running version 3.70 or later versions.

**Note:** Although a chassis may be used as the IxChariot Console, this configuration should not be used except in small test cases.

**Note:** When running an IxChariot test using Ixia Performance Endpoint where statistics are being gathered, avoid starting other IxChariot or non-IxChariot tests on the same Ixia chassis. Statistics data may otherwise be lost.

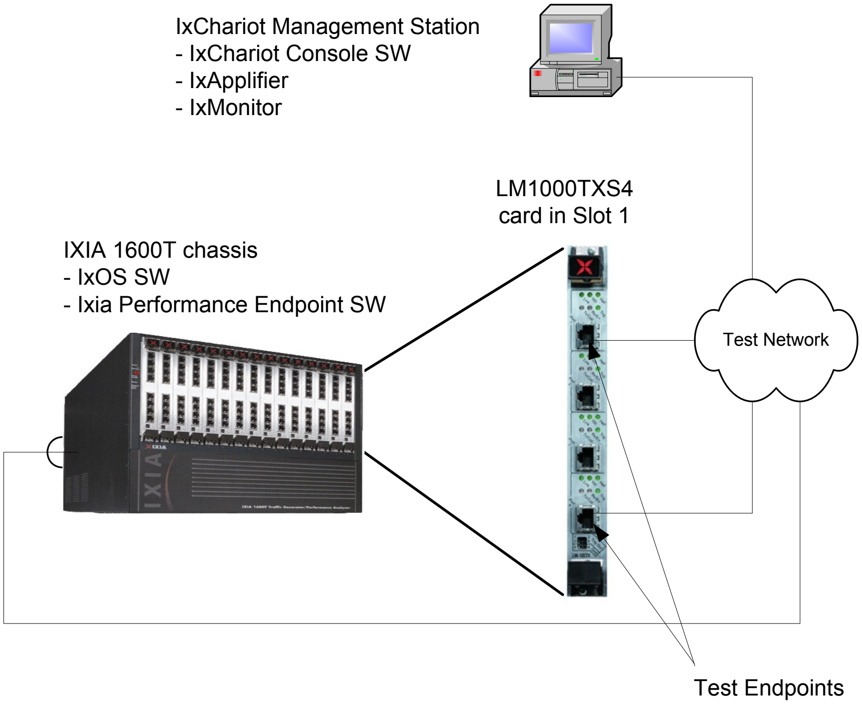
[1] Note the following measured limits on the number of pairs supported by these load modules: 500 for the LM1000TXS4 and ALM1000T-8 and 200 for the LM1000TXS8. The LM622MR’s limits have not been quantified yet.

# Theory of Operation

## Single Network Operation

One possible test model with Ixia Performance Endpoints is illustrated in Figure 3-1.

Figure 3-1. IxChariot test model - using one network



In this model, Ixia ports in an Ixia chassis substitute for the test systems. The components in this model are:

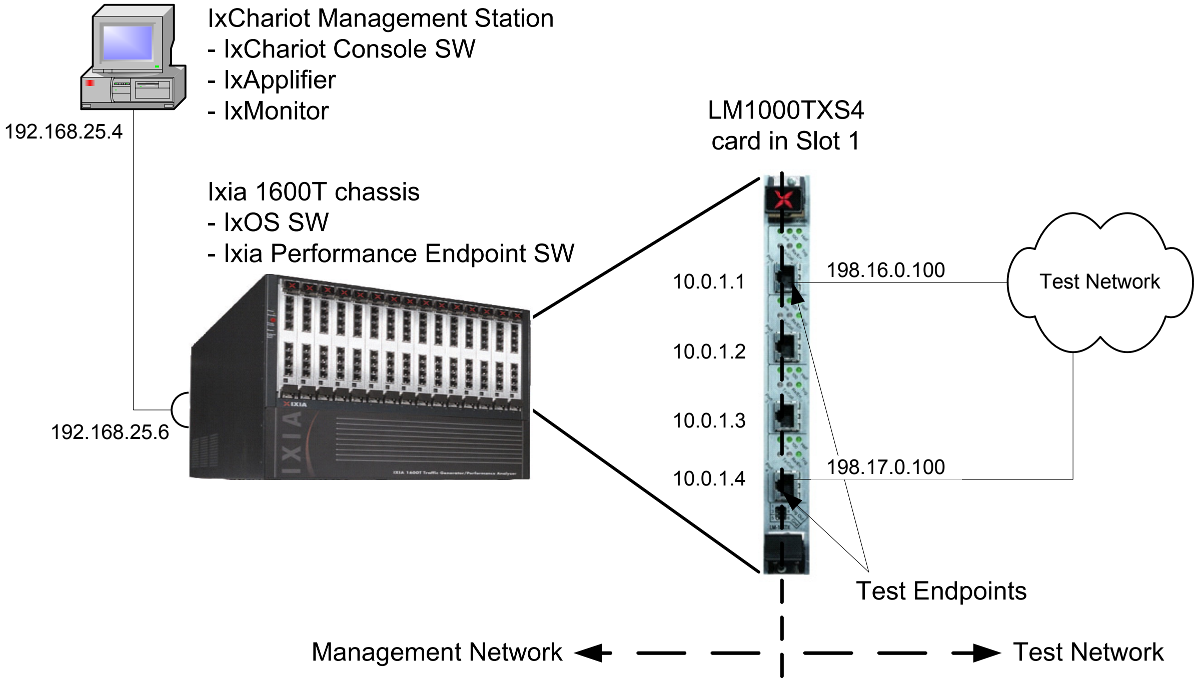
- **IXIA 1600T chassis**–holds the Ixia load modules and must be loaded with appropriate software. Although only one chassis is shown, multiple Ixia chassis may be used; their sync out/sync in cables must be daisy chained.
  - IxOS software.
  - IxChariot Ixia Performance Endpoint Software–this is a set of IxChariot endpoint software specifically compiled and packaged to execute on the processors embedded on the Ixia load modules.
- **Ixia Load Module**–one of the set of applicable Ixia load modules installed in the Ixia chassis. The list of applicable load modules is listed at the beginning of this chapter.
- **IxChariot Management Station**–a Windows-based PC which hosts:
  - IxChariot Console Software.
  - IxApplifier–a tool that sets up IP addresses and other configuration options on the Ixia ports. It also downloads the IxChariot Ixia Performance Endpoint Software to the Ixia ports.

- **Test Network**–a single network that is used to connect the management station, chassis and test ports. Note that the chassis has a network interface of its own, separate from the test interfaces. The test network, of course, contains the network equipment that is the object of the test.

## Two Network Configuration

Although the one network model is simple, it is not the recommended approach for IxChariot testing. When a single network is used, both management and test traffic travel over the same network, providing additional load to the network elements being tested. The preferred method uses two networks and serves to isolate test traffic from all other traffic. Figure 3-2 illustrates a two network configuration.

Figure 3-2. IxChariot test model - using two networks



This configuration logically divides the management network from the test network. The management network includes the IxChariot Management Station, Ixia chassis and the management addresses associated with the ports (10.0.1.1-10.0.1.4 in the figure. The test network contains the network components which are part of the test network (198.*.*.* in the figure).

> **Note:** Best test results will be obtained when the test network is devoid of all other network traffic.

## Setting up Ixia Ports

Ixia ports are general purpose Linux computers, with additional capabilities related to efficient network testing.

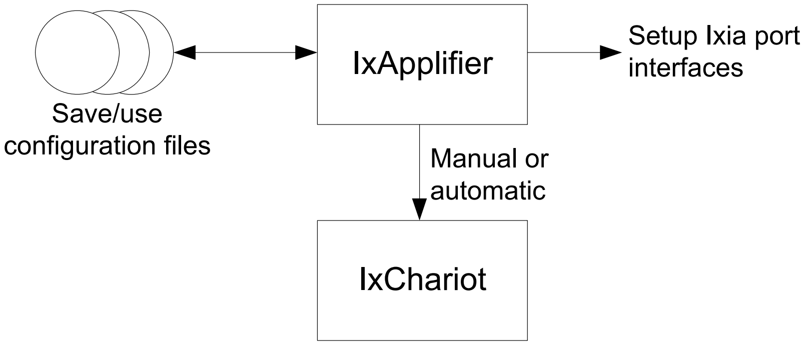
One or more interfaces on the ports must be setup with a number of standard networking characteristics:

- IP Address.
- MAC Address (stand-alone PCs have a hardware configured MAC address, whereas Ixia ports have a programmed value).
- Routing table, including a default gateway.
- VLAN.
- DNS.

In addition, Ixia ports allow filters to be applied to incoming network traffic – so as to ignore irrelevant traffic that would affect test-specific operation.

These parameters may all be set up from *IxApplifier*; the use of which is discussed in *Run IxApplifier* on page 3-27. IxApplifier is normally run before the IxChariot Console. In fact, IxApplifier can be configured to automatically start the IxChariot Console after ports have been configured. The normal flow of control is shown in Figure 3-3.

Figure 3-3. Flow of Control



IxApplifier is run first in order to setup the Ixia port interfaces. IxApplifier configurations may be saved and later re-used. After setting up the ports, IxApplifier may be configured to automatically start IxChariot.

# Installing Ixia Endpoints

The process of installing IxChariot involves two steps, one of which is performed on the chassis and the other on the client, management station.

## IxChariot Chassis Installation

Install IxOS on each chassis to be used for IxChariot testing.

It is important that as part of this installation that the *TCL Server* option be installed as well. If the chassis was installed without this option, you may re-use the installation media or file to add this component to the chassis installation.

Refer to the *Ixia Quick Start Guide* for installation instructions.

## IxChariot Management Station Installation

The IxChariot Ixia Performance Endpoint software should be loaded onto the console computer. Execute the IxiaPE.exe file provided to you electronically or on a CD, or at http://www.ixiacom.com/support/endpoint_library. The kit contains:

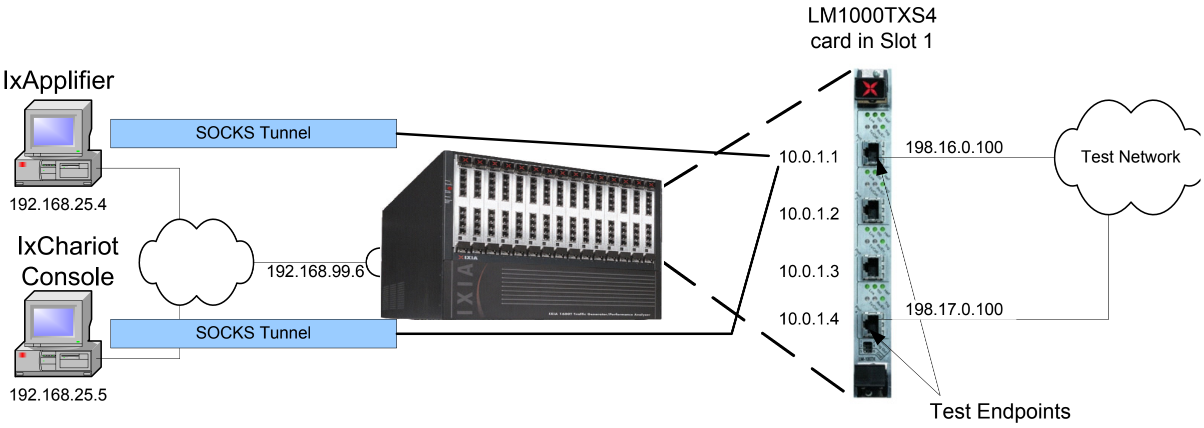
- IxChariot Ixia Performance Endpoint Software.
- IxApplifier.
- IxMonitor.

## Using Different PCs for IxApplifier and IxChariot Console Software

Although it is suggested that both IxApplifier and the IxChariot Console Software run on the same PC, it is possible to run the IxChariot Console Software on a separate PC.

When run on a separate PC, care must be taken with respect to the operation of SOCKS. SOCKS is implemented in a pair of software utilities, one running on a client PC and one running on the Ixia chassis. The pair co-operates to tunnel all traffic to the ports' management addresses through a connection between the client PC and the Ixia chassis. This is very useful if the client PC and the Ixia chassis are connected through a router because the ports' management addresses are normally in the 10.0.*.* range – which are often ignored by routers. This is shown in Figure 3-4.

Figure 3-4. SOCKS Usage



In Figure 3-4, the two PCs independently use SOCKS to create a tunnel from their network interface to the management address for the Ixia ports *through* the interface on the Ixia chassis (192.168.99.6). The SOCKS server on the Ixia chassis forwards the tunneled packets to the port interfaces.

IxApplifier is normally configured to use SOCKS for all chassis connections. When it is so configured, it creates a Windows registry setting and a *socks.cnf* file which controls the tunnel.

The IxChariot Console Software is ‘SOCKS ready’ and will use SOCKS if it is properly configured. When IxApplifier and the IxChariot Console Software are used on the same PC, IxApplifier will set up SOCKS and the Console will automatically use it.

When IxApplifier is not run on the machine that runs IxChariot Console Software and you wish to use SOCKS to avoid router problems, then SOCKS must be manually configured on the IxChariot Console machine in the following two steps:

1. Create a registry key using *regedit* for *HKEY_LOCAL_MACHINE\SOFTWARE\Ixia Communications\socks\config* of type *string value* with a contents of *C:\socks.cnf*.
2. Create the *C:\socks.cnf* file and fill in its contents as shown below:

```
SOCKD @=192.168.99.6 10.0.0.0 255.255.0.0
```

   Replace 192.168.4.169 with the IP address of your chassis and 10.0.0.0 with the base address of the chassis. Use additional lines for additional chassis.

If you do not wish to use SOCKS on the PC running only the IxChariot Console Software, then you must make sure that SOCKS is **not** configured to run. Using *regedit*, find and delete the *HKEY_LOCAL_MACHINE\SOFTWARE\Ixia Communications\socks\config* key. Remember that if you run IxApplifier on the PC, it will re-enable SOCKS.

# Configuring Ixia Endpoints

## Planning

IP network addresses must be assigned to the various components in the system. Refer to the figures in *Theory of Operation* for a description of alternate networking topologies. In some cases, existing addresses may need to be changed to conform to IxChariot rules.

The following networks must not overlap:

- The test network, including the test endpoint addresses.
- The networks between the Ixia chassis and management station.
- The networks corresponding to all the chassis base addresses. Remember that these are /16 networks.

## Step by Step Usage

This section describes the steps necessary to complete a IxChariot test.

The steps required to configure and run a IxChariot test are described in the following sections:

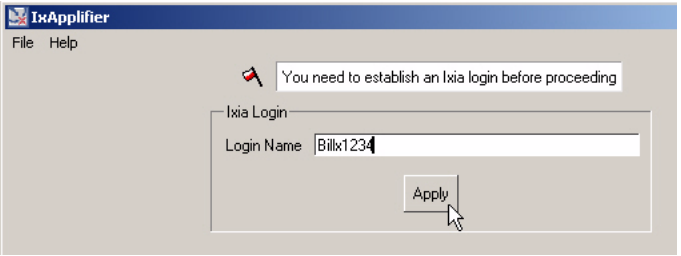
- *Step 1: Start IxApplifier and Login* on page 3-9
- *Step 2: Select Ports for Test* on page 3-11
- *Step 4: Configure Interface Addresses* on page 3-16
- *Step 5: Configure Routing* on page 3-20
- *Step 6: Configure Filters* on page 3-21
- *Step 7: Configure Network Settings* on page 3-24
- *Step 9: Run IxApplifier* on page 3-27

## Step 1: Start IxApplifier and Login

IxApplifier is started by double-clicking on the desktop icon that was created as part of the IxChariot installation. It may also be started through the Windows *Start* button: *Start ... Programs ... Ixia ... IxApplifier.*

The IxApplifier login dialog is shown in Figure 3-5.

Figure 3-5. IxApplifier Login Dialog



You need to establish an Ixia login before proceeding

Login Name: Billx1234

In order to prevent multiple users from attempting to use the same Ixia port, each port is considered *owned* by a particular user. On this screen, you are asked to provide a login name. This will be visible to other users who connect to Ixia chassis. The example above makes it clear to others how *Bill* can be reached - at extension 1234. More information on port sharing can be found in the *IxExplorer Users Guide: General Configuration and Operation.*

Press the *Apply* button to advance to the next step.

The menus and icons visible on the main screen of IxApplifier are described in Table 3-1. These menu choices are available throughout IxApplifier.

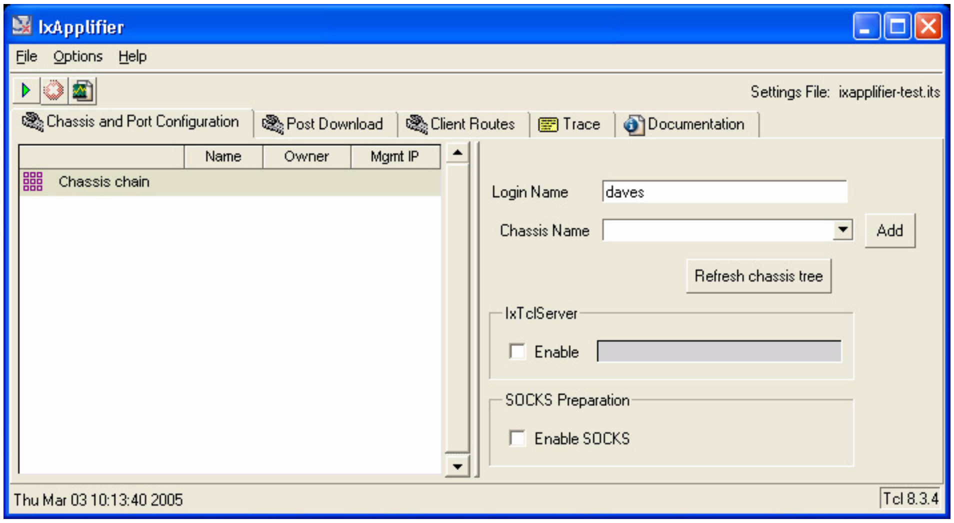
Table 3-1. IxApplifier Menus

| Menu/Icon | Choice | Usage |
|---|---|---|
| File Menu | Save Settings to File | The current configuration options are saved to a file specified in a standard *Save As* dialog. |
| | Retrieve Settings from File | The configuration file selected in a standard *Open* dialog is retrieved and displayed. |
| | New Settings File | This choice will prompt you for the name of a new settings file. |
| | Create Shortcut for Settings | A short-cut is placed on the desktop such that when it is used, the configuration named in the *Settings File* field will be applied to the ports and *IxChariot* will be invoked. The IxApplifier GUI will not be seen. |
| | Exit | Exits IxApplifier. |
| | Autoload Stored Settings Next Time | If checked, when IxApplifier starts it will reload the settings that were active from your last run. |
| Options | Stick to Current View While Running | This option affects the view that you see when the ▶ button is pressed. If selected, the current view at the time stays in place. Otherwise, the trace window is opened. |
| | Allow Multicast MAC Address | If checked, multicast MAC addresses will be allowed. |
| Help Menu | About | Displays the version of IxApplifier and Ixia IxOS software that are running. |
| ▶ | | Applies the port configuration to the port. |
| ○ | | Aborts the apply configuration. |
| 🗎 | | Validates your configuration settings. |

## Step 2: Select Ports for Test

The next step is to select the Ixia ports that you will use for testing, by selecting the *Chassis and Port Configuration* tab. The initial screen is shown in Figure 3-6.

Figure 3-6. Chassis and Port Configuration Screen



Settings File: ixapplifier-test.its

| Name | Owner | Mgmt IP |
|---|---|---|
| Chassis chain | | |

Login Name: daves

Chassis Name:

IxTclServer
☐ Enable

SOCKS Preparation
☐ Enable SOCKS

Thu Mar 03 10:13:40 2005

Tcl 8.3.4

### Chassis Selection

Ixia ports live inside Ixia chassis. A set of several chassis can be connected together with special timing cables to form a *Chassis Chain*[1]. In order to select the Ixia ports to be used during testing, it is necessary to first add the chassis to the chassis chain. A chassis may be added using the following steps:

1. Select the *Chassis Chain* line from the left-hand list, enter the chassis' host name or IP address in the *Chassis Name* field.
2. In the *IxTclServer* box, select the *Enable* checkbox. The name of your chassis will be copied into the field to the right. This indicates that *IxApplifier* will control your Ixia ports by communicating with the *IxTclServer* process on the indicated chassis. The setting for this server must be consistent for all chassis added, and may be the name of any of the chassis. If it is not running already running on the chassis, than the same media or file that was used to install the IxOS software may be reused in *Modify* mode to add *TCL Server*.

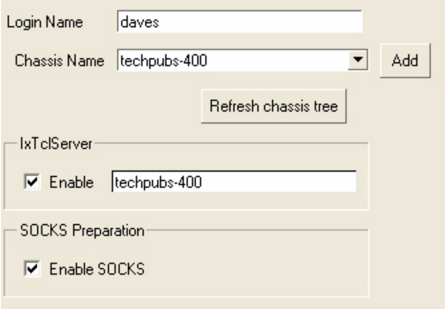
> **Note:** The Tcl Server component of IxOS must have been previously installed on the chassis. If Tcl Server is not running on the chassis, IxApplifier will fail to connect to the chassis.

3. In the *SOCKS Preparation* box, check the *Enable SOCKS.* The *Enable SOCKS* checkbox should be selected for *IxChariot.* Although we suggest that the *IxChariot Console Software* run on the same PC as *IxApplifier*, it is possible to run the *IxChariot Console Software* on a separate PC.

[1] When multiple chassis are used for IxChariot, timing cables need not connect the chassis together.

4. Press the *Add* button, as shown in Figure 3-7.

Figure 3-7. Chassis Added to Chassis Chain



Login Name: daves

Chassis Name: techpubs-400 Add

Refresh chassis tree

IxTclServer

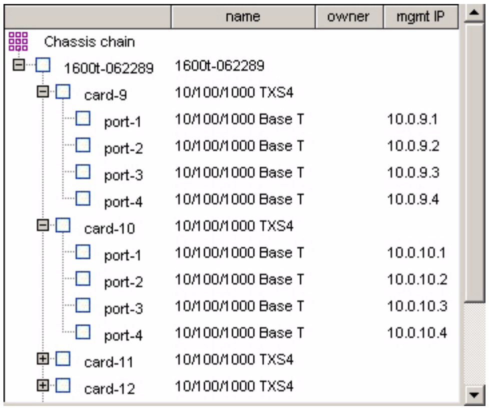
☑ Enable techpubs-400

SOCKS Preparation

☑ Enable SOCKS

The chassis is added to the chassis chain tree. Press the ⊞ buttons at each level to expose the cards and ports. An example is shown in Figure 3-8. Only those cards in the chassis that might be used for IxChariot testing are listed.

Figure 3-8. Ixia Ports Exposed in Chassis Chain Tree



| | name | owner | mgmt IP |
|---|---|---|---|
| Chassis chain | | | |
| ⊟ 1600t-062289 | 1600t-062289 | | |
| ⊟ card-9 | 10/100/1000 TXS4 | | |
| port-1 | 10/100/1000 Base T | | 10.0.9.1 |
| port-2 | 10/100/1000 Base T | | 10.0.9.2 |
| port-3 | 10/100/1000 Base T | | 10.0.9.3 |
| port-4 | 10/100/1000 Base T | | 10.0.9.4 |
| ⊟ card-10 | 10/100/1000 TXS4 | | |
| port-1 | 10/100/1000 Base T | | 10.0.10.1 |
| port-2 | 10/100/1000 Base T | | 10.0.10.2 |
| port-3 | 10/100/1000 Base T | | 10.0.10.3 |
| port-4 | 10/100/1000 Base T | | 10.0.10.4 |
| ⊞ card-11 | 10/100/1000 TXS4 | | |
| ⊞ card-12 | 10/100/1000 TXS4 | | |

The fields in this display described in Table 3-2.

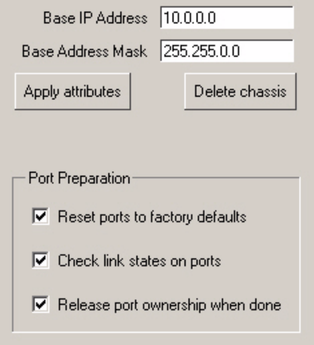
Table 3-2. Chassis Chain Column Interpretation

| Column | Usage |
|---|---|
| <first column> | The chassis name, card number in the chassis or port number on the card, depending on the row. |
| Name | The chassis name, card type or port type, depending on the row. |
| Owner | If a port is in use by another user, then the user's name appears here. |
| Mgmt IP | The management IP address of the Ixia port. This is the address that IxApplifier must use to program the ports. Management addresses are further discussed in *Ixia Port Management Addresses*. |

## Chassis Properties

Several properties related to a chassis are available for modification by selecting the chassis in the tree, as shown in Figure 3-9.

Figure 3-9. Chassis Properties



Base IP Address: 10.0.0.0

Base Address Mask: 255.255.0.0

Port Preparation

☑ Reset ports to factory defaults

☑ Check link states on ports

☑ Release port ownership when done

The properties in this dialog are explained in Table 3-3.

Table 3-3. Chassis Properties

| Property | Usage |
|---|---|
| Base IP Address | The base address associated with the chassis, which is used in the composition of ports' management addresses. See *Ixia Port Management Addresses* on page 3-13. |
| Base Address Mask | 255.255.0.0 is the default. The lowest two octets should never be used in the mask, since these are automatically set to indicate the card and port in use. |
| Reset ports to factory defaults | If this box is checked, when ports are setup, they are reset to their standard factory defaults. The factory defaults for a particular type of port can be checked through the use of *IxExplorer.* |
| Check link state on ports | If this box is checked, the link state of all ports involved in the test are checked. The test will only start if link has been established on all ports. |
| Release port ownership when done | Port ownership is taken in the *Chassis and Port Configuration* tab and used while a port is being configured. If this box is checked, when IxApplifier is exited port ownerships will be released. |

Press the *Apply attributes* button to associate these attributes with the selected chassis. The *Delete chassis* button may be used to remove a chassis from the chassis chain.

### Ixia Port Management Addresses

Each Ixia port has at least two IP addresses:

- **One or more test network addresses**–addresses on the test network used to carry test traffic.

- **A management address**–usually in the 10.0.*.* range, by which the management station sets up the port, including its test network address.

Since multiple Ixia chassis may be used in an IxChariot test, there must be a means of associating different networks with the ports on each chassis. This is accomplished by associating each chassis with a *port management network address*, also known as the *chassis base address.* The convention used to determine a port's management address is:

```
<base addr octet 1>.<base addr octet 2>.<card>.<port>
```

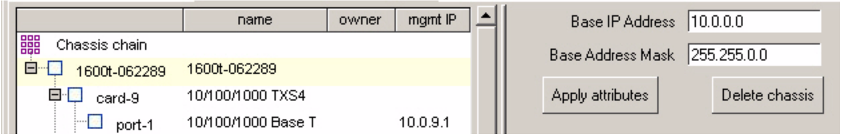
The first two octets of the address are called the base address associated with the chassis housing the card. The default chassis base address is `10.0.0.0`, `<card>` is the load module's slot number, and `<port>` is the port number on the load module. Therefore, the default addressing scheme is:

```
10.0.<card>.<port>
```

For example, using the default port management network address, card 2 port 3's internal IP address is `10.0.2.3`. If ports from more than one chassis are used in a test, then the base addresses of the ports must be different. In order to change a chassis' base address, select the chassis from the chassis chain tree and modify the *Base IP Address* and *Base Address Mask* as shown in Figure 3-10.

The low order 16 bits of the address are automatically set to correspond to the card and port. The *Base IP Address* and *Base Address Mask* should always set these two low order octets to 0's.

Figure 3-10. Setting a Chassis' Base IP Address



| | name | owner | mgmt IP |
|---|---|---|---|
| Chassis chain | | | |
| □ 1600t-062289 | 1600t-062289 | | |
| □ card-9 | 10/100/1000 TXS4 | | |
| □ port-1 | 10/100/1000 Base T | | 10.0.9.1 |

Base IP Address 10.0.0.0

Base Address Mask 255.255.0.0

> **Note:** Other Ixia products utilize the chassis base address. Care should be taken to contact other chassis users before changing a chassis' base address.

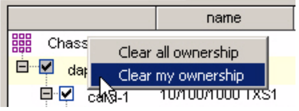
The chassis base address is needed while setting up a IxChariot test. In addition, when two networks are used, the IxChariot test must know one endpoint's management address.

## Port Selection

At this point, the ports to be used can be selected. Do this by pressing the □ box to the left of a chassis, card or port. Checking a card will check all ports on that card. Checking a chassis will check all ports on all cards on the chassis. A ☑ is used to indicate a level of the tree below which one or more, but not all elements are selected.

When you are done with a test and wish to release the ports to another user, you should return to this tab and either uncheck the box to the left of the port or card, or right-click on the *Chassis Chain* row, as shown in Figure 3-11.

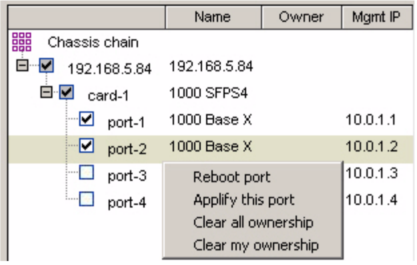
Figure 3-11. Releasing Ownership



Left-click on the *Clear my ownership* choice to release your ownership to another user. If it is necessary to clear another user's ownership, then the *Clear all ownership* choice may be used. Make sure not to clear ownership to ports that you don't need or don't own. Select the appropriate port, card or chassis before right-clicking.

In addition to clearing ownership at the port level, it is possible to perform two additional functions by right-clicking on a port, as shown in Figure 3-12.

Figure 3-12. Port Level Operations



| | Name | Owner | Mgmt IP |
|---|---|---|---|
| Chassis chain | | | |
| ☑ 192.168.5.84 | 192.168.5.84 | | |
| ☑ card-1 | 1000 SFPS4 | | |
| ☑ port-1 | 1000 Base X | | 10.0.1.1 |
| ☑ port-2 | 1000 Base X | | 10.0.1.2 |
| □ port-3 | | | 10.0.1.3 |
| □ port-4 | | | 10.0.1.4 |

The operations are:

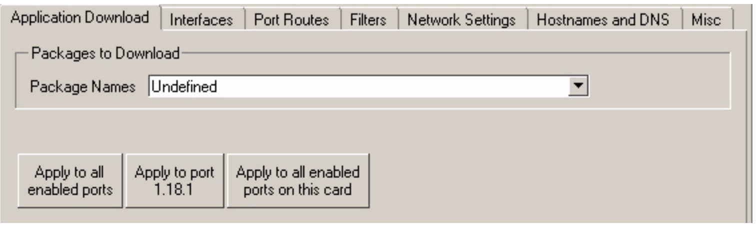
- Reboot port – if a port appears to be non-communicative or entirely non-functional, it may be remotely restarted by selecting this option. Allow a minute for the reboot to complete before attempting to use the port again.
- Applify this port – this choice will program the interfaces and other features of the port by itself, as opposed to all ports at the same time.

## Step 3: Configure Application Download

IxApplifier is a general front-end to a number of different applications. It provides a facility to download the necessary Linux-based port software to the ports when needed. It is necessary to tell IxApplifier which application software is to be downloaded.

Select any of the ports that are to be used in the IxChariot test, and then select the *Application Download* tab from the right-hand side, as shown in Figure 3-13.

Figure 3-13. Port Application Download



Package Names: Undefined

Apply to all enabled ports
Apply to port 1.18.1
Apply to all enabled ports on this card

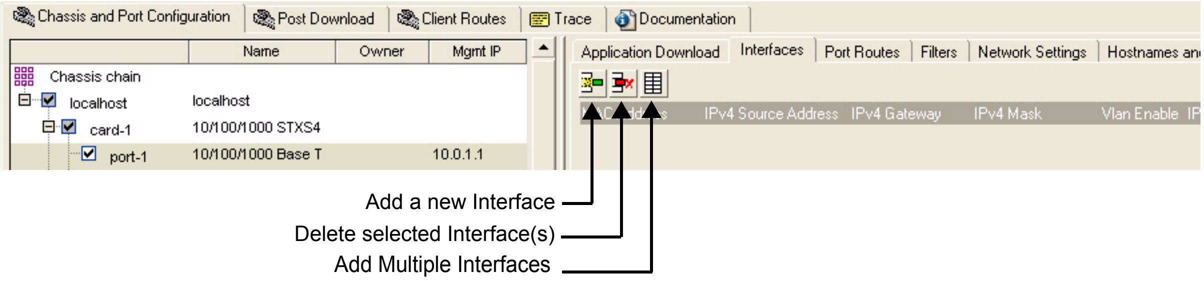
From the pull-down *Package Names* list, select *@ixChariot Performance Endpoint.* Press the *Apply to all enabled ports* button. This will cause the necessary IxChariot endpoint software to be downloaded to all selected ports at run time. The other two buttons provide for sophisticated situations in which non-IxChariot software may be downloaded to other ports. One of the three buttons must be pushed before switching to another tab, otherwise the setting will be lost.

The chassis base address is needed while setting up a IxChariot test.

## Step 4: Configure Interface Addresses

Select the *Interfaces* tab. The other tabs are explained in following steps. The *Interfaces* tab, without any interfaces defined is shown in Figure 3-14.

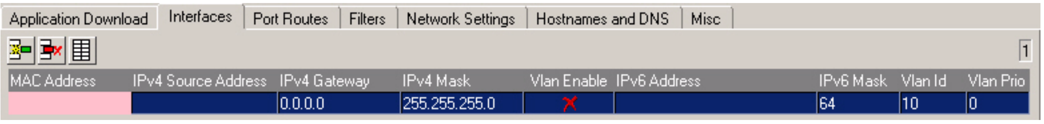
Figure 3-14. Configure Interfaces



| | Name | Owner | Mgmt IP |
|---|---|---|---|
| Chassis chain | | | |
| ☑ localhost | localhost | | |
| ☑ card-1 | 10/100/1000 STXS4 | | |
| ☑ port-1 | 10/100/1000 Base T | | 10.0.1.1 |

The *Add a new Interface* button will add a single interface, with default values. Pressing that button will add a new, blank row to the right hand table. The table entries may be directly edited. Each row allows an IPv4 and or IPv6 interface to be associated with a single MAC address. A completed entry is shown in Figure 3-15.

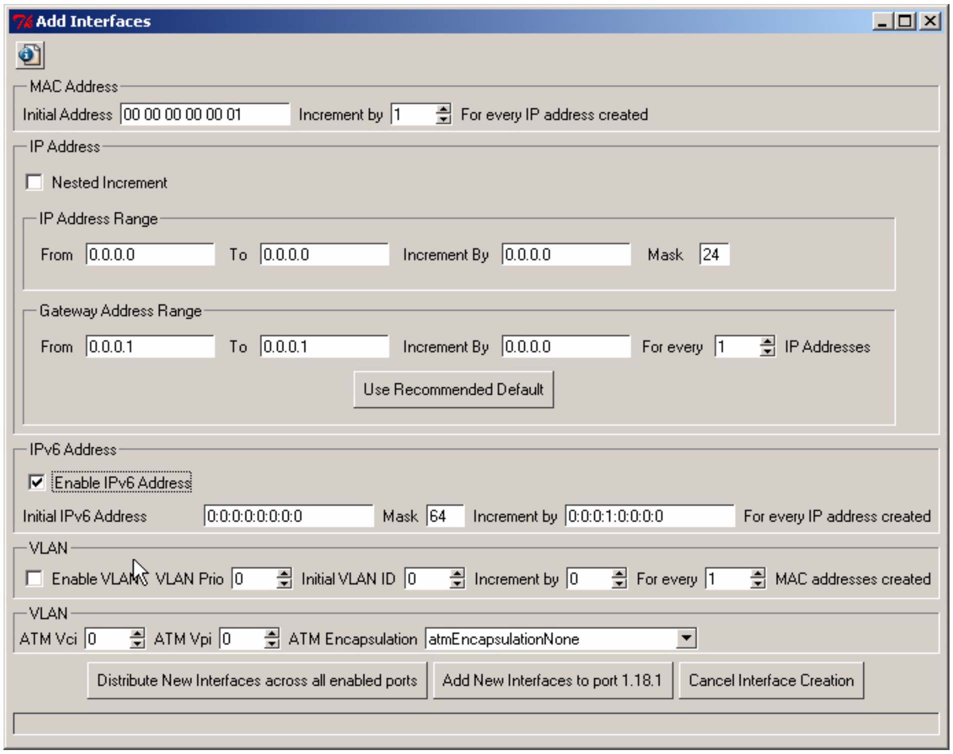
Figure 3-15. An Interface Entry



| MAC Address | IPv4 Source Address | IPv4 Gateway | IPv4 Mask | Vlan Enable | IPv6 Address | IPv6 Mask | Vlan Id | Vlan Prio |
|---|---|---|---|---|---|---|---|---|
| | | 0.0.0.0 | 255.255.255.0 | ✗ | | 64 | 10 | 0 |

The number in the upper right hand corner is the number of interfaces defined for the port. A more powerful means of entering interfaces is through the *Add Multiple Interfaces* button. This allows multiple interfaces to be assigned to a port and even to assign multiple interfaces across all of the enabled ports in the test. The dialog is shown in Figure 3-16 and the fields in the dialog are described in Table 3-4.

Figure 3-16. Add Interfaces Dialog



Add Interfaces

MAC Address

Initial Address 00 00 00 00 00 01 Increment by 1 For every IP address created

IP Address

□ Nested Increment

IP Address Range

From 0.0.0.0 To 0.0.0.0 Increment By 0.0.0.0 Mask 24

Gateway Address Range

From 0.0.0.1 To 0.0.0.1 Increment By 0.0.0.0 For every 1 IP Addresses

IPv6 Address

☑ Enable IPv6 Address

Initial IPv6 Address 0:0:0:0:0:0:0:0 Mask 64 Increment by 0:0:0:1:0:0:0:0 For every IP address created

VLAN

□ Enable VLANs VLAN Prio 0 Initial VLAN ID 0 Increment by 0 For every 1 MAC addresses created

VLAN

ATM Vci 0 ATM Vpi 0 ATM Encapsulation atmEncapsulationNone

Use Recommended Default
Distribute New Interfaces across all enabled ports
Add New Interfaces to port 1.18.1
Cancel Interface Creation

Table 3-4. Add Interfaces Dialog Fields

| Group | Field | Usage |
|---|---|---|
| MAC Address | Initial Address | The first MAC address to be used. A unique MAC address is assigned to each IP address. |
| | Increment by | Each new MAC address is incremented by this decimal value. |
| IP Address Range | Nested Increment | If unchecked, each IPv4 address between the *From* and *To* addresses is incremented by the *Increment By* address.<br>If checked, then each octet of the *Increment By* is incremented in a nested fashion. See below for a further discussion.<br>**Note:** Network addresses of all 0's and all 1's (e.g. x.x.x.0 and x.x.x.255 for a *Mask* value of 24) are skipped. |
| | From | The first IPv4 address generated, inclusive. |
| | To | The last IPv4 address generated, inclusive. |
| | Increment by | The IP address value to be added to the indicated byte for each new IPv4 address. |

Table 3-4. Add Interfaces Dialog Fields

| Group | Field | Usage |
|---|---|---|
| | Mask | The network mask associated with the IPv4 addresses. The mask value does not change across interfaces. |
| Gateway Address Range | From | The first IPv4 gateway address associated with the first interface. |
| | To | The last IPv4 gateway address to be generated. |
| | Increment by | The IP address value to be added to the indicated byte for each new IPv4 address. |
| | For every ... IP Address | The frequency with which the gateway address changes, with respect to generated IPv4 addresses. |
| | Use Recommended Default | Pressing this button will set the values in this group such that the '.1' address in each network is used as a gateway. It may be adjusted if some other network address is needed. |
| IPv6 Address | Enable IPv6 | If this box is checked, then an IPv6 address will be associated with each generated interface entry. |
| | Initial IPv6 Address | The first IPv6 address to be generated, in standard IPv6 format. |
| | Mask | The length of the IPv6 mask. |
| | Increment by... For every IP address created | The decimal value to be added to the indicated byte for each new IPv6 address. |
| VLAN | Enable VLAN | If checked, VLAN tagging is enabled for the interface. |
| | VLAN Prio | The VLAN priority to be associated with the VLAN ID. |
| | Initial VLAN ID | The first VLAN ID to be used. |
| | Increment by | The decimal increment to be applied to the VLAN ID when it is changed. |
| | For every ... MAC addresses created | The same VLAN ID may be used by multiple MAC addresses. This parameter indicates how many MAC addresses will use the same VLAN ID. |
| ATM VLAN | | This group of controls should be used to establish a VLAN on an ATM port through the use of a particular VPI/VCI. |
| | ATM VCI | The VCI to be associated with the interface. |

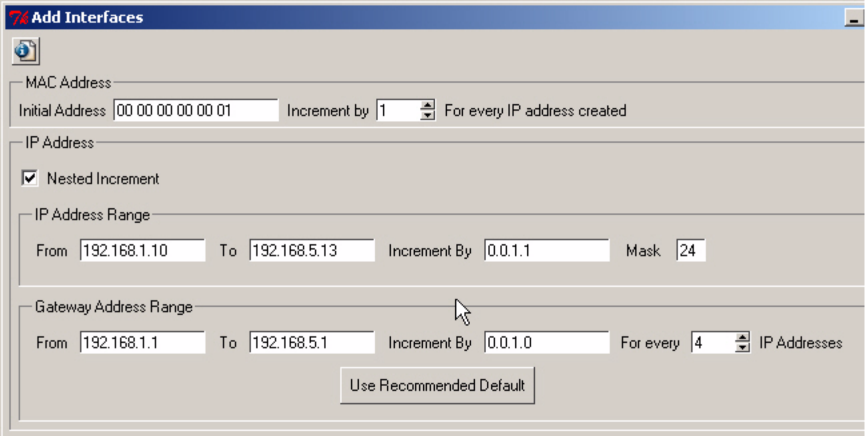
Table 3-4. Add Interfaces Dialog Fields

| Group | Field | Usage |
|---|---|---|
| | ATM VPI | The VPI to be associated with the interface. |
| | ATM Encapsulation | The ATM Encapsulation type. The encapsulations available are:<br>• None<br>• VCC MUX IPv4 Routed<br>• VCC MUX Bridged Ethernet with FCS<br>• VCC MUX Bridged Ethernet without FCS<br>• VCC MUX IPv6 Routed<br>• VCC MUX MPLS Routed<br>• LLC Routed CLIP<br>• LLC Bridged Ethernet with FCS<br>• LLC Bridged Ethernet without FCS<br>• LLC PPPoA<br>• VCC MUX PPPoA<br>• LLC NLPID Routed |
| Buttons | Distribute New Interfaces across all enabled ports | If this button is pushed, then all of the interfaces will be evenly distributed across all of the enabled ports. Any extra interfaces after division by the number of ports are discarded. |
| | Add New Interfaces to port x.y.z | If this button is pushed, then all of the interfaces are added to the currently selected port. |
| | Cancel Interface Creation | No interfaces are generated. |

## Nested Increment

The nested increment facility in this dialog allows ranges of addresses to be generated in multiple networks. For example, using nested increments it is possible to use 4 addresses in each of 5 networks. An example of this is shown in Figure 3-17.

Figure 3-17. Add Interfaces Dialog using Nested Increment



MAC Address
Initial Address: 00 00 00 00 00 01  Increment by: 1  For every IP address created

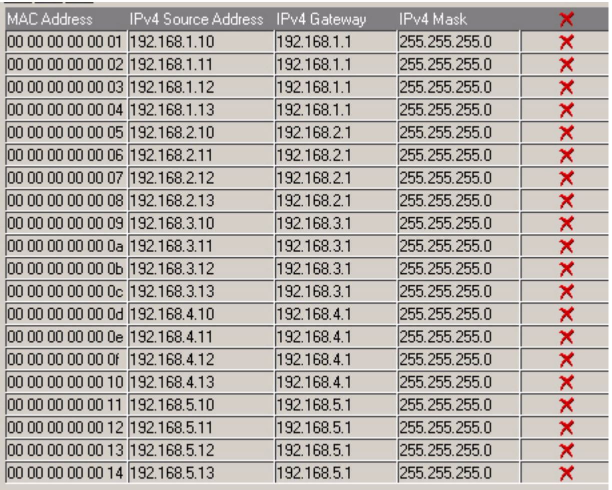
IP Address
☑ Nested Increment

IP Address Range
From: 192.168.1.10  To: 192.168.5.13  Increment By: 0.0.1.1  Mask: 24

Gateway Address Range
From: 192.168.1.1  To: 192.168.5.1  Increment By: 0.0.1.0  For every: 4  IP Addresses

The addresses that are generated are shown in Figure 3-18.

Figure 3-18. Generated Addresses



| MAC Address | IPv4 Source Address | IPv4 Gateway | IPv4 Mask | ✗ |
|---|---|---|---|---|
| 00 00 00 00 00 01 | 192.168.1.10 | 192.168.1.1 | 255.255.255.0 | ✗ |
| 00 00 00 00 00 02 | 192.168.1.11 | 192.168.1.1 | 255.255.255.0 | ✗ |
| 00 00 00 00 00 03 | 192.168.1.12 | 192.168.1.1 | 255.255.255.0 | ✗ |
| 00 00 00 00 00 04 | 192.168.1.13 | 192.168.1.1 | 255.255.255.0 | ✗ |
| 00 00 00 00 00 05 | 192.168.2.10 | 192.168.2.1 | 255.255.255.0 | ✗ |
| 00 00 00 00 00 06 | 192.168.2.11 | 192.168.2.1 | 255.255.255.0 | ✗ |
| 00 00 00 00 00 07 | 192.168.2.12 | 192.168.2.1 | 255.255.255.0 | ✗ |
| 00 00 00 00 00 08 | 192.168.2.13 | 192.168.2.1 | 255.255.255.0 | ✗ |
| 00 00 00 00 00 09 | 192.168.3.10 | 192.168.3.1 | 255.255.255.0 | ✗ |
| 00 00 00 00 00 0a | 192.168.3.11 | 192.168.3.1 | 255.255.255.0 | ✗ |
| 00 00 00 00 00 0b | 192.168.3.12 | 192.168.3.1 | 255.255.255.0 | ✗ |
| 00 00 00 00 00 0c | 192.168.3.13 | 192.168.3.1 | 255.255.255.0 | ✗ |
| 00 00 00 00 00 0d | 192.168.4.10 | 192.168.4.1 | 255.255.255.0 | ✗ |
| 00 00 00 00 00 0e | 192.168.4.11 | 192.168.4.1 | 255.255.255.0 | ✗ |
| 00 00 00 00 00 0f | 192.168.4.12 | 192.168.4.1 | 255.255.255.0 | ✗ |
| 00 00 00 00 00 10 | 192.168.4.13 | 192.168.4.1 | 255.255.255.0 | ✗ |
| 00 00 00 00 00 11 | 192.168.5.10 | 192.168.5.1 | 255.255.255.0 | ✗ |
| 00 00 00 00 00 12 | 192.168.5.11 | 192.168.5.1 | 255.255.255.0 | ✗ |
| 00 00 00 00 00 13 | 192.168.5.12 | 192.168.5.1 | 255.255.255.0 | ✗ |
| 00 00 00 00 00 14 | 192.168.5.13 | 192.168.5.1 | 255.255.255.0 | ✗ |

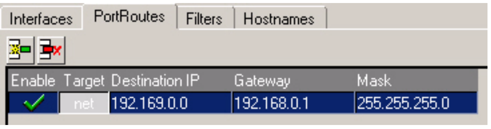
### Step 5: Configure Routing

The routing table for an Ixia port is set up with a single route that allows the port to send traffic to the network associated with the Ixia chassis. This is used for port management from the IxChariot Management Station.

If the DUT and all network elements of a IxChariot test are on the same network as the assigned interfaces, then no additional routes are needed. If, however, other networks are involved, then additional routes may be defined. Routes may be added to a port or all ports by selecting the *Port Routes* tab and pressing the

*Add a new entry* button. The entry may be directly edited, as shown in Figure 3-19.

Figure 3-19. Adding a Port Route



| Enable | Target | Destination IP | Gateway | Mask |
|---|---|---|---|---|
| ✓ | net | 192.169.0.0 | 192.168.0.1 | 255.255.255.0 |

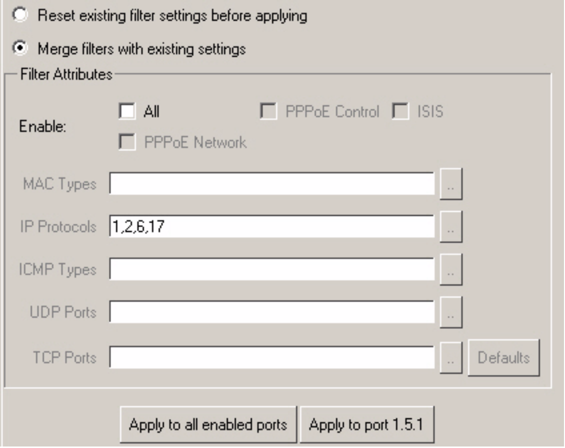
The entry may be edited in place; selecting the *Target* column of the row toggles the entry between a *host* entry and a *net* entry. Press the *Apply to all enabled ports* button to apply the port routes to all ports, or the *Apply to port x.y.z* button to use the routes for the currently selected port.

## Step 6: Configure Filters

Ixia ports have the unique ability to filter network traffic received on the port so as to only look at particular packets of interest. This allows the port's CPU to ignore irrelevant traffic and concentrate on efficient test operation. The filters are visible when the *Filters* tab is selected.

The default settings are shown in Figure 3-20. Table 3-5 explains the items in the dialog.

Figure 3-20. Default IxChariot Filters Settings



○ Reset existing filter settings before applying

● Merge filters with existing settings

Filter Attributes

Enable: □ All □ PPPoE Control □ ISIS □ PPPoE Network

MAC Types:

IP Protocols: 1,2,6,17

ICMP Types:

UDP Ports:

TCP Ports:

> **Note:** When using IxChariot, it is important that the first radio button (*Reset existing filter settings before applying*) be selected. This can be accomplished by selecting that choice and then pressing the *Apply to all enabled ports* button at the bottom of the page.

A filter must be set up on Endpoint 2 of the Endpoint Pair so as to only allow desired traffic from Endpoint 1 to pass. Failure to do this will potentially overwhelm the port's CPU and cause it not to generate any timing records.

The fields in the filter dialog (Figure 3-20) indicate which packet types, protocols and port numbers to **allow** through the filter and into the port's CPU. If a packet matches **any** of the criteria in the dialog, then the packet is allowed through. The filters are applied in the order shown; that is:

- MAC Types

- IP Protocols
- ICMP Types
- UDP Ports
- TCP Ports

This means, for example, that if *MAC Types* includes IP (0x800), then the *IP Protocols, UDP Ports* and *TCP Ports* fields are ignored – any IP packet will be accepted through the filter. Similarly, if *IP Protocols* includes TCP (6)*,* then the *TCP Ports* field will be irrelevant.

Table 3-5. Filters Tab Fields

| **Field** | **Usage** |
|---|---|
| Reset existing filter settings before applying | The filters described on this screen will replace all those currently configured in the port. This should always be the choice when using *IxChariot.* |
| Merge filters with existing settings | The filters described on this screen will be merged with those already configured on the port. |
| All | All traffic is enabled. All other fields are ignored. |
| ISIS | Traffic related to the IS-IS routing protocol. |
| PPPoE Control | Traffic related to PPPoE Discovery and control messages. |
| PPPoE Network | Non-control PPPoE traffic. |
| MAC Types | A list of comma separated MAC types to enable, in decimal or 0x<hex digits> format. Ranges may be separated by a dash ( - ). A blank entry signifies no MAC type filtering. |
| IP Protocols | A list of comma separated IP protocol numbers to enable, in decimal or 0x<hex digits> format. Ranges may be separated by a dash ( - ). A blank entry signifies no IP protocol filtering. |
| ICMP Types | A list of comma separated ICMP types to enable, in decimal or 0x<hex digits> format. Ranges may be separated by a dash ( - ). A blank entry signifies no ICMP type filtering. |
| UDP Ports | A list of comma separated UDP source or destination ports to enable, in decimal or 0x<hex digits> format. Ranges may be separated by a dash ( - ). A blank entry signifies no UDP port filtering.[a] |
| TCP Ports | A list of comma separated TCP source or destination ports to enable, in decimal or 0x<hex digits> format. Ranges may be separated by a dash ( - ). A blank entry signifies no TCP port filtering.[b] |
| Defaults | Restores the default settings for the filters associated with IxChariot. |

Table 3-5. Filters Tab Fields

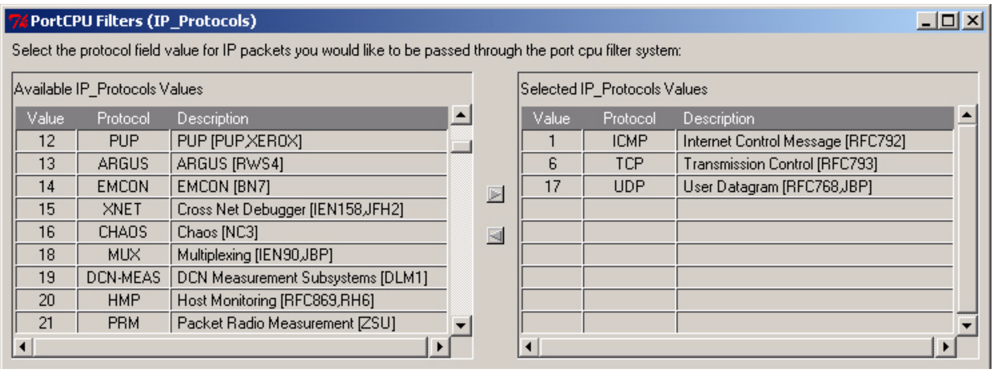
| Field | Usage |
|---|---|
| Apply to all enabled ports | If this button is pressed, then the same filters are applied to all enabled ports. |
| Apply to port x.y.z | If this button is pressed, then the filters are only applied to the currently selected port. |

a. Note that as of this writing, the UDP ports feature does not operate correctly when fragmented UDP packets are present.

b. Note that as of this writing, the TCP ports feature does not operate correctly when fragmented TCP packets are present.

The "..." next to each of *MAC Types, IP Protocols, ICMP Types, UDP Ports* and *TCP Ports* displays a dialog, which you may use to symbolically choose the values to be included in the list. Figure 3-21 shows the display for the *IP Protocols* ... choice.

Figure 3-21. IP Protocols choice box



PortCPU Filters (IP_Protocols)

Select the protocol field value for IP packets you would like to be passed through the port cpu filter system:

Available IP_Protocols Values

| Value | Protocol | Description |
|---|---|---|
| 12 | PUP | PUP [PUP,XEROX] |
| 13 | ARGUS | ARGUS [RWS4] |
| 14 | EMCON | EMCON [BN7] |
| 15 | XNET | Cross Net Debugger [IEN158,JFH2] |
| 16 | CHAOS | Chaos [NC3] |
| 18 | MUX | Multiplexing [IEN90,JBP] |
| 19 | DCN-MEAS | DCN Measurement Subsystems [DLM1] |
| 20 | HMP | Host Monitoring [RFC869,RH6] |
| 21 | PRM | Packet Radio Measurement [ZSU] |

Selected IP_Protocols Values

| Value | Protocol | Description |
|---|---|---|
| 1 | ICMP | Internet Control Message [RFC792] |
| 6 | TCP | Transmission Control [RFC793] |
| 17 | UDP | User Datagram [RFC768,JBP] |

Items are moved from the *Available* column to the *Selected* column by selecting the item from the left-hand list and pressing the ▶ button. Likewise, items may be moved back by selecting them from the right-hand list and using the ◀ button.

## Setting up Endpoint Filters

As noted earlier in this Step, it is necessary to set up filters for ports that are used as Endpoint 2 in an Endpoint Pair and subject to significant amounts of extraneous traffic. A filter ensures that the background traffic is filtered out before being sent to the port's CPU.

The configuration of the filter is dependent on the particular protocols and ports used by the two types of traffic. Three circumstances are described.

1. If distinct IP protocols are sent by the pair's Endpoint 1, then:
    - Select *Reset existing filter settings before applying* at the top of the dialog.
    - Ensure that the *MAC Types* field is empty.
    - In the *IP Protocols* box:
        - Include all the protocols used by the Endpoint Pair.

    - Exclude the protocols used by other network traffic.

2. If the same IP protocols are used by the pair's Endpoint 1 and other network traffic, but distinct destination[1] TCP or UDP ports are used, then:
    - Select *Reset existing filter settings before applying* at the top of the dialog.
    - Ensure that the *MAC Types* is empty and that the *IP Protocols* field does not include the type of port used in the next step.
    - In the *UDP Ports* and/or the *TCP Ports*:
        - Include all the ports used by the Endpoint Pair.
        - Exclude the protocols used by other network traffic[2].
3. If the same IP protocols and UDP/TCP destination ports are used by the pair's Endpoint 1 and other network traffic, then either the application script or the other network traffic (including Hardware Performance Pair traffic streams) must change its use of port numbers. The application script is probably the easier to change. Proceed as in step 2 above.

## Step 7: Configure Network Settings

A number of detailed controls are available for controlling network performance and need not be modified in most circumstances. The options available are discussed in Table 3-6.

Table 3-6. Network Settings

| Parameter | Description |
|---|---|
| Enable Explicit Congestion Notification | If enabled, the client or server uses bits from the Type of Service (TOS) field and the TCP packet header's Reserved field to support Explicit Congestion Notification (ECN). ECN uses the bits as follows:<br>• TOS bit 6 is the ECT (Explicit Congestion Transport) bit, which a recipient sets to indicate that it supports ECN.<br>• TOS bit 7 is the CE (Congestion Experienced) bit, which the recipient sets if its average queue length exceeds a threshold.<br>• TCP header Reserved field bit 5 the CWR (Congestion Window Reduced) bit.<br>• TCP header Reserved field bit 6 is the ECN-Echo bit<br>A SYN packet with both ECN-Echo and CWR bits indicates the sender supports ECN in both directions (as sender and receiver). A combined SYN+ACK packet sets only ECN to indicate ECN capability. |

[1] Source ports are not a reliable filter criteria.

[2] The port numbers used by the Ixia provided streams are described in the *IxChariot Application Scripts* manual, *Ixia Streams* chapter.

Table 3-6. Network Settings

| Parameter | Description |
|---|---|
| | If you enable ECN, the sender sets the ECT bit to indicate that it supports ECN. If the sender receives a packet with the CE bit set, it returns an ACK with the ECN-Echo bit set. If a sender receives a packet with with ECN-Echo set, it should reduce its window size. The first packet it sends after reducing its window size has the CWR bit set. The sender will repeatedly reduce its window size until it receives a packet in return with the CWR bit from the recipient. |
| Enable Time Stamp | If enabled, the client or server inserts a timestamp into each packet. |
| FIN Timeout | Time the client or server waits to receive a final FIN before closing a socket. A FIN Timeout is usually used to prevent denial-of-service attacks. |
| Keepalive Time | If a link has no activity on it for the time specified for Keepalive Time, the port begins sending keepalive probes to determine if the link is still up. |
| Keepalive Probes | Number of keepalive probes that the port sends out before determining that a link is down. |
| Keepalive Interval | Interval between repeated keepalive probes sent by the port. |
| SYN Retries | Number of times the port re-transmits an un-acknowledged SYN for an active TCP connection. This should not be higher than 255 and is only used for outgoing connections. |
| SYN ACK Retries | Number of times the port re-transmits an un-acknowledged SYN-ACK for a passive TCP connection. |
| SYN/ACK (passive open) Retransmit Retries | The number of times that an answer to a TCP connection request is retransmitted before giving up. |
| General Packet Retransmit Retries | Number of times the port attempts to re-transmit a packet for which it has not received an acknowledgement. |
| Transmit Buffer Size Min | The minimum size of the per socket send buffer. The buffer size may be decreased from its default size in low memory systems, but not below this size. |
| Transmit Buffer Size Default | The default size of the per socket send buffer. |
| Transmit Buffer Size Max | The maximum size of the per socket send buffer, which may be increased in systems with available memory. |
| Receive Buffer Size Min | The minimum size of the per socket receive buffer. The buffer size may be decreased from its default size in low memory systems, but not below this size. |

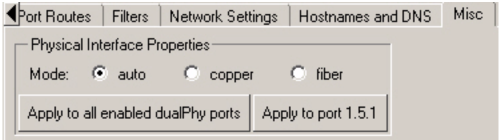
Table 3-6. Network Settings

| Parameter | Description |
|---|---|
| Receive Buffer Size Default | The default size of the per socket receive buffer. |
| Receive Buffer Size Max | The maximum size of the per socket receive buffer, which may be increased in systems with available memory. |
| Use Suggested Defaults | Restores the default settings for the settings associated with IxChariot. |
| Apply to all enabled ports | If this button is pressed, then the same settings are applied to all enabled ports. |
| Apply to port x.y.z | If this button is pressed, then the settings are only applied to the currently selected port. |

## Step 8: Misc Settings

Some of the Ixia cards have additional options specific to that card. These options may be seen on the *Misc* tab for the port. For example, Figure 3-22 shows the *Misc* tab for the LM1000STXS4 port type, which features the ability to use either an copper or fiber interface.

Figure 3-22. Misc tab for LM1000STXS4 ports



Mode: ● auto ○ copper ○ fiber

The choices on this dialog are described in Table 3-7.

Table 3-7. Misc dialog choices for LM1000STXS4 ports

| Category | Field | Usage |
|---|---|---|
| Physical Interface Properties | auto | Allow the port to choose the correct interface, based on signal availability. |
| | copper | Always select the copper interface |
| | fiber | Always select the fiber interface. |
| Buttons | Apply to all enabled dualPhy ports | Pressing this will apply this setting to all similar ports with dual copper and fiber interfaces. |
| | Apply to port ... | Pressing this will apply this setting to this port only. |

### Step 9: Run IxApplifier

Press the ▶ button at the top left of the IxApplifier screen to configure the ports. Specifically, IxApplifier will:

1. Validate that the base addresses of the chassis do not conflict.
2. Take ownership of the ports.
3. Optionally set all ports to factory defaults.
4. Optionally ensure that link has been established on all ports.
5. Configure MAC and IP addresses for the ports.
6. Configure routing, filters, network settings and name resolution for the ports.
7. Install and start the IxChariot package on the ports' processors.
8. If appropriately configured on the *Post Download* page, start *IxChariot.*

## Logging and Messages

The endpoint maintains logs in */var/log/endpoint.log.* The log file is created when an error occurs.

To view an error log, the log file should be moved over to the IxChariot Console and then viewed with the error log viewer, available in the Tools menu from the IxChariot Console main window.

### Message CHR0181

You may receive message **CHR0181** while running a test. If the error was detected at the Linux computer, it says that the endpoint program on Linux has run out of system semaphores. Each instance of Endpoint 1 requires a system semaphore. The maximum number of semaphores is not configurable on Linux, which is hard-coded to a large value (128). To avoid this problem, stop other programs that use semaphores or decrease the number of tests that use the computer as Endpoint 1.

## Starting and Stopping Ixia Endpoints

IxChariot endpoints on Ixia ports are automatically started when the Linux-based processor on the port is booted. If necessary, the procedures in this section can be used to stop the endpoint and restart it. One manner in which the endpoint may be restarted is to reboot the port using IxServer. This can be accomplished in one of three ways:

1. Restart *IxServer* on the chassis. This is the most extreme means of accomplishing the reboot. All use of all ports on the chassis will be immediately aborted. To accomplish this, you must:
   - a: Access the chassis' console.
   - b: Exit the running IxServer process. You will be asked for a confirmation of the termination; answer "yes".

c: Restart IxServer by double clicking the *IxServer* icon on the desktop.

2. Restart the individual ports using *IxServer*. To accomplish this, you must:
   a: Access the chassis' console.
   b: In the *IxServer* window, select *Tools..Diagnostics.*
   c: For each port with an IxChariot endpoint that needs to be restarted:
      i: Enter the card and the port in the fields provided.
      ii: Press the *LP Reboot* button.
3. Follow the two steps listed below.

## Stopping the Endpoint

In order to stop the IxChariot endpoint on an Ixia port, it is necessary to telnet to that port. The IP address of each port is of the form:

```
<base octet 1>.<base octet 2>.<card>.<port>
```

*Base octet 1* and *base octet 2* are the first two octets of the chassis base address, as described in *Ixia Port Management Addresses* on page 3-13. The default base address is *10.0.0.0*. *Card* and *port* are the card and port number of the individual port. Thus, to telnet to the first port on card three for a chassis with a default base address, one would type:

```
telnet 10.0.3.1
```

The *user* name is *root* and no password is needed.

Once you are logged in, you are talking to a Linux system. It is necessary to find and kill all endpoint processes. Use the following two steps:

1. Type: `ps | grep endpoint`.
2. For each of the numbers in the *pid* column, type the command:

```
kill <pid>
```

## Restarting the Endpoint

The IxChariot endpoint may be restarted using the following command, using the telnet session started in the previous section:

```
./bin/endpoint &
```

## Additional Notes

### Reserved TCP/UDP Ports

The Ixia ports run a version of the Linux operating system, complete with standard services. The TCP and UDP ports associated with those services may not be used by an IxChariot test. The following table lists the ports that must be avoided:

Table 3-8. Ixia port - reserved port usage

| Port Number | Use | TCP | UDP |
|---|---|---|---|
| 9 | Discard service | X | X |
| 21 | FTP daemon | X | |

Table 3-8. Ixia port - reserved port usage

| Port Number | Use | TCP | UDP |
|---|---|---|---|
| 23 | Telnet daemon | X | |
| 58 | Reserved | X | |
| 797-800 | NFS mounts | | X |
| 2600-2699 | Ixia reserved | X | |
| 3600-3699 | Ixia reserved | X | |
| 6002 | Capture service | X | |

In addition, the Ixia endpoint will use port 10115 as well as other ports which you assign in the Firewall Options tab.

## Getting the Latest Fixes and Service Updates

The latest version of the Ixia endpoint may be obtained at http://www.ixiacom.com/support/IxChariot.

# 4 Distributing Endpoints using SMS

**Related Topics**

*Installing Endpoints Using SMS* on page 4-1
*Uninstalling Endpoints Using SMS* on page 4-3

Endpoints can be installed and uninstalled on Windows computers automatically using Microsoft's Systems Management Server (SMS). This discussion assumes you are already familiar with package distribution via SMS.

- The SMS Server software must be installed and running properly on a Windows NT server.
- The SMS Client software must be installed and running properly on the Windows computers (that is, Windows 3.1x, plus all Win32 operating systems) where you want to remotely install endpoints. A folder titled "SMS Client" is present when the software has been installed correctly.

Our testing indicates that Version 1.2 of SMS (with Service Pack 2) or later is required.

## Installing Endpoints Using SMS

Follow these steps to install endpoints with SMS version 1.2.

1. If you are installing endpoints on Windows, you need to unzip the `gsendw32.exe` file from the CD. Refer to Using WinZip on page 17-5 for instructions.
2. Once the files are extracted and saved to the directory you selected, create a response file for each distinct set of client computers.

You need to create a response file (typically named `setup.iss`) for each unique installation. Each different operating system or target path is a unique installation. For example, you may have a set of Windows NT x86 computers where you want to install the endpoint in a directory named for our software (that is, `d:\Program Files\Ixia\Endpoint`) and another set where you want to install to a directory named `c:\Programs\Endpoint`. In this case, you would create two separate response files, one for each distinct set of installations.

To create a response file for a set of computers, go to one of the computers in the set and change the current working directory to the one where you extracted and saved the installation files for that computer. Enter a command like the following:

```
setup -noinst -r -f1d:\yourdirectory\setup.iss
```

It is important to run `SETUP` from that directory, because the version of `setup.exe` in your Windows directory will not work.

Here are the parameters for the `SETUP` command:

Table 4-1. SETUP Command Parameters

| Parameter | Comment |
|---|---|
| -noinst | No install: create the `setup.iss` file, but don't really install the endpoint right now. This is a Ixia-specific option and must appear before any setup-defined options, like "`-r`." |
| -r | Records the installation actions in an `.iss` file. |
| -f1 | Gives the path name for the output response file. |

1. Copy the endpoint installation files from the directory to a hard disk, along with the `setup.iss` file.
2. For each distinct set of client computers, create a directory on a hard disk available to the SMS Server. Into each directory, copy the corresponding endpoint installation files. In addition, copy the new `setup.iss` file you just created to the matching directory.

   For example, create directories on the SMS Server's hard disk named `\Endpoint_WNT1` and `\Endpoint_WNT2` for the two sets of client computers discussed in the preceding step. Copy all the unzipped installation files to each of these directories. Finally, copy the `setup.iss` file for the first set of client computers into directory `\Endpoint_WNT1`; copy the other `setup.iss` file into the second directory.
3. Inside the SMS program at the SMS Server, select **File**, then **New**. Click **Import**. Navigate to the drive and path where you've copied the endpoint installation files and their `setup.iss` file. Choose the corresponding `.pdf` file, which should be shown in the file list.

   A dialog box should appear showing the correct package installation information.
4. Click **Workstations**. In the dialog box that follows, move to the same drive and path you selected in step 3 by clicking the **"..."** symbol under "Source Directory." Then choose "Automated Installation" and click **Properties**. You should see the command line string necessary to install the endpoint, similar to the string you entered to create the `setup.iss` file.
5. Click **OK**, **Close**, and then **OK** to finish creating the SMS package. Repeat these steps for each distinct set of client computers.
6. Configure the packages at the SMS Server for your schedules and sites.

7. Decide when you want the endpoints installed, and on which computers. Configure these schedules and sites in SMS as you would with other SMS packages. See the SMS documentation for assistance.

Our software supports SMS Inventory Information, which has been encoded in the `.pdf` files.

# Uninstalling Endpoints Using SMS

Follow these steps to remove endpoint packages, using SMS version 1.2:

1. At the SMS Server, select a package to delete and update the name of the `Delsl?.isu` file.
2. Inside the SMS program at the SMS Server, select **File**, then **Open** the endpoint package you want to uninstall.
3. Click **Workstations**. In the dialog box that follows, move to the drive and path for the package by clicking the "..." symbol under "Source Directory." Then choose **Automated Uninstallation** and click **Properties**. It should show the command line string necessary to uninstall the endpoint, similar to the string you entered to create the `setup.iss` file. You should see a sequence that looks like "`fDelsl?.isu`" in the middle of the string. The "?" here is a number, representing the latest installation on the client computer. For example, if the endpoint has been installed twice, the client computer will have a file named "`Delsl2.isu`" in the directory where you installed the endpoint. This filename at the SMS Server must exactly match the filename at the SMS Client where the endpoint is being uninstalled.
4. Click **OK**, **Close**, and then **OK** to finish the update of the SMS package. Repeat these steps for each distinct set of client computers.
5. Configure the packages at the SMS Server for your schedules and sites.
6. Decide when you want the endpoints uninstalled, and on which computers. Configure these schedules and sites in SMS as you would with other SMS packages. See the SMS documentation for assistance.

# 5 HP-UX

This chapter explains the installation, configuration, and operation of the Performance Endpoint software for Hewlett-Packard's HP-UX 10.10 or later. (Because of their lack of effective multi-threading support, HP-UX versions 9.0 and earlier are no longer supported.)

## Installation Requirements for HP-UX Endpoints

Here's what you need to run the endpoint program with HP-UX:

- A Hewlett-Packard computer capable of running HP-UX.
- At least 32 MBytes of random access memory (RAM).
- The total RAM requirement depends on the RAM usage of the underlying protocol stack and the number of concurrent connection pairs. For large tests involving hundreds of connections through a single endpoint, additional memory may be required.
- A hard disk with at least 4 MBytes of space available.
- HP-UX version 10.10 or later, with TCP/IP networking and corresponding networking hardware installed and configured. This version also supports IP Multicast.
- An Acrobat Reader to view the `PDF` files.

Acrobat readers are loaded on most computers for viewing other documents, but if you do not have one, they are available at Adobe's Web Site: www.adobe.com/prodindex/acrobat/readstep.html.

> NOTE: in the following discussion, the name of the HP endpoint file is pehpx_*Mm*.tar, where *Mm* is the major and minor IxChariot version number; for example *520* for IxChariot release 5.20

# Endpoint Installation for HP-UX

First, ensure that you are logged in as a “root” user. Also, remember that all the commands and parameters discussed here are case-sensitive; use the combination of uppercase and lowercase letters as shown. The following instructions explain how to install an endpoint **from a CD-ROM** and **from the World Wide Web**.

To install the endpoint from a CD-ROM drive, do the following:

1. Put the CD-ROM in your CD-ROM drive.
2. Access to the CD-ROM is done through HP’s Portable File System (PFS). PFS should already be configured and running on your system. For detailed information about PFS, consult your HP-UX documentation. If PFS is not running, a quick way to start it is to enter the following commands:

   ```
   pfs_mountd -v &
   pfsd -v &
   ```

3. If you receive an error that `pfs_mount` is not found, the command `pfs_mount` is not in your path. To find where the command is located, enter the following commands:

   ```
   cd /
   find * -name pfs_mount -print
   ```

4. The directory where the `pfs_mount` command is stored will then be shown. You will need to enter this path before the `pfs_mount` command.
5. Assuming your CD-ROM drive device name is `c201d4s0` and the mount point is `/cdrom`, enter the following commands. Otherwise, enter your device name and mount point instead of `c201d4s0` and `/cdrom`.

   ```
   mkdir /cdrom
   echo “/cdrom” >>/etc/pfs_exports
   pfs_exportfs /cdrom
   pfs_mount -v -x unix -o ro /dev/dsk/c201d4s0 /cdrom
   ```

6. The CD-ROM contains an archive of the endpoint package. First use the `rm` command to ensure a clean temporary install directory. Then, use the `tar` command to extract the archive contents from the CD-ROM:

   ```
   cd /tmp
   rm -fr temp tar -xvf
   /cdrom/endpoint/hpux/pehpx_Mm.tar
   ```

7. Next, run the endpoint’s installation to install our software:

   ```
   ./endpoint.install
   ```

8. You will see the license agreement, presented with the `more` command. Press the spacebar until the end of the agreement is displayed. You are asked whether you accept the terms and conditions of the agreement. If you do, enter “`accept_license`.”
9. The endpoint installs itself in `/opt/Ixia.` During installation, you will see several status messages. Pay close attention to the output. If the installation is successful, you see the following message: “`Installation of endpoint was successful.`”

10. You may instead see the following message:

```
Notice! There were potential problems with migrating from
$oldInstallPath to $installPath. Review the warnings
displayed above for further explanation.
```

11. If you see this message, please review the entire output from the install script for an explanation of the warnings and further instructions.

12. After the installation is complete, use the `pfs_umount` command to unmount the file system from the CD-ROM:

```
pfs_umount /cdrom
```

13. If you need the disk space after installing the endpoint, you may delete the temporary directory and installation script. The installation script and temporary directory are not removed automatically.

14. To remove the temp files, enter:

```
rm -fr temp
rm endpoint.install
rm pehpx_Mm.tar
```

This is a good time to read the `README` file, installed with the endpoint in `/opt/Ixia`, for the latest information about the endpoint program. Use the following command to view the `README` file:

```
more /opt/Ixia/README
```

When you've completed installation, refer to *Configuring HP-UX Endpoints* on page 5-6 to make sure your endpoint is ready to be used in testing and monitoring.

To install an endpoint you've downloaded from the World Wide Web, do the following:

1. First, use the `rm` command to ensure a clean temporary install directory (we'll use `/tmp` in this example).

2. Download the `pehpx_Mm.tar.Z` file to the `/tmp` directory.

**Note**: The endpoint filename is `pehpx_Mm.tar.Z`;(with a capital "`Z`"); however, the Internet Explorer browser you use to download it changes the filename to all lowercase. Therefore, when you specify the filename in the Save As dialog box, you should capitalize the "`Z`" at that time.

3. Uncompress the endpoint by using the `uncompress` command:

```
cd /tmp
uncompress pehpx_Mm.tar
tar -xvf pehpx_Mm.tar
```

4. From the directory where you've downloaded the endpoint, run the endpoint's installation script:

```
./endpoint.install
```

5. You will see the license agreement, presented with the `more` command. Press the spacebar until the end of the agreement is displayed. You are asked whether you accept the terms and conditions of the agreement. If you do, enter "`accept_license`."

6. The endpoint installs itself in /opt/Ixia. During installation, you will see several status messages. Pay close attention to the output. If the installation is successful, you see the following message: "Installation of endpoint was successful."
7. You may instead see the following message:

```
Notice! There were potential problems with migrating from
$oldInstallPath to $installPath. Review the warnings
displayed above for further explanation.
```

8. If you see this message, please review the entire output from the install script for an explanation of the warnings and further instructions.
9. If you need the disk space after installing the endpoint, you may delete the temporary directory and installation script. The installation script and temporary directory are not removed automatically.
10. To remove the temp files, enter:

```
rm -fr temp
rm endpoint.install
rm pehpx_Mm.tar
```

This is a good time to read the README file, installed with the endpoint in /opt/Ixia, for the latest information about the endpoint program. Use the following command to view the README file:

```
more /opt/Ixia/README
```

When you've completed installation, refer to *Configuring HP-UX Endpoints* on page 5-6 to make sure your endpoint is ready to be used in testing and monitoring.

## Unattended Installation for HP-UX

Unattended installation is available for the HP-UX endpoint. You can install the endpoint silently, that is, without providing additional user input.

Complete the steps, as described in *Endpoint Installation for HP-UX* on page 5-2 through the tar command. Next, run the endpoint's installation, adding the "accept_license" parameter:

```
./endpoint.install accept_license
```

## What Happens During Installation

Here's what happens during the installation steps. The endpoint is installed into directory /opt/Ixia. The install directory is created with the following contents:

- The executable programs
- The README file
- Various install and uninstall programs
- Directory cmpfiles.

  This directory contains files with the .cmp file extension. These are files containing data of different types, such as typical text or binary data. These files are used by the endpoint as data on SEND commands. The different data types can be used to vary the data compression performance of your network hardware and software.

- File `endpoint.ini`

See Chapter 2, *Endpoint Initialization File* for information about tailoring this file for individual endpoints.

The installation program stops any copy of the endpoint program that may currently be running and starts a copy of the newly installed endpoint. You can run tests immediately, without a reboot.

No changes are made to the `PATH` environment variable of the root user.

Installation also performs the following additional actions:

- Copies a startup/shutdown script to the `/sbin/init.d` directory.
- Links the startup/shutdown script to `/sbin/rc2.d/S900endpoint`. This is invoked by HP-UX when the computer boots up.
- Links the startup/shutdown script to `/sbin/rc1.d/K100endpoint`. This is invoked by HP-UX when the computer is shut down.
- Copies a configuration file to the `/sbin/rc.config.d` directory. This file should be modified to control whether the endpoint starts when your system boots. By default, the endpoint will start upon reboot.

Should you have reason to install an older endpoint, you should delete any safestore files. **Take the following steps:**

1. Stop the endpoint.
2. Delete the safestore files from the endpoint directory (or from the directory specified by the `SAFESTORE_DIRECTORY` keyword in `endpoint.ini`). Safestore files have an extension of `.q*`; you may delete them using the command:

   ```
   rm *.q*.
   ```
3. Uninstall the current endpoint.
4. Install the desired endpoint.

## Removing the Endpoint Package (Uninstall)

Enter the following command to remove the endpoint (you must be logged in as root to run this program):

```
 /opt/Ixia/endpoint.remove
```

If the removal is successful, you see the following: “`Removal of endpoint was successful`.” This removes the files from `/opt/Ixia`, except for any files that were added to this directory that were not present at installation, such as the `endpoint.ini` file, or any other files you may need if you reinstall the product. For HP-UX version 10.10 systems, the removal script also leaves the `/opt/Ixia` directory.

# Configuring HP-UX Endpoints

The endpoint dynamically configures its own programs, so you do not have to update the configuration files for your communications software. However, your communications software must be configured and running correctly. The following steps guide you through this verification.

1. Determine the network addresses of the computers to be used in tests.
2. Verify the network connections.

Let's look at TCP/IP to see how to accomplish these tasks.

## Configuration for TCP/IP

The RTP, TCP, and UDP protocols use TCP/IP software for network communications. TCP/IP offers two forms of network addresses: IP addresses and domain names. An IP address is a 32-bit numeric address. It is represented in dotted notation as a set of four numbers separated by periods, such as `199.72.46.202`. An alternative, domain names are in a format that is easier to recognize and remember, such as www.ixiacom.com. To use domain names, you need either a Domain Name Server (DNS) set up in your network or an `/etc/hosts` file on each computer.

## Determining Your IP Network Address

Here are two ways to determine the IP address of the local computer you're using:

- If you're using Hewlett Packard's System Administration Manager (SAM) graphical user interface, first open the Networking/Communications menu, and from there select "Network Interface Cards." A window pops up with a list of interface cards and their IP addresses.
- Alternatively, enter the following at a command prompt:

```
netstat -in
```

You may have several network interfaces. If you are using a LAN network, for example, look at the output for the `lan0` interface; your local IP address is shown in the "Address" column.

## Testing the TCP/IP Connection

Ping is a simple utility program, included in all TCP/IP implementations. To check the connection from one computer to another, enter:

```
ping xx.xx.xx.xx 64 1
```

Replace the x's with the IP address of the target computer. If Ping returns a message that says

```
1 packets transmitted, 1 packets received, 0% packet loss
```

then the Ping worked. Otherwise, there will be a delay, and then you'll see

```
1 packets transmitted, 0 packets received, 100% packet
loss
```

This means that the Ping failed, and you can’t reach the target computer.

Make sure that you can run Ping successfully from the IxChariot or Qcheck Console to each computer serving as Endpoint 1, and between each pair of endpoints involved in a test, before starting your testing with TCP/IP.

### Sockets Port Number

The TCP/IP sockets port for endpoints is **10115**. This port number is used during the initialization of a test; during the actual running of the test, other port numbers are used. If the script specifies “`port_number=AUTO`” on the `CONNECT_ACCEPT` command, additional ports are dynamically acquired from the protocol stack. Otherwise, the endpoint issuing the `CONNECT_ACCEPT` commands (usually Endpoint 2) uses the port number specified in the script.TCP/IP applications use their network address to decide which computer to connect to in a network. They use a Sockets *port number* to decide which application program to connect to within a computer.

## Running HP-UX Endpoints

The following sections describe how to manually start and stop the endpoint program, and how to examine error log files if a problem occurs.

### Starting an HP-UX Endpoint

On HP-UX, the endpoint program is installed so that it starts automatically each time HP-UX is rebooted. Screen output goes to file `/var/opt/Ixia/endpoint.console`. If you want to see any error messages generated at this endpoint, enter the following command:

```
tail -f /var/opt/Ixia/endpoint.console
```

The detailed information about the start and stop of each individual connection pair is written to file `endpoint.aud`. The contents of this file vary depending on how you’ve set the `SECURITY_AUDITING` keyword in your `endpoint.ini` file.

See Chapter 2, *Endpoint Initialization File* for more information about `endpoint.aud` and `SECURITY_AUDIT` settings.

Instead of automatic startup, you can choose to manually start the endpoint program at a command prompt. Ensure that you are logged in as a “root” user. To start the endpoint, enter:

```
/opt/Ixia/endpoint &
```

The “&” “parameter indicates to HP-UX that the endpoint program should run in the background. The screen output from the endpoint program is interleaved with other UNIX commands. Just press Enter to enter more commands.

If you choose to manually start the endpoint, consider redirecting its output to the `endpoint.console` file. You can tell by the time stamp of the file when the endpoint program was started and stopped.

If the endpoint program is already running, you get the following message: "**CHR0183**: The endpoint program is already running. Only one copy is allowed at a time."

## Stopping an HP-UX Endpoint

The endpoint program has a special command-line option, -k. If you have an endpoint program you'd like to kill, go to a command prompt on the same computer and enter the following (you must be logged in as root to run this program):

```
/opt/Ixia/endpoint -k
```

The -k command-line option has the purpose of killing any endpoint process running on that computer. You should see the message "Sent exit request to the running endpoint," which indicates that the endpoint program has been sent a request to stop.

If for some reason the request to stop is not handled by the running endpoint program correctly, you may need to use the UNIX "kill -TERM" command. Avoid using "kill -9" to stop the running endpoint program -- it doesn't clean up what has been created (so you'll need to do the steps outlined in *Cleanup after Unexpected Errors* on page 5-8).

## Cleanup after Unexpected Errors

If the endpoint should fail or be killed abnormally (or encounter assertion conditions), you may also need to do additional cleanup. If the endpoint is still running, try to stop it using the command "endpoint -k". If that does not stop the endpoint, kill the endpoint using the UNIX kill command.

Next, enter the following command:

```
rm /var/opt/Ixia/.IXIA.ENDPOINT.PID
```

## How to Tell If an HP-UX Endpoint Is Active

You can use traditional UNIX commands to determine if the endpoint program is active. At a command prompt, enter the following:

```
ps -ef | grep endpoint
```

If the endpoint program is running, you will see output similar to the following:

```
root 2516    1 0 Apr 22 ?     0:00 /opt/Ixia/endpoint
```

## Disabling Automatic Startup

To disable automatic startup, edit the /etc/rc.config.d/endpoint file so that the START_ON_INIT variable is set to 0 (zero).

## Messages CHR0174, CHR0204, CHR0210, or CHR0245

You may see one of these error messages if you've exceeded the soft file limit per process allowed by HP-UX. You can verify this by examining the /var/opt/Ixia/endpoint.console file for the following text:

```
%Internal DCE Threads problem (version CMA BL10+),
terminating execution.
% Reason: cma__ts_open: fd is too large
% See 'cma_dump.log' for state information.
```

You may need to stop and restart the endpoint program using the methods outlined in *Starting an HP-UX Endpoint* on page 5-7 and *Stopping an HP-UX Endpoint* on page 5-8. You can use the HP-UX `SAM` facility to increase the number of open files allowed per process by changing the `maxfiles` kernel configurable parameters.

# Logging and Messages

Although most error messages encountered on an endpoint are returned to the IxChariot or Qcheck Console, some may be logged to disk. Errors are logged to file `/var/opt/Ixia/endpoint.log`. To view an error log, use the program named `FMTLOG`. `FMTLOG` reads from a binary log file, and writes its formatted output to `stdout`. Here is the syntax of the `FMTLOG` command:

```
/opt/Ixia/fmtlog log_filename >output_filename
```

For example, enter the following to write a readable ASCII version of the error log to a filename “`myoutput`”:

```
/opt/Ixia/fmtlog /var/opt/Ixia/endpoint.log >myoutput
```

The endpoint code does a lot of internal checking on itself. Our software captures details related to the problem in an ASCII text file. Assertion failures are written to the file `/var/opt/Ixia/assert.err`. Save a copy of the file and send it to us via email for problem determination.

## CORE and CMA_DUMP.LOG Files

We have seen situations where the endpoint core dumps on HP-UX, and the operating system writes a file named `cma_dump.log` to the directory `/opt/Ixia` or `/tmp`, and a file named `core` to `/opt/Ixia`. If a core dump occurs, please save a copy of the files `core` and `cma_dump.log` and return them to us for debugging.

## Message CHR0181

You may receive the error message CHR0181 while running a test. If the error was detected at the HP-UX computer, it says that the endpoint program on HP-UX has run out of system semaphores. Each instance of Endpoint 1 requires a system semaphore. You can use the HP-UX `SAM` facility to increase the number of available system semaphores. Use the following procedure to change the kernel configurable parameters:

This can be done using the HP-UX `SAM` facility:

1. As a root user, start `SAM` by typing `sam`.
2. Open the Kernel Configuration menu.
3. Open the Configurable Parameters menu.
4. Update the `semmap`, `semmni`, `semmns`, and `semmnu` parameters as necessary.

After changing the kernel parameters, you must reboot HP-UX to have the changes take effect. See the HP-UX System Administration Tasks manual for the definitions of these parameters.

# Updates for HP-UX

We've found that communications software is often fragile. Its developers are constantly working to make it more robust, as the software gets used in an ever-wider set of situations.

We therefore recommend working with the very latest software for the underlying operating system and communications software.

Check the following Web sites for code and driver updates:

- Hewlett-Packard's Web site: www.hp.com
- HP Electronic Support Centers:
  - http://us-support.external.hp.com/(US, Canada, Asia-Pacific, and Latin America)
  - http://europe-support.external.hp.com/(Europe)

6

# IBM AIX

This chapter explains the installation, configuration, and operation of the Performance Endpoint software for IBM's AIX on the RISC System/6000 (RS/6000).

## Installation Requirements for AIX Endpoints

Here's what you need to run the endpoint program with AIX:

- An IBM RS/6000 computer capable of running AIX.
- At least 32 MBytes of random access memory (RAM).

  The total RAM requirement depends on the RAM usage of the underlying protocol stack and the number of concurrent connection pairs. Large tests involving hundreds of connections through a single endpoint may require additional memory.
- A hard disk with at least 4 MBytes of space available.
- AIX version 4.1 or later, with TCP/IP networking and corresponding networking hardware installed and configured. This version also supports IP Multicast.
- An Acrobat Reader to view the PDF files.

  Acrobat readers are loaded on most computers for viewing other documents, but if you do not have one, they are available at Adobe's Web site: www.adobe.com/prodindex/acrobat/readstep.html.

NOTE: in the following discussion, the name of the HP endpoint file is peaix_*Mm*.tar, where *Mm* is the major and minor IxChariot version number; for example *520* for IxChariot release 5.20

## Endpoint Installation for AIX

First, ensure that you are logged in as a "root" user. Also, remember that all the commands and parameters discussed here are case-sensitive; use the combination of uppercase and lowercase letters as shown. The following instructions explain how to install an endpoint **from a CD-ROM** and **from the World Wide Web**.

To install the endpoint from a CD-ROM, do the following:

1. Put the endpoint CD-ROM in your CD-ROM drive.
2. Enter the following commands, assuming your CD-ROM drive device name is `cd0` and you're able to create a temporary directory named `cdrom`:

   ```
   mkdir /cdrom
   mount -v cdrfs -r /dev/cd0 /cdrom
   ```

3. The CD-ROM contains an archive of the endpoint package. First, use the `rm` command to ensure a clean temporary install directory. Then use the `tar` command to extract the archive contents from the CD-ROM:

   ```
   cd /tmp
   rm -fr temp
   tar -xvf /cdrom/endpoint/aix/peaix_Mm.tar
   ```

4. Next, run the endpoint's installation script to install our software:

   ```
   ./endpoint.install
   ```

5. You will see the license agreement, presented with the "`more`" command. Press the spacebar until the end of the agreement is shown. You are asked whether you accept the terms and conditions of the agreement. If you do, enter "`accept_license`" and press Return.

   The endpoint installs itself in `/usr/lpp/Ixia.` During installation, you will see several status messages. Pay close attention to the output. If the installation is successful, you see the message "`Installation of endpoint was successful.`"

   You may instead see the following message:

   ```
   Notice! There were potential problems with migrating from
   $oldInstallPath to $installPath. Review the warnings
   displayed above for further explanation.
   ```

   If you see this message, please review the entire output from the install script for an explanation of the warnings and further instructions.

6. After the installation is complete, use the `unmount` command to unmount the file system from the CD-ROM:

   ```
   umount /cdrom
   ```

If you need the disk space after installing the endpoint, you may delete the temporary directory and installation script. The installation script and temporary directory are not removed automatically.

To remove the temp files, enter:

```
rm -fr temp
rm endpoint.install
rm peaix_Mm.tar
```

This is a good time to read the `README` file, installed with the endpoint in `/usr/lpp/Ixia`, for the latest information about the endpoint program. Enter the `more` command to view the `README` file:

```
more /usr/lpp/Ixia/README
```

See *Configuring AIX Endpoints* on page 6-5 for information about your network connections.

If all connections are in order, you're ready to use this endpoint in testing and monitoring.

To install an endpoint you've downloaded from the World Wide Web, do the following:

1. First, use the `rm` command to ensure a clean temporary install directory. Then save the endpoint to that directory (we'll use `/tmp` in this example).
2. Download the `peaix_Mm.tar.Z` file to the `/tmp` directory.
3. Uncompress the endpoint file by using the `uncompress` command:

   ```
   cd /tmp
   uncompress peaix_Mm.tar
   tar -xvf peaix_Mm.tar
   ```
4. From the directory where you've downloaded the endpoint, run the endpoint's installation script to install our software:

   ```
   ./endpoint.install
   ```
5. You will see the license agreement, presented with the "`more`" command. Press the spacebar until the end of the agreement is shown. You are asked whether you accept the terms and conditions of the agreement. If you do, enter "`accept_license`" and press Return.

The endpoint installs itself in `/usr/lpp/Ixia.` During installation, you will see several status messages. Pay close attention to the output. If the installation is successful, you see the message "`Installation of endpoint was successful.`"

You may instead see the following message:

Notice! There were potential problems with migrating from $oldInstallPath to $installPath. Review the warnings displayed above for further explanation.

If you see this message, please review the entire output from the install script for an explanation of the warnings and further instructions.

If you need the disk space after installing the endpoint, you may delete the temporary directory and installation script. The installation script and temporary directory are not removed automatically.

**To remove the temp files, enter:**

```
rm -fr temp
rm endpoint.install
rm peaix_Mm.tar
```

This is a good time to read the `README` file, installed with the endpoint in `/usr/lpp/Ixia`, for the latest information about the endpoint program. Enter the `more` command to view the `README` file:

```
more /usr/lpp/Ixia/README
```

See *Configuring AIX Endpoints* on page 6-5 for information about your network connections.

If all connections are in order, you're ready to use this endpoint in testing and monitoring.

## Unattended Installation for AIX

Unattended installation is available for the AIX endpoint. You can install the endpoint silently, that is, without providing any additional user input.

Complete the steps, as described in *Endpoint Installation for AIX* on page 6-1 through the `tar` command. Next, run the endpoint's installation, adding the "`accept_license`" parameter:

```
./endpoint.install accept_license
```

## What Happens During Installation

Here's what happens during the installation steps. The endpoint is installed into the directory `/usr/lpp/Ixia`. A directory is created with the following contents:

- The executable programs.
- The `README` file.
- Various install and uninstall programs.
- Directory `cmpfiles`. This directory contains files with the `.cmp` file extension. These are files containing data of different types, such as typical text or binary data. These files are used by the endpoint as data on `SEND` commands. The different data types can be used to vary the data compression performance of your network hardware and software.
- File `endpoint.ini`
- See Chapter 2, *Endpoint Initialization File* for information about tailoring this file for individual endpoints.

The installation program stops any copy of the endpoint program that may currently be running and starts a copy of the newly installed endpoint. You can run tests immediately, without a reboot.

Our software does the following so the endpoint is started every time your system boots:

- Copies the `rc.ixia` initialization script to the `/etc` directory.
- Updates `/etc/inittab` to invoke `/etc/rc.ixia`

No changes are made to the `PATH` environment variable of the root user.

Should you have reason to install an older endpoint, you should delete any safestore files using the following steps:

1. Stop the endpoint.

2. Delete the safestore files from the endpoint directory (or from the directory specified by the `SAFESTORE_DIRECTORY` keyword in `endpoint.ini`). Safestore files have an extension of `.q*`; you may delete them using the command:

   ```
   rm *.q*.
   ```
3. Uninstall the current endpoint.
4. Install the desired endpoint.

### Removing the Endpoint Package (Uninstall)

Use the following command to remove the endpoint (you must be logged in as root to run this program):/usr/lpp/Ixia/endpoint.remove

If the removal is successful, you see the following: "`Removal of endpoint was successful.`"

This removes the files from `/usr/lpp/Ixia`, except for any files that were added to this directory that were not present at installation, such as the `endpoint.ini` file, and does not delete the directory. The remove program does not automatically delete files that have been added to the directory that you may need if you reinstall the product.

## Configuring AIX Endpoints

The endpoint dynamically configures its own programs, so you do not have to update the configuration files for your communications software. However, your communications software must be configured and running correctly. The following steps guide you through this verification.

1. Determine the network addresses of the computers to be used in tests.
2. Verify the network connections.

Let's look at TCP/IP to see how to accomplish these tasks.

### Configuration for TCP/IP

The RTP, TCP, and UDP protocols use TCP/IP software for network communications. TCP/IP offers two forms of network addresses: IP addresses and domain names. An IP address is a 32-bit numeric address. It is represented in dotted notation as a set of four numbers separated by periods, such as `199.72.46.202`. The alternative, domain names are in a format that is easier to recognize and remember, such as www.ixiacom.com. To use domain names, you need either a Domain Name Server (DNS) set up in your network or an `/etc/hosts` file on each computer.

### Determining Your IP Network Address

Here are two ways to determine the IP address of the local computer you're using:

- If you're using IBM's System Management Interface Tool (SMIT), first open the Communications Applications and Services menu, then the TCP/IP menu, and then the Minimum Configuration & Startup menu. Next, select the network interface used to reach other endpoints (for example, en0 or tr0). SMIT displays the network interface's configuration; your host's IP address is in the "Internet ADDRESS" field.

- Alternatively, enter the following at a command prompt:

```
netstat -in
```

You may have several network interfaces. If you are using a LAN network, for example, look at the output for the en0 interface; your local IP address is shown in the "Address" column.

### Testing the TCP/IP Connection

Ping is a simple utility program, included in all TCP/IP implementations. To try out the connection from one computer to another, enter:

```
ping xx.xx.xx.xx 64 1
```

Replace the x's with the IP address of the target computer. If Ping returns a message that says "1 packets transmitted, 1 packets received, 0% packet loss," the Ping worked. Otherwise, there will be a delay, and then you'll see the following:

```
1 packets transmitted, 0 packets received, 100% packet
loss
```

This means that the Ping failed, and you can't reach the target computer.

Make sure that you can run Ping successfully from the IxChariot or Qcheck Console to each computer serving as Endpoint 1, and between each pair of endpoints involved in a test, before starting your testing with TCP/IP.

### Sockets Port Number

TCP/IP applications use their network address to decide which computer to connect to in a network. They use a Sockets *port number* to decide which application program to connect to within a computer.

The TCP/IP sockets port for endpoints is **10115**. This port number is used during the initialization of a test; during the actual running of the test, other port numbers are used. If the script specifies "port_number=AUTO" on the CONNECT_ACCEPT command, additional ports are dynamically acquired from the protocol stack. Otherwise, the endpoint issuing the CONNECT_ACCEPT commands (usually Endpoint 2) uses the port number specified in the script.

## Running AIX Endpoints

The following sections describe how to manually start and stop the endpoint program, and how to examine error log files if a problem occurs.

### Starting an AIX Endpoint

The endpoint program is installed so that it starts automatically each time AIX is rebooted. It sends its screen output to file /var/adm/endpoint.console. If you want to see any error messages generated at this endpoint, enter the following command:

```
tail -f /var/adm/endpoint.console
```

The detailed information about the start and stop of each individual connection pair is written to file `endpoint.aud`. The contents of this file vary depending on how you've set the `SECURITY_AUDITING` keyword in your `endpoint.ini` file.

See Chapter 2, *Endpoint Initialization File* for more information about `endpoint.aud` and `SECURITY_AUDIT` settings.

Instead of automatic startup, you can choose to manually start the endpoint program at a command prompt. Ensure that you are logged in as a "root" user. To start the endpoint, enter the following:

```
/usr/lpp/Ixia/endpoint &
```

The "`&`" parameter indicates to AIX that the endpoint program should run in the background. The screen output from the endpoint program is interleaved with other UNIX commands. Just press Return to enter more commands.

If you choose to manually start the endpoint, consider redirecting its output to the `endpoint.console` file. You can tell by the time stamp of the file when the endpoint program was started and stopped.

If the endpoint program is already running, you get the following message: "**CHR0183**: The endpoint program is already running. Only one copy is allowed at a time."

## Stopping an AIX Endpoint

The endpoint program has a special command-line option, `-k`. If you have an endpoint program you'd like to kill, go to a command prompt on the same computer and enter the following (you must be logged in as root to run this program):

```
/usr/lpp/Ixia/endpoint -k
```

The `-k` command-line option has the purpose of killing any endpoint process running on that computer. You should see the message "`Sent exit request to the running endpoint`," which indicates that the endpoint program has been sent a request to stop.

If for some reason the request to stop is not handled by the running endpoint program correctly, you may need to use the UNIX "`kill -TERM" command`.

## Cleanup after Unexpected Errors

If the endpoint should fail or be killed abnormally (or encounter assertion conditions), you may also need to do additional cleanup. If the endpoint is still running, try to stop it using the command "`endpoint -k"`. If that does not stop the endpoint, kill the endpoint using the UNIX "`kill"` command.

Next, enter the following command:

```
rm /var/adm/.IXIA.ENDPOINT.PID
```

## How to Tell If an AIX Endpoint Is Active

You can use traditional UNIX commands to determine if the endpoint program is active. At a command prompt, enter:

```
ps -ef | grep endpoint
```

If the endpoint program is running, you will see output similar to this:

```
root 9888 1 0 19:19:54 - 0:00 /usr/lpp/Ixia/endpoint -G
7477 -T 3
root 7477 1 0 18:37:47 - 0:00 /usr/lpp/Ixia/endpoint
```

### Disabling Automatic Startup

To disable automatic startup, comment out or remove the following lines from the /etc/rc.ixia script:

```
if test -f $installPath/endpoint; then
echo "Starting the Ixia Endpoint."
$installPath/endpoint 1>$outputPath/endpoint console 2>&1
&
fi
```

## Logging and Messages

Although most error messages encountered on an endpoint are returned to the IxChariot or Qcheck Console, some may be logged to disk. Errors are saved in a file named *endpoint.log,* in the /var/adm directory. To view an error log, use the program named FMTLOG. FMTLOG reads from a binary log file, and writes its formatted output to stdout. Use the following FMTLOG command:

```
/usr/lpp/Ixia/fmtlog /var/adm/endpoint.log
>output_filename
```

The endpoint code does a lot of internal checking on itself. Our software captures details related to the problem in an ASCII text file named assert.err in the /var/adm directory. Save a copy of the file and send it to us via email for problem determination.

### Message CHR0181

You may receive message **CHR0181** while running a test. If the error was detected at the AIX computer, it says that the endpoint program on AIX has run out of system semaphores. Each instance of Endpoint 1 requires a system semaphore. The maximum number of semaphores is not configurable on AIX; it is hard-coded to a large value (4096). To avoid this problem, stop other programs that use semaphores, or decrease the number of connection pairs that use the AIX computer as Endpoint 1.

### Updates for AIX

We've found that communications software is often fragile. Its developers are constantly working to make it more robust, as the software gets used in an ever-wider set of situations.

We therefore recommend working with the very latest software for the underlying operating system and communications software.

Check the following Web site for code and driver updates:

http://techsupport.services.ibm.com/rs6000/support

# 7 Linux

This chapter explains the installation, configuration, and operation of the Performance Endpoint software for Linux.

We have concentrated our testing on the most popular Linux operating systems. The Performance Endpoint software requires kernel 2.0 with threading support. Our Linux endpoints run on the following Linux platforms:

- Red Hat Linux (version 6.1 and later ships with a current version of the Performance Endpoint) for x86 processors
- Slackware Version 2.0 and later (x86 processors)
- Cobalt Web Server RaQ (for MIPS processors), RaQ 2 (MIPS), and RaQ 3 (x86 processors)
- Other Linux systems that incorporate the Linux 2.0 kernel (or later) with threading support on x86 processors. This includes Caldera Systems, for example.

The version of the Linux endpoint that runs on RaQ and RaQ 2 MIPS processors has been archived at endpoint version 4.2. Thus, some functionality new in the latest versions of IxChariot may not be supported. The endpoint for RaQ 3 (x86 processors) is still fully supported and has been updated to the latest endpoint version.

Endpoints are also available for the Linux IA-64 processor architecture. See the documentation for Linux IA-64 for more information.

If you are using a Linux system other than Red Hat or the Cobalt Web Server, use the instructions listed for TAR-based (Slackware) installation and operation.

The Linux endpoint topics contain installation instructions for the following Linux configurations:

- *RPM-Based Endpoint Installation for Linux* on page 7-3
- *Cobalt-Based Endpoint Installation for Linux* on page 7-4
- *TAR-Based Endpoint Installation for Linux* on page 7-8

NOTE: in the following discussion, the name of the HP endpoint file is pelnx_*Mm*.tar, where *Mm* is the major and minor IxChariot version number; for example *520* for IxChariot release 5.20

## Installation Requirements for Linux Endpoints

Here is what you need to run the endpoint program with Linux:

- A computer capable of running Linux well.
  - For Cobalt servers, the RaQ or RaQ 2 systems, which use a MIPS processor, and the RaQ 3 system, which uses an Intel x86 processor, give good performance. Endpoint support for RaQ or RaQ 2 (that is, for MIPS) has been archived at endpoint version 4.2.
  - For x86 computers, this implies a CPU such as an Intel 80386, 80486, a member of the Pentium family, or equivalent. A Pentium or better is recommended.
- 16 MBytes of random access memory (RAM).
- The total RAM requirement depends on RAM usage of the underlying protocol stack and the number of concurrent connection pairs. For very large tests involving hundreds of connections through a single endpoint, additional memory may be required.
- A hard disk with at least 8 MBytes of space available
- Linux kernel 2.0 with “pthreads support” (which is at least version 2.0.6 of glibc). TCP/IP networking and corresponding networking hardware must be installed and configured, plus ELF support. Some older installations of Linux may not have this installed. At the Web site www.linuxdoc.org/HOWTO/Glibc2-HOWTO.html, you can find information about Linux, as well as download the file `glibc-2.0`, which you need to have loaded to install the endpoint. We have changed our installation procedures to check for this file, as it is required to run the endpoint.

  **i:** We have tested with Red Hat (kernel 2.0.32); Red Hat 5.0 or higher is required for IP Multicast. Red Hat 8.0 or higher is required for IPv6. We have also tested with Slackware 3.6.
- An Acrobat Reader to view the PDF files. Acrobat readers are loaded on most computers for viewing other documents, but if you do not have one, they are available at Adobe’s Web Site: www.adobe.com/prodindex/acrobat/readstep.html.

For Linux endpoints, there are three types of installation procedures. The basic procedure uses `TAR` files, which should be used for Slackware and Linux systems other than Red Hat or Cobalt. These systems have made changes to application installations that are described in later sections of this guide.

## RPM-Based Endpoint Installation for Linux

Use the RPM-based (x86 processor) installation if you are installing the endpoint on Red Hat Linux.

First, ensure that you are logged in as a "root" user. Also, remember that all commands and parameters discussed here are case-sensitive. Use the combination of uppercase and lowercase letters as shown in the following procedure. The following instructions explain how to install an endpoint from a CD-ROM and from the World Wide Web.

To install the endpoint from a CD-ROM, do the following:

1. Put the CD-ROM in your CD-ROM drive.
2. Enter the following commands, assuming your CD-ROM drive device name is `/dev/cdrom` and you are able to create a temporary directory named `cdrom`:

   ```
   mkdir /cdrom
   mount /dev/cdrom /cdrom
   ```

3. Copy the `pelnx_Mm.rpm` file from the CD-ROM to a local directory (like `tmp,` for example).

   ```
   cp /cdrom/endpoint/linux/pelnx_Mm.rpm /tmp
   ```

4. Use the `RPM` command to install the endpoint:

   ```
   rpm -Uvh /tmp/pelnx_Mm.rpm
   ```

   During installation, you will see several status messages. Pay close attention to the output. When the installation is successful, you see the message "`Installation of endpoint was successful.`"

   You may instead see the following message:

   ```
   Notice! There were potential problems with migrating from
   $oldInstallPath to $installPath. Review the warnings
   displayed above for further explanation.
   ```

   If you see this message, please review the entire output from the install script for an explanation of the warnings and further instructions.

5. After the installation is complete, use the `UMOUNT` command to unmount the file system from the CD-ROM.

   ```
   umount /cdrom
   ```

This is a good time to read the `README` file, installed with the endpoint in `/usr/local/ixia`, for the latest information about the endpoint program. Enter the `more` command to view the `README` file:

```
more /usr/local/ixia/README
```

When you've completed installation, refer to *Configuring Linux Endpoints* on page 7-11 to make sure your endpoint is ready to be used in testing and monitoring.

To install an endpoint you've downloaded from the World Wide Web, do the following:

1. First, use the rm command to ensure a clean temporary install directory (we'll use /tmp in this example).
2. Save the pelnx_Mm.rpm file to the /tmp directory.
3. Use the RPM command to install the endpoint:

   ```
   cd /tmp
   rpm -Uvh pelnx_Mm.rpm
   ```

   During installation, you will see several status messages. Pay close attention to the output. When the installation is successful, you see the message "Installation of endpoint was successful."

   You may instead see the following message:

   ```
   Notice! There were potential problems with migrating from
   $oldInstallPath to $installPath. Review the warnings
   displayed above for further explanation.
   ```

   If you see this message, please review the entire output from the install script for an explanation of the warnings and further instructions.

This is a good time to read the README file, installed with the endpoint in /usr/local/ixia, for the latest information about the endpoint program. Enter the more command to view the README file:

```
more /usr/local/ixia/README
```

When you've completed installation, refer to *Configuring Linux Endpoints* on page 7-11 to make sure your endpoint is ready to be used in testing and monitoring.

## Removing the RPM Endpoint Package (Uninstall)

Use the following command to uninstall the endpoint for Linux (you must be logged in as root to run this program):

```
rpm -e endpoint
```

If the removal is successful, you will see the following: "Removal of endpoint was successful." This removes the files from /usr/local/ixia, except for any files that were added to this directory that were not present at installation, such as the endpoint.ini file. This command does not delete the directory. The remove program does not automatically delete files added to the directory that you may need if you reinstall the product.

If anything goes wrong during the process of uninstalling the endpoint, a reinstalled endpoint may not run. You may need to do some extra cleanup. Check for the hidden file /usr/local/ixia/.IXIA.ENDPOINT.PID. You can use the command ls -a to view hidden files. Then enter the following command to delete it:

```
rm /usr/local/ixia/.IXIA.ENDPOINT.PID
```

## Cobalt-Based Endpoint Installation for Linux

Use the Cobalt installation if you are installing the endpoint for MIPS (RaQ and RaQ 2) or for x86 (RaQ 3) on a Cobalt Web server. There are some prerequisite levels of Cobalt software that are needed to operate the Performance Endpoint on a Cobalt Web server. Following are instruction for installing the endpoint from a CD-ROM and from the World Wide Web.

For Original Cobalt RaQ Products (whose model number starts with "R15"), you must upgrade two elements of the Cobalt software: `glibc` and the `kernel`. At this time, this cannot be done via the one-button package file installation mechanism. Using FTP and Telnet, take these steps:

1. As the root user, `FTP` the `glibc` and `kernel RPM` files to your RaQ. The filenames are: `glibc-2.0.7-9.mips.rpm` and `kernel-2.0.34-C18.mips.rpm`.
2. Install these files using the following commands:

```
% rpm -U glibc-2.0.7-9.mips.rpm
% rpm -U --force kernel-2.0.34-C18.mips.rpm
```

3. List all installed RPM files, using the following command to ensure that the new files were installed correctly:

```
% rpm  -qa
```

4. Reboot the server after installation to initialize the new files.

Cobalt RaQ 2 Products (MIPS processors whose model number starts with "R28") and RaQ 3 products (x86 processors) ship with newer versions of software that already supports the netiq endpoint software. Therefore, it is not necessary to install the `glibc` and `kernel` files.

**Note:** Typically, when you install an endpoint, you can automatically upgrade from the previous version of the endpoint. However, beginning with endpoint version 3.5, when you're upgrading the endpoint on a Cobalt RaQ 3 (x86) computer, you must first remove the previous version of the endpoint, following the directions in *Removing the Cobalt Endpoint Package (Uninstall)* on page 7-7. Then install the new version of the endpoint (see below). This limitation only applies to the Cobalt RaQ 3 endpoint.

If you attempt to install Endpoint 4.4 without first removing Endpoint 4.3, you get a message stating, "`Error uninstalling RPM`." This means that the upgrade failed.

**Installing the endpoint from a CD-ROM:**

The following instructions are for installing on the Cobalt Web server (for either MIPS or x86 processors) from a Web browser running on a Linux computer. You can, however, install the endpoint from computers running other platforms. In these cases, proceed with the following instructions, but do not mount the CD-ROM.

1. Put the CD-ROM in your CD-ROM drive.
2. Enter the following commands, assuming your CD-ROM drive device name is `/dev/cdrom` and you are able to create a temporary directory named `cdrom`:

```
mkdir /cdrom
mount /dev/cdrom /cdrom
```

3. Access the “Welcome to Cobalt” page on the Cobalt Web Server and click on the link to the RaQ Server Management section. The Username and Password Required dialog is shown.
4. Enter the username and password for Administrator.
5. Click **Maintenance** on the Server Management dialog box and then click **Install Software**.
6. In the **Software to install** field, enter the location of the package. If you are using the Browse function, make sure that the filename and extension are in lowercase.
   - To install on the RaQ or RaQ 2 (MIPS processor), use this example:

     ```
     /cdrom/endpoint/archive/cobalt/endcblr.pkg
     ```

   - To install on the RaQ 3 (x86 processor), use this example:

     ```
     /cdrom/endpoint/linux/endcbl3.pkg
     ```

7. If prompted, enter the password for Administrator. Click **Install a ‘.pkg’ package**. After the endpoint is installed, a message stating that the endpoint has been installed is shown. If you do not get this message, please go to the server management panel and browse the box labeled “Software on the Cobalt Server.” This contains a list of products installed on the Cobalt computer. You should see a line that reads, “`Performance Endpoint X.X`,” where `X.X` is the release number of the Performance Endpoint.
8. After the installation is complete, use the `UMOUNT` command to unmount the file system from the CD-ROM:

```
umount/cdrom
```

The installation script and temporary directory are not removed automatically if the installation is successful. If you need the disk space after installing the endpoint, you may delete the temporary directory and installation script.

This is a good time to read the `README` file, installed with the endpoint in `/usr/local/netiq/`, for the latest information about the endpoint program. Enter the `more` command to view the `README` file:

more /usr/local/netiq/README

When you’ve completed installation, refer to *Configuring Linux Endpoints* on page 7-11 to make sure your endpoint is ready to be used in testing and monitoring.

**Installing an endpoint you’ve downloaded from the World Wide Web:**

1. First, use the `rm` command to ensure a clean temporary install directory (we’ll use `/tmp` in this example).
2. Save the file appropriate for your operating system to the `/tmp` directory.
3. Access the “Welcome to Cobalt” page on the Cobalt Web Server and click on the link to the RaQ Server Management section. The Username and Password Required dialog is shown.
4. Enter the username and password for Administrator.

5. Click **Maintenance** on the Server Management dialog box and then click **Install Software**.
6. In the **Software to install** field, enter the location of the package. If you are using the Browse function, make sure that the filename and extension are in lowercase.
   - To install on the RaQ or RaQ 2 (MIPS processor), use this example:

     ```
     /tmp/endcblr.pkg
     ```
   - To install on the RaQ 3 (x86 processor), use this example:

     ```
     /tmp/endcbl3.pkg
     ```
7. If prompted, enter the password for Administrator. Click **Install a '.pkg' package**. After the endpoint is installed, a message stating that the endpoint has been installed is shown. If you do not get this message, go to the server management panel and browse the box labeled "Software on the Cobalt Server." This contains a list of products installed on the Cobalt computer. You should see a line that reads, "`Performance Endpoint X.X,`" where `X.X` is the release number of the Performance Endpoint software.

When you've completed installation, refer to *Configuring Linux Endpoints* on page 7-11 to make sure your endpoint is ready to be used in testing and monitoring.

## Removing the Cobalt Endpoint Package (Uninstall)

You must be logged in as the root user to remove the endpoint package. Do not use the `RPM` command to remove the Cobalt endpoint. First, stop the endpoint program (if it's running).

Next, enter the following command at a command prompt:

```
/bin/sh /var/lib/cobalt/uninstallers/endpoint-4.4.uninst
```

**Note**: The actual command depends on the version of the endpoint installed. If you have a different version installed, for instance version 4.3, run the "`endpoint-4.3.uninst`" command instead. The endpoint for Cobalt RaQ and RaQ 2 is archived at version 4.2.

After the script removing the endpoint package has completed, enter the following command:

```
/usr/admserv/cgi-bin/.cobalt/install/install.cgi
```

At the prompt, press the **Ctrl +D** keys. The endpoint has been removed.

If anything goes wrong during the process of uninstalling the endpoint, a reinstalled endpoint may not run. You may need to do some extra cleanup. Check for the hidden file `/usr/local/netiq/.NETIQ.ENDPOINT.PID` by using the `ls -a` command. Enter the following:

```
rm /usr/local/ixia/.NETIQ.ENDPOINT.PID
```

## TAR-Based Endpoint Installation for Linux

Use the TAR-based installation if you are installing the endpoint on any x86 Linux platform other than Red Hat or Cobalt RaQ 3 Web Server.

First, make sure that you are logged in as a "root" user. Also, remember that all commands and parameters discussed here are case-sensitive. Use the combination of uppercase and lowercase letters as shown in the following procedure. The following instructions explain how to install an endpoint from a CD-ROM and from the World Wide Web.

**To install the endpoint from a CD-ROM, do the following:**

1. Put the CD-ROM in your CD-ROM drive.
2. Enter the following commands, assuming your CD-ROM drive device name is `/dev/cdrom` and you are able to create a temporary directory named `cdrom`:

   ```
   mkdir /cdrom
   mount /dev/cdrom /cdrom
   ```

3. The CD-ROM contains an archive of the endpoint package. First use the `rm` command to ensure a clean temporary install directory. Then use the `tar` command to extract the archive contents from the CD-ROM:

   ```
   cd /tmp
   rm -fr temp
   tar -xvf /cdrom/endpoint/linux/pelnx_Mm.tar
   ```

4. Next, run the endpoint's installation script to install the endpoint:

   ```
   ./endpoint.install
   ```

5. You will see the license agreement, presented with the "`more`" command. Press the spacebar until the end of the agreement is displayed. You are asked whether you accept the terms and conditions of the agreement. If you do, enter "`accept_license.`"

   The endpoint installs itself in `/usr/local/ixia.` During installation you will see several status messages. When the installation is successful, you see the message "`Installation of endpoint was successful.`"

   You may instead see the following message:

   ```
   Notice! There were potential problems with migrating from
   $oldInstallPath to $installPath. Review the warnings
   displayed above for further explanation.
   ```

   If you see this message, please review the entire output from the install script for an explanation of the warnings and further instructions.

6. After the installation is complete, use the `UMOUNT` command to unmount the file system from the CD-ROM:

   ```
   umount /cdrom
   ```

   The installation script and temporary directory are not removed automatically if the installation is successful. If you need the disk space after installing the endpoint, you may delete the temporary directory and installation script.

**To remove the temp files, enter:**

```
rm -fr temp
rm endpoint.install
```

This is a good time to read the `README` file, installed with the endpoint in `/usr/local/ixia`, for the latest information about the endpoint program. Enter the `more` command to view the `README` file:

```
more /usr/local/ixia/README
```

When you've completed installation, refer to *Configuring Linux Endpoints* on page 7-11 to make sure your endpoint is ready to be used in testing and monitoring.

To install an endpoint you've downloaded from the World Wide Web, do the following:

1. First use the `rm` command to ensure a clean temporary install directory (we'll use `/tmp` in this example).
2. Save the endpoint to the `/tmp` directory.
3. Uncompress the endpoint by using the `uncompress` command:

   ```
   cd /tmp
   uncompress pelnx_Mm.tar
   tar -xvf pelnx_Mm.tar
   ```

4. From the directory where you've downloaded the endpoint, run the endpoint's installation script:

   ```
   ./endpoint.install
   ```

   The endpoint installs itself in `/usr/local/ixia.` During installation, you will see several status messages. When the installation is successful, you see the message "`Installation of endpoint was successful.`"

   You may instead see the following message:

   ```
   Notice! There were potential problems with migrating from
   $oldInstallPath to $installPath. Review the warnings
   displayed above for further explanation.
   ```

   If you see this message, please review the entire output from the install script for an explanation of the warnings and further instructions.

   The installation script and temporary directory are not removed automatically if the installation is successful. If you need the disk space after installing the endpoint, you may delete the temporary directory and installation script.

**To remove the temp files, enter:**

```
rm -fr temp
rm endpoint.install
```

This is a good time to read the `README` file, installed with the endpoint in `/usr/local/ixia`, for the latest information about the endpoint program. Enter the `more` command to view the `README` file:

```
more /usr/local/ixia/README
```

When you've completed installation, refer to *Configuring Linux Endpoints* on page 7-11 to make sure your endpoint is ready to be used in testing and monitoring.

## Removing the TAR-Based Endpoint Package (Uninstall)

Use the following command to remove the TAR-based (x86) Linux endpoint (you must be logged in as `root` to run this program):

```
/usr/local/ixia/endpoint.remove
```

If the removal is successful, you will see the following: "`Removal of endpoint was successful`." This removes the files from `/usr/local/ixia`, except for any files that were added to this directory that were not present at installation, such as the `endpoint.ini` file. This command does not delete the directory. The remove program does not automatically delete files added to the directory that you may need if you reinstall the product.

If anything goes wrong during the process of uninstalling the endpoint, a reinstalled endpoint may not run. You may need to do some extra cleanup. Check for the hidden file `/var/local/ixia/.IXIA.ENDPOINT.PID` `by using the` `ls -a` command. This file should be manually removed. Enter the following command:

```
rm /var/local/ixia/.IXIA.ENDPOINT.PID
```

## Unattended Installation for TAR-Based Linux

You can install the endpoint silently, that is, without providing any additional user input.

Complete the steps, as described in *TAR-Based Endpoint Installation for Linux* on page 7-8, through the `tar` command. Next, run the endpoint's installation, adding the "`accept_license`" parameter:

```
./endpoint.install accept_license
```

## What Happens During Installation

Here is what happens during the installation steps. The endpoint is installed into the directory `/usr/local/ixia`. A directory is created with the following contents:

- The executable programs
- The `README` file
- Various install and uninstall programs
- The directory `cmpfiles`. This directory contains files with the `.cmp` file extension. These are files containing data of different types, such as typical text or binary data. These files are used by the endpoint as data on `SEND` commands. The different data types can be used to vary the data compression performance of your network hardware and software.
- The file `endpoint.ini`

  See Chapter 2, *Endpoint Initialization File* for information about tailoring this file for individual endpoints.

The installation program stops any copy of the endpoint program currently running and starts a copy of the newly installed endpoint. You can run tests immediately, without restarting your computer.

Our software displays information on how to update your system to have the endpoint start automatically upon restarting.

No changes are made to the `PATH` environment variable of the root user.

Should you have reason to install an older endpoint, you should delete any safestore files using the following steps:

1. Stop the endpoint.
2. Delete the safestore files from the endpoint directory (or from the directory specified by the `SAFESTORE_DIRECTORY` keyword in `endpoint.ini`). Safestore files have an extension of `.q*`; you may delete them using the command:

   ```
   rm *.q*.
   ```
3. Uninstall the current endpoint.
4. Install the desired endpoint.

# Configuring Linux Endpoints

The endpoint dynamically configures its own programs, so you do not have to update the configuration files for your communications software. However, your communications software must be configured and running correctly. Take the following steps to verify that your network is ready for testing and/or monitoring:

1. Determine the network addresses of the computers for use in tests.
2. Verify the network connections.

Let’s look at TCP/IP to see how to accomplish these tasks.

## Configuration for TCP/IP

The TCP and UDP protocols use TCP/IP software for network communications. TCP/IP offers two forms of network addresses: IP addresses and domain names. An IP address is a 32-bit numeric address. It is represented in dotted notation as a set of four numbers separated by periods, such as `199.72.46.202`. The alternative, domain names are in a format that is easier to recognize and remember, such as www.ixiacom.com. To use domain names, you need either a Domain Name Server (DNS) set up in your network or an `/etc/hosts` file on each computer.

## Determining Your IP Network Address for TAR and RPM Linux

To determine the IP address of the local computer you are using, enter the following at a command prompt:

```
/sbin/ifconfig
```

## Determining Your IP Network Address for Cobalt

Access the Welcome to Cobalt page on the Cobalt Web Server and click the link to the RAQ Server Management section. The Username and Password Required dialog is shown.

After you enter the Administrator username and password, the IP address is shown on the Server Management Page.

## Testing the TCP Connection

Ping is a simple utility program, included in all TCP/IP implementations. To try out the connection from one computer to another, enter the following:

```
ping xx.xx.xx.xx -c 1
```

Replace the `x`'s with the IP address of the target computer. If Ping returns a message that says

```
1 packets transmitted, 1 packets received, 0% packet loss
```

then the Ping worked. Otherwise, there will be a delay, and you'll see

```
1 packets transmitted, 0 packets received, 100% packet
loss
```

This means that the Ping failed, and you cannot reach the target computer.

Make sure that you can run Ping successfully from the IxChariot or Qcheck Console to each computer serving as Endpoint 1, and between each pair of endpoints involved in a test, before starting your testing with TCP/IP.

## Sockets Port Number

TCP/IP applications use their network address to decide which computer to connect to in a network. They use a Sockets *port number* to decide which application program to connect to within a computer.

The TCP/IP sockets port for endpoints is **10115.** This port number is used during the initialization of a test; during the actual running of the test, other port numbers are used. If the script specifies "`port_number=AUTO`" on the `CONNECT_ACCEPT` command, additional ports are dynamically acquired from the protocol stack. Otherwise, the endpoint issuing the `CONNECT_ACCEPT` commands (usually Endpoint 2) uses the port number specified in the script.

## Autostarting the Endpoint

For the endpoint to automatically start when your computer restarts, you must update your system `rc` scripts.

If your Linux system uses `rc.local`, which is used by older Linux systems, such as Slackware, add the following line to the `rc.local` file:

```
/usr/local/ixia/endpoint 1>>/var/local/endpoint.console
2>&1 &
```

Don't forget the ampersand (`&`) at the end of the line. If this character is not included, the boot process does not continue, and you may be unable to log in at the Console.

If you have previously installed the endpoint in a `Ganymede` directory, the install script displays the following message:

```
The endpoint install directory now uses $installPath
instead of $oldInstallPath. If your rc.local referenced
```

```
$oldInstallPath, you should change it to use the new
directory.
```

If your Linux system is more recent, it probably supports System V `init rc` scripts. Red Hat software uses this type of `init rc` files. Copy `usr/local/ixia/rc2exec.lnx` to the appropriate places. For example, with Red Hat Linux 5.0, you may run these commands:

```
cp /usr/local/ixia/rc2exec.lnx /etc/rc.d/init.d/endpoint
ln -fs /etc/rc.d/init.d/endpoint /etc/rc.d/rc2.d/
S81endpoint
ln -fs /etc/rc.d/init.d/endpoint /etc/rc.d/rc3.d/
S81endpoint
ln -fs /etc/rc.d/init.d/endpoint /etc/rc.d/rc6.d/
K81endpoint
```

For Red Hat Linux 5.2 or later, or for Cobalt, the recommended commands are the following:

```
cp /usr/local/ixia/rc2exec.lnx /etc/rc.d/init.d/endpoint
/sbin/chkconfig endpoint reset
```

# Running Linux Endpoints

The following sections describe how to manually start and stop the endpoint program, and how to examine error log files if a problem occurs.

## Starting a Linux Endpoint

The endpoint program is installed so that it starts automatically each time Linux is rebooted.

- On Slackware, it sends its screen output to file `/var/adm/endpoint.console.`
- On Red Hat and Cobalt, it sends its screen output to file `/var/local/endpoint.console`.

If you want to see any error messages generated at this endpoint, enter one of the following:

**for Slackware:**

```
tail -f /var/adm/endpoint.console
```

**for Red Hat or Cobalt:**

```
tail -f /var/local/endpoint.console
```

The detailed information about the start and stop of each individual connection pair is written to file `endpoint.aud`. The contents of this file vary depending on how you've set the `SECURITY_AUDITING` keyword in your `endpoint.ini` file.

See Chapter 2, *Endpoint Initialization File* for more information about `endpoint.aud` and `SECURITY_AUDIT` settings.

Instead of automatic startup, you can choose to manually start the endpoint program at a command prompt. Ensure that you are logged in as a "root" user. To start the endpoint, enter the following:

```
/usr/local/ixia/endpoint &
```

The "&" parameter indicates to Linux that the endpoint program should run in the background. The screen output from the endpoint program is interleaved with other UNIX commands. Just press **Return** to enter more commands.

If you choose to manually start the endpoint, consider redirecting its output to the endpoint.console file. You can tell by the time stamp of the file when the endpoint program was started or stopped.

If the endpoint program is already running, you get the following message, "**CHR0183**: The endpoint program is already running. Only one copy is allowed at a time."

Use the ps command to check all running processes and make sure the endpoint is running (see the section, *How to Tell If a Linux Endpoint Is Active* on page 7-15 for more information). If you repeatedly get error message **CHR0183** but it appears that the endpoint is not running, you may need to do some extra cleanup. Check for the hidden file /usr/local/ixia/IXIA.ENDPOINT.PID by using the ls -a command. This file should be manually removed.

## Stopping a Linux Endpoint

The endpoint program has a special command-line option, -k. If you'd like to kill an endpoint program, go to a command prompt on the same computer and enter the following (you must be logged in as root to run this program):

```
/usr/local/ixia/endpoint -k
```

The -k command-line option has the purpose of killing any endpoint process running on that computer. You should see the message "Sent exit request to the running endpoint," which indicates that the endpoint program has been sent a request to stop.

If, for some reason, the request to stop is not handled correctly by the running endpoint program, you may need to use the UNIX "kill -TERM" command. Avoid using "kill -9" to stop the running endpoint program—it doesn't clean up what's been created (so you'll need to do the steps outlined in *Cleanup after Unexpected Errors* on page 7-14).

## Cleanup after Unexpected Errors

If the endpoint should fail or be killed abnormally (or encounter assertion conditions), you may also need to do additional cleanup. If the endpoint is still running, try to stop it using the command "endpoint -k". If that does not stop the endpoint, kill the endpoint using the UNIX kill command.

Then enter the following command:

```
rm /usr/local/ixia/.IXIA.ENDPOINT.PID
```

### How to Tell If a Linux Endpoint Is Active

Use traditional UNIX commands to determine if a Linux endpoint is active. At a command prompt, enter:

```
ps axf | grep endpoint
```

If the endpoint program is running, you will see output similar to this:

```
366 p0 S 0:00  \_ /usr/local/ixia/endpoint
367 p0 S 0:00  |      \_ /usr/local/ixia/endpoint
368 p0 S 0:00  |           \_/usr/local/ixia/endpoint
369 p0 S 0:00  |           \_ /usr/local/ixia/endpoint
```

### Disabling Automatic Startup

If you run a Linux system that uses `rc.local` to invoke applications, remove the invocation of `/usr/local/ixia/`endpoint from `/etc/rc.d/rc.local`.

If you use a Linux system that supports System V style `init rc` scripts, remove `/etc/rc.d/rc2.d/S81endpoint` from `/etc/rc.d/rc2.d`.

If you are using Red Hat Linux versions 5.2 or later, or Cobalt Web server, and have enabled the automatic startup through the `CHKCONFIG` utility, you can also disable the automatic startup through the `CHKCONFIG` utility. Here is the syntax to use the `CHKCONFIG` utility to disable the automatic startup:

```
/sbin/chkconfig -del endpoint
```

## Logging and Messages

While most error messages encountered on an endpoint are returned to the IxChariot or Qcheck Console, some may be logged to disk. Errors are saved in the following file:

```
/var/log/endpoint.log
```

A log file is not created until an error occurs. To view an error log, use the program named `FMTLOG`. `FMTLOG` reads from a binary log file, and writes its formatted output to `stdout`. Use the following `FMTLOG` command:

```
/usr/local/ixia/fmtlog /var/log/endpoint.log
>output_filename
```

The endpoint code does a lot of internal checking on itself. Our software captures details related to the problem in an ASCII text file:

- On Slackware, file `/var/adm/assert.err`
- On Red Hat or Cobalt, file `/var/local/assert.err`

Save a copy of the file and send it to us via email for problem determination.

### Message CHR0181

You may receive message **CHR0181** while running a test. If the error was detected at the Linux computer, it says that the endpoint program on Linux has run out of system semaphores. Each instance of Endpoint 1 requires a system semaphore. The maximum number of semaphores is not configurable on Linux, which is hard-coded to a large value (128). To avoid this problem, stop other programs that use semaphores or decrease the number of tests that use the computer as Endpoint 1.

### Increasing the Number of Concurrent Connections

Some parameters are tuned in Linux by rebuilding the Linux kernel. If you're adventurous and skilled enough, you can change the number of concurrent endpoint connections. Consult your Linux documentation for information about increasing the maximum open files allowed per process (this probably involves redefining `NR_FILES` and other macros). Alternatively, search Linux newsgroups on the Internet (using DejaNews, for example) for something like "max open files per process."

## Updates for Linux

We've found that communications software is often fragile. Its developers are constantly working to make it more robust, as the software gets used in an ever-wider set of situations.

We therefore recommend working with the very latest software for the underlying operating system and communications software.

Check the following Web sites for code and driver updates:

- www.redhat.com/
- for Cobalt, wwws.sun.com/software/linux/index.html
- www.slackware.com/
- www.calderasystems.com/
- www.linux.org/
- www.ibiblio.org/
- www.suse.com
- www.mandrake.com

We've found the following site good for ordering Linux software: www.linux-mall.com/.

# 8 Linux IA-64

The following topics explain the installation, configuration, and operation of the Performance Endpoint software for Linux IA-64. This Linux operating system is also called TurboLinux, and it runs on an Itanium processor. For 32-bit versions of Linux, download one of the Linux endpoints for Cobalt RaQ1, RaQ2, or RaQ3, RPM (Red Hat Linux), or the x86 TAR-based install.

- The Performance Endpoint software requires kernel 2.4 with pthreads support.
- We've tested the endpoint with beta versions of the Linux 64-bit operating system.

## Installing Linux IA-64 Endpoints

Here's what you need to run the endpoint program with 64-bit Linux:

- A computer capable of running Linux IA-64 well.

  This implies a 64-bit CPU such as an Intel Itanium processor.
- At least 16 MBytes of random access memory (RAM).

  We have tested on an Intel Itanium processor with 1 GByte of RAM. 16 MBytes was sufficient for 32-bit Linux platforms. The final memory requirements for 64-bit Linux are still being investigated.

  The total RAM requirement depends on RAM usage of the underlying protocol stack and the number of concurrent endpoint pairs. For tests involving over one hundred connections through a single endpoint, additional memory may be required.
- A hard disk with at least 24 MBytes of space available.
- Linux kernel 2.4 with *pthreads* support (which is at least version 2.2 of *glibc*). TCP/IP networking and corresponding networking hardware must be installed and configured, plus ELF support. We've tested with TurboLinux Frontier IA-64 Beta 3 (2001-03-07), which implements Linux kernel 2.4, on an Intel Itanium step A-3.
- An Acrobat Reader to view the PDF files.

Acrobat readers are loaded on most computers for viewing other documents, but if you don't have one, they are available at Adobe's Web Site: www.adobe.com/prodindex/acrobat/readstep.html.

## TAR-Based Endpoint Installation for Linux IA-64

First, make sure that you are logged in as a "root" user. Also, remember that all commands and parameters discussed here are case-sensitive. Use the combination of uppercase and lowercase letters shown. The following instructions explain how to install an endpoint **from a CD-ROM** and **from the World Wide Web**.

**To install the endpoint from a CD-ROM, do the following:**

1. Put the CD-ROM in your CD-ROM drive.
2. Enter the following commands, assuming your CD-ROM drive device name is `/dev/cdrom` and you are able to create a temporary directory named `cdrom`:

   ```
   mkdir /cdrom
   mount /dev/cdrom /cdrom
   ```

3. The CD-ROM contains an archive of the endpoint package. First use the `rm` command to ensure a clean temporary install directory. Then use the `tar` command to extract the archive contents from the CD-ROM:

   ```
   cd /tmp
   rm -fr temp
   tar -xvf /cdrom/endpoint/linux64/endl64r.tar
   ```

4. Next, run the endpoint's installation script to install the endpoint:

   ```
   ./endpoint.install
   ```

5. You will see the license agreement, presented with the "more" command. Press the space bar until the end of the agreement is displayed. You are asked whether you accept the terms and conditions of the agreement. If you do, enter "`accept_license`."

The endpoint installs itself in `/usr/local/Ixia`. During installation you will see several status messages. When the installation is successful, you see the message "`Installation of endpoint was successful`."

After the installation is complete, use the `UMOUNT` command to unmount the file system from the CD-ROM:

```
umount /cdrom
```

The installation script and temporary directory are not removed automatically if the installation is successful. If you need the disk space after installing the endpoint, you may delete the temporary directory and installation script.

To remove the temp files, enter:

```
rm -fr temp

rm endpoint.install
```

This is a good time to read the README file, installed with the endpoint in /usr/local/Ixia, for the latest information about the endpoint program. Enter the more command to view the README file:

```
more /usr/local/Ixia/README
```

When you've completed installation, your endpoint should be ready to be used in testing and monitoring.

**To install an endpoint you've downloaded from the World Wide Web, do the following:**

1. First use the rm command to ensure a clean temporary install directory (we'll use /tmp in this example).
2. Save the endpoint to the /tmp directory.
3. Uncompress the endpoint by using the uncompress command:

   ```
   cd /tmp
   uncompress endl64r.tar.z
   tar -xvf endl64r.tar
   ```

4. From the directory where you've downloaded the endpoint, run the endpoint's installation script:

   ```
   ./endpoint.install
   ```

   The endpoint installs itself in /usr/local/Ixia. During installation you will see several status messages. When the installation is successful, you see the message "Installation of endpoint was successful."

   The installation script and temporary directory are not removed automatically if the installation is successful. If you need the disk space after installing the endpoint, you may delete the temporary directory and installation script.

   To remove the temp files, enter:

   ```
   rm -fr temp rm endpoint.install
   ```

This is a good time to read the README file, installed with the endpoint in /usr/local/Ixia, for the latest information about the endpoint program. Enter the more command to view the README file:

```
more /usr/local/Ixia/README
```

When you've completed installation, your endpoint should be ready to be used in testing and monitoring.

## What We Do During Installation

Here is what happens during the installation steps. The endpoint is installed into the directory /usr/local/Ixia. A directory is created with the following contents:

- the executable programs;
- the README file;
- various install and uninstall programs;
- the directory cmpfiles. This directory contains files with the .cmp file extension. These are files containing data of different types, such as typical

text or binary data. These files are used by the endpoint as data on `SEND` commands. The different data types can be used to vary the data compression performance of your network hardware and software.

- the file `endpoint.ini`. See Chapter 2, *Endpoint Initialization File* for information about tailoring this file for individual endpoints.

The installation program stops any copy of the endpoint program currently running and starts a copy of the newly installed endpoint. You can run tests immediately, without restarting your computer.

Our software displays information on how to update your system to have the endpoint start automatically upon restarting.

No changes are made to the `PATH` environment variable of the root user.

Should you have reason to install an older endpoint, you should delete any safestore files taking the following steps:

1. Stop the endpoint.
2. Delete the safestore files from the endpoint directory (or from the directory specified by the `SAFESTORE_DIRECTORY` keyword in `endpoint.ini`). Safestore files have an extension of `.q*`; you may delete them using the command:

   ```
   rm *.q*.
   ```
3. Uninstall the current endpoint.
4. Install the desired endpoint.

## Removing the TAR-Based Endpoint Package (Uninstall)

Use the following command to remove the endpoint (you must be logged in as root to run this program):

```
/usr/local/Ixia/endpoint.remove
```

If the removal is successful, you will see the following: "`Removal of endpoint was successful`." This removes the files from `/usr/local/Ixia`, except for any files that were added to this directory that were not present at installation, such as the `endpoint.ini` file. This command does not delete the directory. The remove program does not automatically delete files added to the directory that you may need if you reinstall the product.

If anything goes wrong during the process of uninstalling the endpoint, a reinstalled endpoint may not run. You may need to do some extra cleanup. Check for the hidden file `/var/local/Ixia/.IXIA.ENDPOINT.PID` by using the `ls -a` command. This file should be manually removed. Enter the following command:

```
rm /var/local/Ixia/.IXIA.ENDPOINT.PID
```

## Unattended Installation for TAR-Based Linux IA-64

You can install the endpoint silently, that is, without providing any additional user input.

Complete the steps, as described above, through the `tar` command. Next, run the endpoint's installation, adding the "`accept_license`" parameter:

```
./endpoint.install accept_license
```

# Configuring Linux IA-64 Endpoints

The endpoint dynamically configures its own programs, so you do not have to update the configuration files for your communications software. However, your communications software must be configured and running correctly. Take the following steps to verify that your network is ready for testing and/or monitoring:

- Determine the network addresses of the computers for use in tests.
- Verify the network connections.

The following topics explain how to accomplish these tasks for TCP/IP.

## Configuration for TCP/IP

The TCP and UDP protocols use TCP/IP software for network communications. TCP/IP offers two forms of network addresses: IP addresses and domain names. An IP address is a 32-bit numeric address. It is represented in dotted notation as a set of four numbers separated by periods, such as 199.72.46.202. The alternative, domain names are in a format that is easier to recognize and remember, such as www.ixiacom.com. To use domain names, you need either a Domain Name Server (DNS) set up in your network or an /etc/hosts file on each computer.

## Determining Your IP Network Address

To determine the IP address of the local computer you are using, enter the following at a command prompt:

```
/sbin/ifconfig
```

## Sockets Port Number

TCP/IP applications use their network address (as described above) to decide which computer to connect to in a network. They use a Sockets port number to decide which application program to connect to within a computer.

The TCP/IP sockets port used by IxChariot endpoints is 10115. This port number is used during the initialization of a test; during the actual running of the test, other port numbers are used. If the script specifies “port_number=AUTO” on the CONNECT_ACCEPT command, additional ports are dynamically acquired from the protocol stack. Otherwise, the endpoint issuing the CONNECT_ACCEPT commands (usually Endpoint 2) uses the port number specified in the script.

## Testing the TCP Connection

Ping is a simple utility program, included in all TCP/IP implementations. To try out the connection from one computer to another, enter the following:

```
ping xx.xx.xx.xx -c 1
```

Replace the x’s with the IP address of the target computer. If Ping returns a message that says

```
1 packets transmitted, 1 packets received, 0% packet loss
```

the Ping worked. Otherwise, there will be a delay, and you’ll see

```
1 packets transmitted, 0 packets received, 100% packet
loss
```

This means that the Ping failed, and you cannot reach the target computer.

Make sure that you can run Ping successfully from the IxChariot or Ixia Qcheck Console to each computer serving as Endpoint 1, and between each pair of endpoints involved in a test, before starting your testing with TCP/IP.

# Running Linux IA-64 Endpoints

The following topics describe how to manually start and stop the endpoint program, and how to examine error log files if a problem occurs.

## Autostarting the Endpoint

For the endpoint to automatically start when your computer restarts, you must update your system `rc` scripts.

Use the following command:

```
cp /usr/local/Ixia/rc2exec.lnx /etc/rc.d/init.d/endpoint
/sbin/chkconfig endpoint reset
```

## Starting a Linux IA-64 Endpoint

The endpoint program is installed so that it starts automatically each time Linux is rebooted.

It sends its screen output to file `/var/local/endpoint.console`.

If you want to see any error messages generated at this endpoint, enter the following:

```
tail -f /var/local/endpoint.console
```

The detailed information about the start and stop of each individual connection pair is written to file `endpoint.aud`. The contents of this file vary depending on how you've set the `SECURITY_AUDITING` keyword in your `endpoint.ini` file.

See Chapter 2, *Endpoint Initialization File* for more information about `endpoint.aud` and `SECURITY_AUDIT` settings.

Instead of automatic startup, you can choose to manually start the endpoint program at a command prompt. Ensure that you are logged in as a "root" user. To start the endpoint, enter the following:

```
/usr/local/Ixia/endpoint &
```

The "&" parameter indicates to Linux that the endpoint program should run in the background. The screen output from the endpoint program is interleaved with other UNIX commands. Just press Return to enter more commands.

If you choose to manually start the endpoint, consider redirecting its output to the `endpoint.console` file. You can tell by the time stamp of the file when the endpoint program was started or stopped.

If the endpoint program is already running, you get the following message, “**CHR0183**: The endpoint program is already running. Only one copy is allowed at a time.”

Use the ps command to check all running processes and make sure the endpoint is running. If you repeatedly get error message **CHR0183**, but it appears that the endpoint is not running, you may need to do some extra cleanup. Check for the hidden file `/usr/local/Ixia/.IXIA.ENDPOINT.PID` by using the `ls -a` command. This file should be manually removed.

## Stopping a Linux IA-64 Endpoint

The endpoint program has a special command-line option, `-k`. If you’d like to kill an endpoint program, go to a command prompt on the same computer and enter the following (you must be logged in as root to run this program):

```
/usr/local/Ixia/endpoint -k
```

The `-k` command-line option has the purpose of killing any endpoint process running on that computer. You should see the message “`Sent exit request to the running endpoint`,” which indicates that the endpoint program has been sent a request to stop.

If for some reason the request to stop is not handled correctly by the running endpoint program, you may need to use the UNIX “`kill -TERM`” command. Avoid using “`kill -9`” to stop the running endpoint program—it doesn’t clean up what’s been created (so you’ll need to do the steps outlined in the following topics).

## Cleanup after Unexpected Errors

If the endpoint should fail or be killed abnormally (or encounter assertion conditions), you may also need to do additional cleanup. If the endpoint is still running, try to stop it using the command “`endpoint -k`” (described above). If that does not stop the endpoint, kill the endpoint using the UNIX `kill` command.

Then enter the following command:

```
rm /usr/local/Ixia/.IXIA.ENDPOINT.PID
```

## How to Tell If an IA-64 Linux Endpoint Is Active

Use traditional UNIX commands to determine if a Linux IA-64 endpoint is active. At a command prompt, enter:

```
ps axf | grep endpoint
```

If the endpoint program is running, you will see output similar to this:

```
11118 pts/1 S 0:00 \_ grep endpoint
7652 pts/0 S 0:00 /usr/local/Ixia/endpoint
7653 pts/0 S 0:00 \_ /usr/local/Ixia/endpoint
7654 pts/0 S 0:00 \_ /usr/local/Ixia/endpoint
7655 pts/0 S 0:00 \_ /usr/local/Ixia/endpoint
7656 pts/0 S 0:00 \_ /usr/local/Ixia/endpoint
```

## Disabling Automatic Startup

Use the following command to disable the automatic startup:

```
/sbin/chkconfig --del endpoint
```

### Increasing the Number of Concurrent Connections

Some parameters are tuned in Linux by rebuilding the Linux kernel. If you're adventurous and skilled enough, you can change the number of concurrent endpoint connections. Consult your Linux IA-64 documentation for information about increasing the maximum open files allowed per process (this probably involves redefining `NR_FILES` and other macros). Alternatively, search Linux newsgroups on the Internet (using DejaNews, for example) for something like "max open files per process."

## Logging and Messages

While most error messages encountered on an endpoint are returned to the IxChariot or Qcheck Console, some may be logged to disk. Errors are saved in the following file:

```
/var/log/endpoint.log
```

The log file is not created until an error occurs. To view an error log, use the program named `FMTLOG`. `FMTLOG` reads from a binary log file, and writes its formatted output to `stdout`. Use the following `FMTLOG` command:

```
/usr/local/Ixia/fmtlog /var/log/endpoint.log
>output_filename
```

The endpoint code performs a good deal of internal checking. Our software captures details related to the problem in an ASCII text file:

```
/var/local/assert.err.
```

Save a copy of the file and send it to us via email for problem determination.

### Message CHR0181

You may receive message CHR0181 while running a test. If the error was detected at the Linux computer, it says that the endpoint program on Linux has run out of system semaphores. Each instance of Endpoint 1 requires a system semaphore. The maximum number of semaphores cannot be configured on Linux, which is hard-coded to a large value (128). To avoid this problem, stop other programs that use semaphores or decrease the number of tests that use the computer as Endpoint 1.

## Getting the Latest Fixes and Service Updates

We've found that communications software is often fragile. Its developers are constantly working to make it more robust, as the software gets used in an ever-wider set of situations.

We therefore recommend working with the very latest communications software for the underlying operating system. Following are the best sources we've found for the Linux software used by the endpoint program.

### Updates for Linux IA-64

Check the following Web site for code and driver updates: www.linuxia64.org

# 9 Mac OS X

This chapter explains the installation, configuration, and operation of the Performance Endpoint software for Mac OS X.

We have concentrated our testing on the most popular Mac OS X operating systems. Our Mac OS X endpoints run on the following Mac OS X platforms:

## Installation

Here is what you need to run the endpoint program with Mac OS X:

- A computer capable of running Mac OS X well.
- 128 MBytes of random access memory (RAM).
- The total RAM requirement depends on RAM usage of the underlying protocol stack and the number of concurrent connection pairs. For very large tests involving hundreds of connections through a single endpoint, additional memory may be required.
- A hard disk with at least 10 MBytes of space available
- Mac OS X version 10.3 and above (Panther)

### Installation Procedure

First, ensure that you are logged in as a user with administrative privileges.

Find the Apple Mac OS X endpoint from our web site's endpoint library at: http://www.ixiacom.com/support/endpoint_library/ and double click on the endpoint file (for example *pemac_520_eb.dmg*). The endpoint will be downloaded and the installation started. You should follow the instructions to complete the installation. During the installation, you will be offered the opportunity to view the `README` file, which contains the latest information about the endpoint program.

The endpoint is installed in your *Applications* folder as a MAC appliation. To start the endpoint, browse the application folder and double click on the endpoint icon. The `README` file contains instructions on how to install the endpoint as a service.

When you've completed installation, refer to *Configuring Mac OS X Endpoints* on page 9-2 to make sure your endpoint is ready to be used in testing and monitoring.

### Removing the Endpoint (Uninstall)

Using Finder, delete the Endpoint bundle.

### What Happens During Installation

Here is what happens during the installation steps. The endpoint is installed into the *Applications* folder. A directory is created with the following contents:

- The executable programs
- The `README` file
- Various install and uninstall programs
- The directory `cmpfiles`. This directory contains files with the `.cmp` file extension. These are files containing data of different types, such as typical text or binary data. These files are used by the endpoint as data on `SEND` commands. The different data types can be used to vary the data compression performance of your network hardware and software.
- The file `endpoint.ini`

  See Chapter 2, *Endpoint Initialization File* for information about tailoring this file for individual endpoints.

If an earlier version of the endpoint is installed, you will be asked if you wish to upgrade. If you agree, the installation program stops any copy of the endpoint program currently running and starts a copy of the newly installed endpoint. You can run tests immediately, without restarting your computer.

## Configuring Mac OS X Endpoints

The endpoint dynamically configures its own programs, so you do not have to update the configuration files for your communications software. However, your communications software must be configured and running correctly. Take the following steps to verify that your network is ready for testing and/or monitoring:

1. Determine the network addresses of the computers for use in tests.
2. Verify the network connections.

Let's look at TCP/IP to see how to accomplish these tasks.

### Configuration for TCP/IP

The TCP and UDP protocols use TCP/IP software for network communications. TCP/IP offers two forms of network addresses: IP addresses and domain names. An IP address is a 32-bit numeric address. It is represented in dotted notation as a set of four numbers separated by periods, such as `192.168.46.202`. The alternative, domain names are in a format that is easier to recognize and remember, such as www.ixiacom.com. To use domain names, you need either a Domain Name Server (DNS) set up in your network or an `/etc/hosts` file on each computer.

### Determining Your IP Network Address

To determine the IP address of the local computer you are using, enter the following in a Terminal window:

```
/sbin/ifconfig
```

### Testing the TCP Connection

Ping is a simple utility program, included in all TCP/IP implementations. To try out the connection from one computer to another, enter the following:

```
ping xx.xx.xx.xx -c 1
```

Replace the `x`'s with the IP address of the target computer. If Ping returns a message that says

```
1 packets transmitted, 1 packets received, 0% packet loss
```

then the Ping worked. Otherwise, there will be a delay, and you'll see

```
1 packets transmitted, 0 packets received, 100% packet loss
```

This means that the Ping failed, and you cannot reach the target computer.

Make sure that you can run Ping successfully from the IxChariot or Qcheck Console to each computer serving as Endpoint 1, and between each pair of endpoints involved in a test, before starting your testing with TCP/IP.

### Sockets Port Number

TCP/IP applications use their network address to decide which computer to connect to in a network. They use a TCP or UDP *port number* to decide which application program to connect to within a computer.

The port number for endpoints is **10115.** This port number is used during the initialization of a test; during the actual running of the test, other port numbers are used. If the script specifies "`port_number=AUTO`" on the `CONNECT_ACCEPT` command, additional ports are dynamically acquired from the protocol stack. Otherwise, the endpoint issuing the `CONNECT_ACCEPT` commands (usually Endpoint 2) uses the port number specified in the script.

## Running Mac OS X Endpoints

The following sections describe how to manually start and stop the endpoint program, and how to examine error log files if a problem occurs.

The endpoint icon, located in the *Applications* folder, may be used to manually start the Mac OS X endpoint. Alternatively, you may set up the endpoint to automatically start with your computer by dragging and dropping the icon into the `/library/StartupItems` folder using Finder.

If you want to see any error messages generated by the endpoint, use the *fmtlog* command to view the *Endpoint.log* file located in */private/var/log*.

The detailed information about the start and stop of each individual connection pair is written to file `endpoint.aud`. The contents of this file vary depending on how you've set the `SECURITY_AUDITING` keyword in your `endpoint.ini` file.

See Chapter 2, *Endpoint Initialization File* for more information about `endpoint.aud` and `SECURITY_AUDIT` settings.

If the endpoint program is already running, you get the following message, “**CHR0183**: The endpoint program is already running. Only one copy is allowed at a time.”

Use the `ps` command to check all running processes and make sure the endpoint is running (see the section, *How to Tell If a Mac OS X Endpoint Is Active* on page 9-4 for more information). If you repeatedly get error message **CHR0183** but it appears that the endpoint is not running, you may need to do some extra cleanup. Check for the file `/private/var/log/.ENDPOINT.PID` by using Finder. This file should be manually removed.

## Stopping a Mac OS X Endpoint

If the endpoint was started manually, it may be terminated by selecting `Quit` from the desktop icon.

If the endpoint was started automatically, then it may be terminated by using the `SystemStarter` command:

```
sudo SystemStarter Stop Endpoint
```

A password may be required.

If the endpoint does not stop, then you will need to use

```
kill -9 <pid>
```

to stop the running endpoint program. See *How to Tell If a Mac OS X Endpoint Is Active* below for instructions on using the `ps` command and determining the process id (pid) of the endpoint. With the “-9” argument, the endpoint doesn’t clean up what’s been created (so you’ll need to do the steps outlined in *Cleanup after Unexpected Errors* on page 9-4).

## Cleanup after Unexpected Errors

If the endpoint should fail or be killed abnormally (or encounter assertion conditions), you may also need to do additional cleanup. Enter the following command:

```
rm -f /usr/local/ixia/IXIA.ENDPOINT.PID
```

## How to Tell If a Mac OS X Endpoint Is Active

Use traditional UNIX commands to determine if a Mac OS X endpoint is active. At a command prompt, enter:

```
ps ax | grep endpoint
```

If the endpoint program is running, you will see output similar to this:

```
855 ?? S 3:19:90 ./endpoint

2846 std R+ 0:00:00 grep endpoint
```

### Disabling Automatic Startup

If you wish to disable the Mac OS X from running as a service, then stop it as described above and remove the endpoint folder from the `/Library/StartupItems` folder.

## Logging and Messages

While most error messages encountered on an endpoint are returned to the IxChariot or Qcheck Console, some may be logged to disk. Errors are saved in the following file:

- /private/var/log/endpoint.log

To view an error log, use the IxChariot Console's Tool menu, View Error Log choice.

The endpoint code does a lot of internal checking on itself. Our software captures details related to the problem in an ASCII text file:

- /private/var/log/`assert.err`

Save a copy of the file and send it to us via email for problem determination.

## Updates for Mac OS X

We've found that communications software is often fragile. Its developers are constantly working to make it more robust, as the software gets used in an ever-wider set of situations.

We therefore recommend working with the very latest software for the underlying operating system and communications software.

Use the Software Update program that is included with Mac OS to keep your Mac software up to date.

10

# Microsoft Windows CE

This chapter explains the installation, configuration, and operation of the Performance Endpoint software for Microsoft Windows CE.

## Installation Requirements

Here’s what you need to run the endpoint software with Microsoft Windows CE:

- Processor:
  - Intel Strong Arm® Processor
  - Intel XScale® Processor
  - Intel x86 compatible processor
- Operating System Version:
  - Windows CE 4.2
- 64 MB of random access memory (RAM).

Microsoft Windows Mobile software for Pocket PC versions 2002 and 2003 are supported. Microsoft explains that their Pocket PC 2003 is based on a new operating system, Microsoft Windows CE .NET 4.1 or 4.2.

## Network Protocol Stacks

We recommend that you configure your networking software–and make sure that it is working correctly–before installing our endpoint software.

We suggest that you use the built-in network protocol stack. In addition, you may need to purchase and configure a wireless or wired adapter.

# Endpoint Installation for Windows CE

## Installation on Intel Strong Arm and XScale Processors

**To install the endpoint:**

The following installation instructions assume that the Windows CE device to be tested is already synched to your desktop computer:

1. From your desktop PC, navigate to the Windows CE endpoint at www.ixiacom.com/support/ixchariot.
2. Download the Windows CE endpoint package to your desktop PC.
3. Copy the file `pewcearm_Mm.exe` to the Windows Clipboard using the Windows Explorer. *Mm* is the endpoint release number; for example, 520 for release 5.20.
4. Paste the file to the following directory:

   ```
   [Mobile Device]\My Pocket PC\Windows\Start Menu
   ```

5. On your Windows CE device, tap **Start** > **pewcearm_*Mm*.exe**. *Mm* is the endpoint release number; for example, 520 for release 5.20.

## Installation on Intel x86 Processors

**To install the endpoint:**

The following installation instructions assume that the Windows CE device to be tested is already synched to your desktop computer:

1. From your desktop PC, navigate to the Windows CE endpoint at www.ixiacom.com/support/ixchariot.
2. Download the Windows CE endpoint package to your desktop PC.
3. Copy the file `pewcex86_Mm.exe` to the Windows Clipboard using the Windows Explorer. *Mm* is the endpoint release number; for example, 520 for release 5.20.
4. Paste the file to the following directory:

   ```
   [Mobile Device]\My Pocket PC\Windows\Start Menu
   ```

5. On your Windows CE device, tap **Start** > **pewcex86_*Mm*.exe**. *Mm* is the endpoint release number; for example, 520 for release 5.20.

**Alternate Installation**

Since the Windows CE for the x86 architecture is similar to standard Windows, the **pewcex86_*Mm*.exe** executable may be copied from another computer via a network share or FTP. *Mm* is the endpoint release number; for example, 520 for release 5.20. It may be installed in any location on the Windows CE drive and executed from that location.

**Note**: If the Start menu on the Pocket PC where you're installing the endpoint has already reached the maximum number of icons it can display, the endpoint software is automatically copied to the directory [Mobile Device]\My Pocket PC\Windows\Start Menu\Programs.

See the following HP business support document for more information: http://h20000.www2.hp.com/bizsupport/TechSupport/Document.jsp?locale=en_US&taskId=115&prodSeriesId=306693&prodTypeId=215348&objectID=PSD_MH030919_CW01.

# Removing the Endpoint Package (Uninstall)

The following installation instructions assume that the Windows CE pocket PC or device to be tested is already synched to your desktop computer:

Delete `pewcearm_`*`Mm`*`.exe` (for the Strong Arm version) or `pewcex86_`*`Mm`*`.exe` (for the x86 based version) from the following directory on your desktop PC:

```
[Mobile Device]\My Pocket PC\Windows\Start Menu
```

On the x86 version, if the `pewcex86_`*`Mm`*`.exe` executable was installed in an alternate location, find and delete that file. *Mm* is the endpoint release number; for example, 520 for release 5.20.

# Windows CE Configuration for TCP/IP

The RTP, TCP, and UDP protocols use TCP/IP software for network communications. TCP/IP offers two forms of network addresses: IP addresses and domain names. An IP address is a 32-bit numeric address. It is represented in dotted notation as a set of four numbers separated by periods, such as `199.72.46.202`. The alternative, domain name, is in a format that is easier to recognize and remember, such as www.ixiacom.com. To use domain names, you need a Domain Name Server (DNS) set up in your network.

## Determining Your IP Network Address

On your Windows CE device, tap **Start** > **Settings** > **Connections** and tap the **Network Adapters** icon. Select an adapter and then tap **Properties**.

Look at your adapter configuration. If you are using DHCP, your adapter configuration may not show your address. In that case, contact your network administrator to find out which IP address the DHCP server has assigned to the adapter.

## Testing the TCP Connection

Ping is a simple utility program, included in all TCP/IP implementations. To check the connection from one computer to another, enter the following at an MS-DOS command prompt:

```
ping xxx.xxx.xxx.xxx
```

Replace the xxx's with the IP address of the target computer. If Ping returns a message that says "`Reply from xxx.xxx.xxx.xxx ...`," the Ping worked. If the message says "Request timed out," the Ping failed, and you have a configuration problem.

Make sure that you can run Ping successfully from the IxChariot or Ixia Qcheck Console to each computer serving as Endpoint 1, and between each pair of endpoints involved in a test, before starting your testing with TCP/IP.

### Sockets Port Number

TCP/IP applications use their network address (as described above) to decide which computer to connect to in a network. They use a Sockets *port number* to decide which application program to connect to within a computer.

The TCP/IP sockets port used by IxChariot endpoints is **10115**. This port number is used during the initialization of a test. During the actual running of the test, other port numbers are used. If the script specifies "`port_number=AUTO`" on the `CONNECT_ACCEPT` command, additional ports are dynamically acquired from the protocol stack. Otherwise, the endpoint issuing the `CONNECT_ACCEPT` commands (usually Endpoint 2) uses the port number specified in the script.

## Running Windows CE Endpoints

The following sections describe some of the limitations associated with the Windows CE operating system, and how to start and stop an endpoint. A final section describes how the endpoint handles error messages.

### Limitations of the Windows CE Endpoint

The endpoint for Windows CE does not support the following IxChariot test parameters:

- One-way delay measurements for voice over IP testing.
- Disabling the UDP checksum.
- DiffServ QoS templates.
- Application scripts with `.cmp` data files as the datatype.
- No scripts, such as the Internet scripts, that use `.cmp` files by default will run to this endpoint. As a work-around, edit the scripts to use `NOCOMPRESS` as the `send_datatype` instead of a `.cmp` file.
- CPU utilization measurements.
- Traceroute testing.

In addition, by default Windows CE will not support a UDP IxChariot test with a datagram window of more than two datagrams. The test will time out with error message **CHR0216**. This problem will only occur if you adjust the `send_buffer_size` or Window Size parameter to include more than two UDP datagrams in a window.

This Windows CE limitation has been documented in the Microsoft Knowledge Base article Q290206. The article explains that the default internal UDP buffer queue size on Windows CE is 2. To support applications that deliver more than 2 datagrams in a very short time, the default limit can be raised to a value between 2 and 10 hex. For example, change the following Registry setting:

```
[HKEY_LOCAL_MACHINE\Comm\Afd]
DgramBuffer=dword:8
```

The device must be reset for this parameter to take effect.

## Intel Strong Arm and XScale Processor Based Operation

### Starting a Windows CE Endpoint

On your Windows CE device, tap **Start** > **pewcearm_*Mm*.exe**. *Mm* is the endpoint release number; for example, 520 for release 5.20.

### Stopping a Windows CE Endpoint

To stop the endpoint program, use the following menu path on your Windows CE device:

1. Tap **Start** > **Settings** > **System** > **Memory** > **Running Programs**.
2. Select **Performance Endpoint** and then tap **Stop**.

## Intel x86 Processor Based Operation

### Starting a Windows CE Endpoint

On your Windows CE device, tap **Start** > **pewcex86_*Mm*.exe**. If the executable was installed in an alternate location, find and tap on the `pewcex86_Mm.exe` executable. *Mm* is the endpoint release number; for example, 520 for release 5.20.

### Stopping a Windows CE Endpoint

To stop the endpoint program, use the following menu path on your Windows CE device:

1. Click on the **X** at the top right corner of the application, or use the **File** > **Exit** menu choice.

**NOTE**: One some versions of Windows CE, such as the iPac, the Ixia endpoint application is surrounded by an outer window. Make sure to press the **X** on the inner window to stop the endpoint.

## Checking the Endpoint Version

The current version should be displayed on the endpoint main window.

## Error Messages

All error messages encountered by an endpoint are returned to the IxChariot or Qcheck Console, but are not logged to disk.

# 11 Microsoft Windows NT/ 2000/2003/XP

This chapter explains the installation, configuration, and operation of the Performance Endpoint software for Microsoft Windows NT, Windows 2000, and Windows XP, and Windows Server 2003. (The endpoint has not been renamed to reflect its support for Windows Server 2003.) Separate versions of the endpoint operate on the “x86” and “Alpha” versions of Windows NT. A separate version of the endpoint is also available for the 64-bit version of Windows XP; see our Web site for more information.

- x86 computers are commonly known as “PCs”; they contain CPUs made by Intel, AMD, Cyrix, and others.
- Alpha computers contain CPUs made by Compaq Corporation (formerly Digital Equipment Corporation, or DEC).

This endpoint now supports IxChariot testing with the Microsoft Windows XP Tablet PC Edition.

The Performance Endpoint for the Windows 98 operating system has been archived at version 4.3. It will not support new features in recent releases of Ixia products; however, it is still available on the Ixia Web site at www.ixiacom.com/support/ixchariot.

> NOTE: in the following discussion, the name of the HP endpoint file is pew32_*Mm*.tar, where *Mm* is the major and minor IxChariot version number; for example *520* for IxChariot release 5.20

# Installation Requirements for Windows Endpoints

Here’s what you need to run the endpoint program with Windows NT, Windows 2000, or Windows XP:

- A computer capable of running Windows NT, Windows 2000, Windows XP, or Windows Server 2003 well.

  For x86 computers, this implies a CPU such as an Intel 80486, a member of the Pentium family, or equivalent. A Pentium or better is recommended.

  For Alpha computers, any system seems to give good performance.
- 32 MBytes of random access memory (RAM).
- The total RAM requirement depends on the RAM usage of the underlying protocol stack and the number of concurrent connection pairs. For very large tests involving hundreds of connections through a single endpoint, additional memory may be required.
- A hard disk with at least 8 MBytes of space available.
- Microsoft Windows NT version 4.0, Windows 2000, or Windows XP, or Windows 2003.

  Both the Workstation and Server of these operating systems are supported.
  - for IP Multicast: Windows NT 4.0 with Service Pack 3 (or later), Windows 2000, or Windows XP is required.
  - for IP QoS: Windows 2000 requires the QoS Packet Scheduler.
- The latest service packs for Windows NT. On Windows NT with Service Pack 3, Microsoft Internet Explorer version 4.0 and higher is **required**. Use Service Pack 6a instead. Service Pack 6 is not supported.

See the `README` file for this endpoint to see the latest Microsoft service packs with which we’ve tested.

You also need compatible network protocol software:

- **for APPC, one of the following**

  Three APPC stacks for Windows NT or Windows 2000 are supported by the endpoint.
  - IBM Personal Communications version 4.3 (PCOMM for Windows NT, also called networks or SecureWay): runs on x86 computers where its communications APIs are installed.
  - IBM Communications Server version 6.0 (for Windows NT and Windows 2000): runs only on the server computer of Communications Server’s “split stack” model.
  - Microsoft Windows SNA Server for x86: runs on either a client or the server computer of SNA Server’s “split stack” model. We recommend version 4.0 of SNA Server for Windows NT 4.0, with the latest service packs.

  - SNA Server 4.0 requires a fix to use fully qualified LU names. See the Ixia Support Web site to download the fix, which we obtained from Microsoft.

- **for IPX and SPX**

  IPX and SPX software is provided as part of the network support in the Windows NT, Windows 2000, Windows XP, and Windows Server 2003 operating systems.

  Microsoft improved their IPX/SPX support for Windows NT, Windows 2000, and Windows XP, using "SPX II." SPX II is also present on Novell NetWare 4.x (or later). SPX II allows a window size greater than 1, and buffer sizes up to the size the underlying transport supports.

  The SPX protocol supplied by Microsoft in Windows NT 4.0 is subject to slowdowns when running to itself, that is, with loopback.

  Our software does not support connections between Windows NT and OS/2, using IPX or SPX.

- **for RTP, TCP, and UDP**

  TCP/IP software is provided as part of the network support with Windows NT, Windows 2000, Windows XP, and Windows Server 2003.

  Microsoft's Service Pack 3 for Windows NT 4.0 fixes several TCP/IP bugs; Service Pack 3 (or later) is strongly recommended for users of Windows NT 4.0. Service Pack 3 (or later) is required for IP Multicast testing.

  Quality of Service (QoS) support for TCP/IP is part of Microsoft Windows 2000, Windows XP, and Windows Server 2003. On Windows NT, Type of Service is available for UDP and RTP only. See the *User Guide* for IxChariot for more information.

There's one version of the endpoint for Windows NT 4.0 on x86 computers, and a separate version for Windows NT on Alpha. This software supports a range of underlying functions, which vary by operating system level and service pack. This functional support is summarized in the table we included in *Endpoint Capabilities* on page 1-4.We recommend that you get up-to-date with the latest Windows NT, Windows 2000, and Windows XP service levels. *Getting the Latest Fixes and Service Updates* on page 11-17 discusses where to get the latest software upgrades.

# Endpoint Installation

We recommend configuring your networking software -- and ensuring that it is working correctly -- before installing our software. See the Help for your networking software, and see *Configuring Windows Endpoints* on page 11-9 for more assistance.

> **Note**: Before installing the endpoint on Windows 2000, plan to close any other network applications. During the endpoint installation, Windows 2000 recycles the protocol stack, causing some client applications to lose connectivity to their servers. Some of these applications don't retry their connectivity before exiting and must be restarted.

The endpoint for Windows NT, Windows 2000, or Windows XP, and Windows Server 2003 is installed and runs as a service. Only a user ID with Administrator authority is permitted to install services. To successfully install the endpoint, you must be logged in with Administrator authority. The permissions of the directory where the endpoint is installed must also be set to allow the `SYSTEM` (the operating system) full control access. Be sure to give the System "Full Control" permission on all files in the `C:\Program Files\Ixia\Endpoint` directory or the directory where you've installed the endpoint, plus any relevant subdirectories, if any.

The security implementation in Windows Server 2003 differs noticeably from that in earlier versions of Windows. Before you install the endpoint on Windows Server 2003, make sure your user account is running in *Install* mode and not in *Execute* mode. To change the mode so that you have the necessary installation privileges, run the following at a command prompt:

```
change user /install
```

The installation on Windows Server 2003 will fail with the message "The InstallShield-generated file that allows uninstallation is missing" if you're trying to install from the wrong mode.

Following are directions for installing the endpoint **from a CD-ROM** and **from the World Wide Web**:

**To install the endpoint from a CD-ROM, do the following:**

1. Put the CD-ROM in your CD-ROM drive.
2. Go to a command prompt.
   - For x86 computers, go to the directory `WIN32` and enter the following:

     ```
     [drive:]\Endpoint\Win32\pew32_Mm.exe
     ```

   - For Alpha computers, go to the directory `archive\winnta` and enter the following:

     ```
     [drive:]\Endpoint\archive\Winnta\setup.exe
     ```

3. Select the directory where the endpoint will be installed. We recommend installing it on a local hard disk of the computer you're using. If you install on a LAN drive, the additional network traffic may influence your performance results. The default directory is `C:\Program Files\Ixia\Endpoint`, on your boot drive.
4. If you have a previous installation of the endpoint, you will be asked if you want it removed. If you select "**Yes**," the previous installation is removed, and the new installation continues. If you select "**No**," the install program exits with no changes to your existing installation because a new version cannot be added until the old version is removed. The installation program adds the endpoint program as a service.
5. The next dialog box contains three checkboxes.
   - Check the first check box to install pre-built data files. We recommend you leave this box checked. You can save a small amount of disk space by not installing the files used for compression testing -- but the defaults in many

application scripts specify these files. If these `CMP` files are not installed, many application scripts cannot be used in tests until they are modified.

- Check the second check box to specify an LU alias for Microsoft SNA.

  If you plan to test with APPC using Microsoft SNA Server on this endpoint, check this box. The next screen prompts you to enter the APPC LU alias for this computer. If you need to specify an LU alias for SNA Server later, you can use our software's `SETALIAS` program. See the Support area of our Web site.

  If you enter an APPC LU Alias, it must be defined already at the SNA Server, and must be unique in the network. The LU Alias you enter won't take effect until after the computer is restarted (or the SnaBase service is stopped and restarted).

- Check the third check box to start the endpoint on installation. If you leave the box unchecked, the endpoint starts when you restart the computer. No window is shown while the endpoint is running because it runs as a service.

  A Windows NT, Windows 2000, Windows XP, or Windows Server 2003 service is controlled from the Services dialog inside the Control Panel. If you want to restart a service without restarting Windows NT, Windows 2000, or Windows XP, use the Services dialog box. For example, to start SnaBase, go to the Services dialog box, select the SnaBase line, and click **Start** (or **Play**). The status changes to "started" when the service is successfully started.

  You can also manually start the endpoint after installation. See *Starting the Endpoint* on page 11-13 for instructions.

6. Finally, you are asked whether you want to install application monitoring support. This option is ***not*** recommended. The `README` file contains a list of significant operating restrictions. Click the radio button next to the option you've selected. Click **Next** to accept the default option, which does not install the extra support. The endpoint installation copies the necessary files to your hard disk.

> **Note**: Application monitoring support should *not* be installed on a server unless it has been thoroughly tested beforehand. Interaction problems may arise; proceed with caution. Consult the endpoint `README` file for more information.

The installation is now complete; you can remove the CD-ROM from its drive.

To prevent the endpoint from running automatically on startup, see the section titled *Disabling Automatic Startup* on page 11-15.

When you've completed installation, refer to *Configuring Windows Endpoints* on page 11-9 to make sure your endpoint is ready for testing and monitoring.

**To install an endpoint you've downloaded from the World Wide Web, do the following:**

1. Save the `pew32_Mm.exe` file to a local directory.
2. Use the Windows Explorer to navigate to the file and double-click to start the installation.
3. The first screen after the Setup dialog box lets you select the directory where the endpoint will be installed. We recommend installing it on a local hard disk of the computer you're using. If you install on a LAN drive, the additional network traffic may influence your performance results. The default directory is `C:\Program Files\Ixia\Endpoint`, on your boot drive.
4. If you have a previous installation of the endpoint, you will be asked if you want it removed. If you select "**Yes**," the previous installation is removed, and the new installation continues. If you select "**No**," the install program exits with no changes to your existing installation because a new version cannot be added until the old version is removed. It then adds Endpoint (the endpoint program) as a service.
5. The next dialog contains three check boxes.
   - Check the first check box to install pre-built data files. We recommend you leave this box checked. You can save a small amount of disk space by not installing the files used for compression testing -- but the defaults in many application scripts specify these files. If these `.CMP` files are not installed, many application scripts cannot be used in tests until they are modified.
   - Check the second check box to specify an LU alias for Microsoft SNA.

     If you plan to test with APPC using Microsoft SNA Server on this endpoint, check this box. The next screen prompts you to enter the APPC LU alias for this computer. If you need to specify an LU alias for SNA Server later, you can use our software's `SETALIAS` program.

     If you enter an APPC LU Alias, it must be defined already at the SNA Server, and must be unique in the network. The LU Alias you enter won't take effect until after the computer is restarted (or the SnaBase service is stopped and restarted).
   - Check the third check box to start the endpoint on installation. If you leave the box cleared, the endpoint starts when you restart the computer. No window is shown while the endpoint is running, since it runs as a service.

     A Windows NT, Windows 2000, Windows XP, or Windows Server 2003 service is controlled from the Services dialog inside the Control Panel. If you want to restart a service without restarting Windows NT, Windows 2000, or Windows XP, use the Services dialog box. For example, to start SnaBase, go to the Services dialog box, select the SnaBase line, and click **Start** (or **Play**). The status changes to "started" when the service is successfully started.

     You can also manually start the endpoint after installation. See *Starting the Endpoint* on page 11-13 for instructions.
6. Finally, you are asked whether you want to install application monitoring support. This option is ***not*** recommended. The `README` file contains a list of significant operating restrictions. Click the radio button next to the option you've selected. Click **Next** to accept the default option, which does not

install the extra support. The endpoint installation copies the necessary files to your hard disk.

**Note**: Application monitoring support should *not* be installed on a server unless it has been thoroughly tested beforehand. Interaction problems may arise; proceed with caution. Consult the endpoint `README` file for more information.

To prevent the endpoint from running automatically on startup, see the section titled *Disabling Automatic Startup* on page 11-15. If you want to restore the setting later, you must do so manually.

When you've completed installation, refer to *Configuring Windows Endpoints* on page 11-9 to make sure your endpoint is ready for testing and monitoring.

## What Happens During Installation

Here's what happens during the installation steps. Let's say you install the endpoint into the directory `C:\Program Files\Ixia\Endpoint`. A directory is created with the following contents:

- The executable programs
- The `README` file
- The directory `Cmpfiles`. This directory contains files with the `.CMP` file extension. These are files containing data of different types, such as typical text or binary data. These files are used by the endpoint as data on `SEND` commands. The different data types can be used to vary the data compression performance of your network hardware and software.
- The file `endpoint.ini`
- See Chapter 2, *Endpoint Initialization File* for information about tailoring this file for individual endpoints.

The endpoint is installed as a service, which means there's nothing visible while it's running. During installation, the endpoint is configured to automatically start when the system reboots. A service can be controlled from the Services dialog box inside the Control Panel; this process is described in *Running Windows Endpoints* on page 11-13.

Should you have reason to install an older endpoint, you should delete any safestore files, taking the following steps:

1. Stop the endpoint.
2. Delete the safestore files from the endpoint directory (or from the directory specified by the `SAFESTORE_DIRECTORY` keyword in `endpoint.ini`). Safestore files have an extension of `.q*`; you may delete them using the command delete `*.q*`.
3. Uninstall the current endpoint.
4. Install the desired endpoint.

## Unattended Installation

Unattended installation (also called silent installation) is available for the endpoints for Windows. You install an endpoint once, by hand, while the install facility saves your input in an answer file. You can then install that same endpoint silently on other computers, that is, without providing input other than the answer file.

First, run `pew32_Mm.exe`. An answer file called `update.iss` is created in the `\Updates` subdirectory of the directory where you installed the endpoint.

To perform a silent installation, specify the "`-s`" option on `SETUP`. Make sure the answers documented in the answer file `update.iss` are appropriate for the silent installation. If the `update.iss` file is not in the same directory as `setup.exe`, then specify the path and filename with the "`-f1`" option. For example, here's how to install using the `update.iss` file in the `\Program Files\Ixia\Endpoint` directory on our `n:` LAN drive:

```
SETUP -s -f1n:\Program Files\Ixia\Endpoint\update.iss
```

If you don't specify the path and filename with `-f1, the default filename is setup.iss.` Don't mix the `.iss` files among different Windows operating systems because their endpoint installations require slightly different input.

It's common to use unattended install from a LAN drive. Be sure you've copied all of the files for each type of endpoint into a single directory (rather than into separate diskette images), and you've created your initial `update.iss` file from that directory. Unattended install does not keep track of diskette label information, and will need user input if you install from separate disk images. You probably don't want your unattended install to ask you for `n:\disk1\`, `n:\disk2\`, and so on.

If you're planning to use APPC with the endpoint for Windows NT/2000/XP, do NOT enter an LU alias in your initial installation that would be propagated to all the other Windows computers. All the APPC LU aliases MUST be unique (like IP addresses or MAC addresses). So when doing the initial installation, leave the check box asking about LU alias unchecked. Go back later and create LU aliases using the `SETALIAS` program.

## Installing the Windows Endpoint with SMS

See Chapter 4, *Distributing Endpoints using SMS* for information on automatically installing (and uninstalling) endpoints, using Microsoft's Systems Management Server (SMS).

## Removing the Endpoint Package (Uninstall)

**To remove the endpoint package from your hard disk, follow these steps:**

1. On the Start menu, click **Settings** and then **Control Panel**.
2. Click on **Add/Remove Programs**. The Add/Remove Programs Properties dialog box is shown.
3. Highlight **Ixia Endpoint for Windows** and press **Add/Remove**. The uninstallation program begins. After the program is completed, the endpoint should be uninstalled.

### Removing the Endpoint Manually

If the uninstallation program is unable to uninstall the endpoint, you will need to manually uninstall it. For detailed instructions on manually removing the endpoints, see the Performance Endpoints FAQ page in the Knowledge Base on our Web site at www.ixiacom.com/support/chariot/knowledge_base.php.

## Configuring Windows Endpoints

The endpoint program uses the network application programming interfaces, such as Sockets and APPC, for all of its communications. The endpoint dynamically configures its own programs, so you do not have to update the configuration files for your communications software. However, your communications software must be configured and running correctly. The following steps guide you through this verification process.

1. Determine the network addresses of the computers to be used in tests.
2. Select a service quality.
3. Verify the network connections.

The following sections describe how to accomplish these steps for Windows NT, Windows 2000, or Windows XP:

- *Windows NT and Windows 2000 Configuration for APPC*
- *Windows NT, Windows 2000, or Windows XP Configuration for IPX and SPX* on page 11-10
- *Windows NT, Windows 2000, or Windows XP Configuration for TCP/IP* on page 11-12

### Windows NT and Windows 2000 Configuration for APPC

APPC has not been tested on Windows XP and may not be supported. On Windows NT or Windows 2000, the endpoint supports three APPC stacks:

- IBM Personal Communications AS/400 version 4.3 (for Windows NT)
- IBM Personal Communications version 5.0 (for Windows 2000)
- IBM Communications Server version 6.0 (for Windows NT and Windows 2000)
- Microsoft Windows SNA Server version 4.0 (for Windows NT 4.0)

IBM has created a thorough (but aging) “redbook” to assist in setting up APPC across a variety of platforms. This guide is called the *MultiPlatform APPC Configuration Guide* and can be viewed or downloaded from the Web at: www.redbooks.ibm.com/pubs/pdfs/redbooks/gg244485.pdf.

### APPC TP Name

APPC applications use an *LU name* to decide which computer to connect to in a network. They use a *TP name* to decide which application program to connect to within a computer.

Our software uses the string GANYMEDE.CHARIOT.ENDPOINT as its TP name. This TP name is used when communicating with endpoints via an APPC connection.

## Testing the APPC Connection

Now that you know the LUs and modes you are using, you can run a quick check using a program named APING.

APING is a small application packaged with most APPC stacks. It is similar to Ping in TCP/IP; it is an echo program that sends a block of data to another computer. That computer receives the data and sends it back. APING verifies that APPC is correctly installed at a pair of computers, that they are connected to the network, and that it is possible to get an APPC session using the mode you have selected.

To run APING, go to the IBM Communications Server or IBM Personal Communications Programs folder:

1. Select **Utilities**.
2. Select **APPC** and **CPI C Utilities**.
3. Select **Check Connection APING**.

Enter the LU name of the partner you want to connect with. You might want to try entering your own local LU name the first time, just to see how it works. Click **Start**, or click **Start** on the Action menu. It uses the mode name #INTER, by default. (In our software, the mode name is known as the "*service quality*.") If APING works, APING shows a table of timing information. This endpoint should be ready for APPC testing. Continue testing connections to the other endpoints you will use.

If you get any other APPC return code, you have a configuration problem somewhere. You should correct this before starting to run our software.

Make sure that you can run APING successfully from the IxChariot or Qcheck Console to each computer serving as Endpoint 1, and between each pair of endpoints involved in a test, before starting your testing with APPC.

If you create a connection pair with the same Windows NT or Windows 2000 computer running APPC configured as both Endpoint 1 and Endpoint 2 (that is, a loopback connection), the endpoint returns message **CHR0182** to indicate an error.

## Windows NT, Windows 2000, or Windows XP Configuration for IPX and SPX

To use the IPX or SPX protocol in tests, IPX addresses must be supplied as the network address when adding a connection pair. IPX addresses consist of a 4-byte network number (8 hexadecimal digits) followed by a 6-byte node ID (12 hex digits). A colon separates the network number and node ID. The 6-byte node ID (also known as the *device number*) is usually the same as the MAC address of the LAN adapter you're using.

In IxChariot, it's tedious to enter IPX addresses when adding new connection pairs. When using the IPX or SPX protocol in your tests, our software can maintain an easy-to-remember alias in the Edit Pair dialog. You can set up the mapping once, and use the alias names ever after. The underlying file, named `spxdir.dat`, is like the `HOSTS` file used in TCP/IP, or the LU alias definitions offered with APPC.

For Win32 operating systems, endpoints make WinSock version 1.1 Sockets-compatible calls when using the IPX or SPX network protocol.

## Determining Your IPX Network Address

To determine a Windows NT, Windows 2000, or Windows XP computer's local IPX address, enter the following at a command prompt:

```
IPXROUTE CONFIG
```

If your IPX software support is configured correctly, your output will look like the following (this output is taken from Windows NT 4.0):

```
NWLink IPX Routing and Source Routing Control Program v2.00
net 1: network number 00000002, frame type 802.2, device AMDPCN1
(0207011a3082)
```

The 8-digit network number is shown first; here, it's `00000002`. The 12-digit node ID is shown in parentheses at the end; here it's `0207011a3082`, which is our Ethernet MAC address. Thus, the IPX address to be used in tests is `00000002:0207011a3082`.

Another method: if you already know the IP address of a computer -- and thus can Ping to that computer -- it's easy find its MAC address. First, Ping to the target computer from a computer on the same network segment, using its IP address. Then, enter the following command:

```
arp -a
```

A list of recently cached IP addresses is shown, along with their MAC addresses if they are LAN-attached. The `arp` command only reports the physical address of computers it can reach without crossing a router. It also won't give you the physical address of the local computer.

## Stopping Connections Doing SPX Loopback

An IxChariot Console user can observe that stopping can take between 20 and 50 seconds when running connections using SPX on Windows NT, doing loopback (that is, both endpoints have the same address). If the endpoint is on a `Receive` call, the protocol stack can pause for almost a minute before returning.

## Windows NT, Windows 2000, or Windows XP Configuration for TCP/IP

The RTP, TCP, and UDP protocols use TCP/IP software for network communications. TCP/IP offers two forms of network addresses: IP addresses and domain names. An IP address is a 32-bit numeric address. It is represented in dotted notation as a set of four numbers separated by periods, such as 199.72.46.202. An alternative, domain names are in a format that is easier to recognize and remember, such as www.ixiacom.com. To use domain names, you need either a Domain Name Server (DNS) set up in your network or an /etc/hosts file on each computer.

### Determining Your IP Network Address

To determine a Windows NT, Windows 2000, or Windows XP computer's local IP address, enter the following command:

```
IPCONFIG
```

If your TCP/IP stack is configured correctly, your output will look like the following (this output is taken from Windows NT 4.0):

```
Windows NT IP Configuration
Ethernet adapter AMDPCN1:
IP Address. . . . . . . . . : 10.10.44.3
Subnet Mask . . . . . . . . : 255.255.255.0
Default Gateway . . . . . . : 10.10.44.254
```

Its local IP address is shown in the first row; here it's 10.10.44.3.

You can also find your IP address using the graphical user interface. Select the **Control Panel** folder, and double-click on the **Network** icon. The installed network components are shown. Double-click **TCP/IP Protocol** in the list to get to the **TCP/IP Configuration**. Your IP address and subnet mask are shown.

To determine a Windows NT, Windows 2000, or Windows XP computer's local hostname, enter the following command:

```
HOSTNAME
```

The current hostname is shown in the first row.

From the graphical user interface, return to the TCP/IP Protocol configuration. Select **DNS** (Domain Name System) to see or change your domain name. If the DNS Configuration is empty, avoid using domain names as network addresses; use numeric IP addresses instead.

### Testing the TCP/IP Connection

Ping is a simple utility program, included in all TCP/IP implementations. To check the connection from one computer to another, enter the following at an MS-DOS command prompt:

```
ping xx.xx.xx.xx
```

Replace the x's with the IP address of the target computer. If Ping returns a message that says "`Reply from xx.xx.xx.xx ...`," the Ping worked. If it says "`Request timed out`," the Ping failed, and you have a configuration problem.

Make sure that you can run Ping successfully from the IxChariot or Qcheck Console to each computer serving as Endpoint 1, and between each pair of endpoints involved in a test, before starting your testing with TCP/IP.

# Running Windows Endpoints

The following topics describe starting and stopping an endpoint in the Windows NT, Windows 2000, Windows XP, or Windows Server 2003 operating systems, as well as some of the messages and information that become available during testing with this endpoint. The Windows NT/2000/XP endpoint is controlled from the Services dialog box. For Windows NT 4.0 or click **Settings**, then **Control Panel** on the Start menu, then double-click **Services**. For Windows 2000, click **Settings**, then **Control Panel** on the Start menu, double-click **Administrative Tools**, and then double-click **Services**. The Services dialog box lets you start or stop the endpoint, listed as "Ixia Endpoint."

Only a user ID with Administrator authority is permitted to start or stop Windows NT, Windows 2000, Windows XP, or Windows Server 2003 services.

## Starting the Endpoint

By default, the endpoint program is configured to start automatically, which means that you will not see a window for the program when it is running. Because the endpoint runs as a service, you do not have to be logged into your workstation for the endpoint to run.

If you stop the endpoint service, you can restart it without restarting Windows NT, Windows 2000, or Windows XP. There are two ways to restart the endpoint service:

1. At a command prompt, enter:

   ```
   net start IxiaEndpoint
   ```

2. In the Services dialog box, select **Ixia Endpoint** and click **Start** (or **Play**). The status changes to "started" when the endpoint is successfully started.

**Note**: A single running copy of the endpoint service handles one or multiple concurrent tests.

## Stopping a Windows Endpoint

There are two ways to stop the endpoint service:

- At a command prompt, enter the following:

  ```
  net stop IxiaEndpoint
  ```

- In the Services dialog box, click **Ixia Endpoint** and click **Stop**. The status is blank when the endpoint program has stopped.

### Disable Your Screen Saver

Screen savers in Windows NT and Windows 2000 can significantly lower the throughput that's measured by an endpoint. We recommend disabling your screen saver at endpoint computers while running tests.

## The SetAddr Utility

Endpoints for Windows operating systems now ship with a utility that helps you quickly create virtual IP addresses on Windows NT, Windows 2000, and Windows XP (32-bit and 64-bit) endpoint computers. Virtual addresses are chiefly useful when you're testing hundreds or even thousands of endpoint pairs using only a few computers as endpoints. To all intents and purposes, the traffic on the network is identical, whether you're using "real" or virtual addresses.

For more information about creating virtual addresses, consult "Configuring Virtual Addresses on Endpoint Computers" in the *User Guide* for IxChariot.

When you install a Windows endpoint, `Setaddr.exe` for 32-bit Windows is automatically installed in the same directory. For 64-bit Windows, a 64-bit version of `Setaddr.exe` is installed. The two versions of SetAddr cannot be used across operating systems with different architectures.

The usage is as follows:

```
setaddr [-dr] -a N -f Addr -t Addr -i Addr -s Addr

| -l[a]

| -da

| -ds -f Addr -s Addr
```

(where "`N`" indicates the adapter number of the NIC card you're assigning virtual addresses to, and "`Addr`" indicates the virtual addresses or subnet mask you're assigning to it).

**Options**:

```
-l     List all network adapters

-la    List all network adapters and their IP addresses

-a     Adapter to modify (number given by -l options)

-dr    Delete a range of addresses

-da    Delete all addresses

-ds    Delete a single address

-f     From address

-t     To address

-i     Increment by

-s     Subnet Mask
```

The `-d` flags cannot be used to delete a computer's primary IP address.

The `-i` flag lets you determine how the range of addresses will be created. This is an optional field; by default, SetAddr increments the range by one in the final byte only. This “`increment by`” value is represented as “`0.0.0.1`”. Enter a value (0-255) for each byte of the 4-byte IP address. A value of `1` specifies that the address values in that byte will be incremented by one when SetAddr creates the range. For example, enter

```
setaddr -f 10.40.1.1 -t 10.40.4.250 -i 0.0.1.1 -s
255.255.0.0
```

SetAddr creates 1000 virtual addresses.

**Known Limitations:**

- IPv4 only.
- Windows NT, Windows 2000, and Windows XP computers only.
- SetAddr only works on computers with fixed IP addresses. DHCP-enabled adapters can’t be used.
- You must restart the computer to whose NIC you’ve assigned virtual IP addresses before you begin testing with that computer. SetAddr modifies some Windows Registry keys, and restarting is required for the changes to take effect.
- The number of virtual addresses you can assign to a single adapter depends on the protocol stack and the size of the Windows Registry. We benchmarked measurements using computers running up to 2500 virtual addresses, which is a recommended limit.
- No checking is done to ensure that thousands of addresses are not being created. Be careful! More TCP/IP stack resources are required to manage virtual addresses.
- You may only add Class A, B, and C virtual IP addresses. Loopback addresses and Class D and E IP addresses are invalid. Valid address ranges, then, are `1.x.x.x` to `233.x.x.x`, excluding `127.x.x.x`.
- When more than 2250 virtual address are defined on Windows 2000 computers, all the LAN adaptor icons disappear from the Network and Dial-up Connections dialog box in My Network Places. You can still see the adaptors by invoking `ipconfig` or `setaddr` from the command line, and the addresses are still reachable. Removing some virtual addresses so that fewer than 2250 were specified and restarting the computer solved the problem.

## Disabling Automatic Startup

**To disable the automatic starting of the endpoint, take the following steps in Windows 2000:**

1. On the Start menu, click **Settings**, then **Control Panel**, then **Administrative Tools**, then **Services**. The Services dialog box appears.
2. Double-click **Ixia Endpoint**.
3. On the Startup type menu, click **Manual**.
4. Click **OK** to save the new setting and exit the dialog box. The endpoint will no longer start automatically when you restart the computer. However, you can manually start the endpoint.

**Take the following steps in Windows NT:**

1. On the Start menu, click **Settings** and then **Control Panel**. The Control Panel appears.
2. Double-click the **Services** icon.
3. Highlight **Ixia Endpoint** and click **Startup**.
4. Click **Manual**.
5. Click **OK** and then **Close**. The endpoint will no longer start automatically when you restart the computer. However, you can manually start the endpoint.

### How to Tell If a Windows Endpoint Is Active

The status field in the Services dialog box shows whether the Ixia Endpoint service has started.

Similarly, the Windows Performance Monitor program can be used to look at various aspects of the endpoint. Start Performance Monitor by double-clicking its icon in the Administrative tools group. Click **Add to Chart** on the Edit menu. Select the **Process** object and the **Endpoint** instance. Then add the counters you are interested in, such as thread count or % of processor time. In the Steady state (that is, no tests are active), Thread Count will show about 6 threads active for the endpoint; the answer depends on the number of protocols in use.

## Logging and Messages

While most error messages encountered on an endpoint are returned to the IxChariot or Qcheck Console, some may be logged to disk. Errors are saved in a file named ENDPOINT.LOG, in the directory where you installed the endpoint. To view an error log, use the command-line program named FMTLOG.EXE. The program FMTLOG.EXE reads from a binary log file, and writes its formatted output to stdout. Use the following FMTLOG command:

```
FMTLOG log_filename > output_file
```

This endpoint performs extensive internal cross-checking to catch unexpected conditions early. If an assertion failure occurs, the file assert.err is written to the directory where you installed the endpoint.

### Application Monitoring Support with CheckPoint VPN Software

Near the end of the endpoint installation, you are asked if you want to install application monitoring support. This is ***not*** recommended; the README file for the Windows endpoint has a full list of known interaction issues.

# Getting the Latest Fixes and Service Updates

We've found that communications software is often fragile. Its developers are constantly working to make it more robust, as the software gets used in an ever-wider set of situations.

We therefore recommend working with the very latest software for the underlying operating system and communications software. Here are the best sources we've found for the Windows NT, Windows 2000, or Windows XP software used by the endpoint program.

## Updates and Information for Windows

Microsoft posts code and driver updates to the following Web site: www.microsoft.com/windows/downloads/.

For information about configuring TCP/IP to make it work better on Windows NT, consult the following Web site: www.microsoft.com/windows2000/techinfo/howitworks/communications/networkbasics/tcpip_implement.asp.

## Updates for Microsoft SNA Server

Microsoft posts code and driver updates to the following Web site: http://support.microsoft.com/support/sna/sp.asp.

## Updates for IBM SNA Software for Windows

For information on IBM's Personal Communications (PCOMM) family of software, see: www.software.ibm.com/network/pcomm/support/.

# 12 Microsoft Windows XP/ 2003 64-bit Edition

The following topics explain the installation, configuration, and operation of the Performance Endpoint software for Windows XP/2003 64-bit Edition.

This endpoint is designed to run on the IA-64 architecture or AMD Opteron. On a 32-bit Windows system, only 32-bit binaries will be installed.

## Installation Requirements for Windows XP/2003 64-Bit Edition Endpoints

The endpoint requirements for Microsoft Windows XP/2003 64-bit Edition are:

- A computer capable of running Windows XP/2003 64-bit Edition well.

  The computer should have a processor based on the IA-64 architecture or AMD64 architecture, such as the Intel Itanium/Itanium 2 or the AMD Opteron/Athlon FX/Athlon 64 CPU.
- 512 MByte of random access memory (RAM).

  The total RAM requirement depends on the RAM usage of the underlying protocol stack and the number of concurrent connection pairs. For very large tests involving hundreds of connections through a single endpoint, additional memory may be required.
- A hard disk with at least 10 MBytes of space available.
- Windows XP/2003 64-bit Edition

You also need compatible network protocol software:

- For RTP, TCP, and UDP

  TCP/IP software is provided as part of the network support with Windows XP/2003 64-bit Edition.
- for IPX, SPX and APPC

  Windows XP/2003 64-bit edition does not support IPX/SPX or APPC.

We recommend that you get up-to-date with the latest Windows service levels.

# Endpoint Installation for Windows XP/2003 64-bit Edition

We recommend configuring your networking software—and ensuring that it is working correctly—before installing Ixia Endpoint software. See the Help for your networking software, and see *Configuring 64-bit Windows XP/2003 Endpoints* on page 12-5 for more assistance.

The endpoint for Windows XP/2003 64-bit Edition is installed and runs as a service. Only a user with Administrator authority is permitted to install services. To successfully install the endpoint, you must be logged in with Administrator authority. If you are installing the endpoint in an NTFS directory, the permissions of the directory must also be set to allow the `SYSTEM` (the operating system) full control access. Be sure to give the System "Full Control" permission on all files in the `Ixia\Endpoint` directory or the directory where you've installed the endpoint, plus any relevant subdirectories, if any.

The following are directions for installing the endpoint **from a CD-ROM** and **from the World Wide Web**:

**To install the endpoint from a CD-ROM, do the following:**

1. Put the CD-ROM in your CD-ROM drive.
2. Navigate to the directory **Windows** and enter the following:

   ```
   [drive:]\Endpoint\Windows\pewindows_xxx.exe
   ```

   where *xxx* is the version of the software without any periods. For example, when installing the endpoint from release version 5.20, *xxx* would be *520.*
3. If you have a previous installation of the endpoint, you will be asked if you want it removed. If you select "**yes**," the previous installation is removed, and the new installation continues. If you select "**no**," the install program exits with no changes to your existing installation because a new version cannot be added until the old version is removed. The installation program adds the endpoint program as a service.
4. Select the directory where the endpoint will be installed. We recommend installing it on a local hard disk of the computer you're using. If you install on a LAN drive, the additional network traffic may influence your performance results. The default directory is `\Program Files\Ixia\Endpoint`, on your boot drive.
5. The next dialog box contains two checkboxes.
   - **Install prebuilt data files.** We recommend you leave this box checked. You can save a small amount of disk space by not installing the files used for compression testing—but the defaults in many application scripts specify these files. If these `.cmp` files are not installed, many application scripts cannot be used in tests until they are modified.
   - **Start the endpoint on installation.** If you leave the box unchecked, the endpoint starts when you restart the computer. No window is shown while the endpoint is running, since it runs as a service.

A Windows XP/2003 64-bit Edition service is controlled from the Services dialog box, accessible by selecting **Programs\Administrative Tools\Services** from the Start menu. If you want to restart a service without restarting Windows, use the Services dialog box. Go to the Services dialog, select **Ixia Endpoint**, and select a Startup type from the pull-down. Press **Start** to start the endpoint.

You can also manually start the endpoint after installation. See *Starting a Windows XP/2003 Endpoint* on page 12-7 for instructions.

The copying of files is now complete; you can remove the CD-ROM from its drive.

To prevent the endpoint from running automatically on startup, see *Disabling Automatic Startup* on page 12-8.

When you've completed installation, refer to *Configuring 64-bit Windows XP/2003 Endpoints* on page 12-5 to make sure your endpoint is ready for testing and monitoring.

**To install an endpoint you've downloaded from the World Wide Web, do the following:**

1. Save the `pewia64_xxx.exe or pewamd64_xxx.exe` file to a local directory.
2. Use the Windows Explorer to navigate to the file and double-click to start the installation.
3. Read and agree to the End-User License Agreement.
4. If you have a previous installation of the endpoint, you will be asked if you want it removed. If you select "**yes**," the previous installation is removed, and the new installation continues. If you select "**no**," the install program exits with no changes to your existing installation because a new version cannot be added until the old version is removed.
5. The next screen lets you select the directory where the endpoint will be installed. We recommend installing it on a local hard disk of the computer you're using. If you install on a LAN drive, the additional network traffic may influence your performance results. The default directory is `\Program Files\Ixia\Endpoint`, on your boot drive.
6. The next step installs the Endpoint software to your hard disk.
7. Finally, you are asked if you wish to view the *Readme* file to check on last minute release notes.

   The Endpoint service is controlled from the Services dialog, accessible by selecting **Start - Settings - Control Panel - Administrative Tools - Services** from the Start menu. If you want to restart a service without restarting Windows, from the Services dialog box, select **Ixia Endpoint**, and select a Startup type from the pull-down. Click **Start** to start the endpoint.

   You can also manually start the endpoint after installation. See *Starting a Windows XP/2003 Endpoint* on page 12-7 for instructions.

The copying of files is now complete.

To prevent the endpoint from running automatically on startup, see *Disabling Automatic Startup* on page 12-8.

If you want to restore the setting later, you must do so manually.

When you've completed installation, refer to *Configuring 64-bit Windows XP/ 2003 Endpoints* on page 12-5 to make sure your endpoint is ready for testing and monitoring.

## Unattended Installation for 64-bit Windows XP/2003

Unattended installation (also called *silent installation*) is available. You install an endpoint once, by hand, while the install facility saves your input in a *response* file. You can then install that same endpoint silently on other computers, that is, without providing input other than the answer file.

First, unzip the `pewia64_xxx.exe or pewamd64_xxx.exe` file from the CD. We recommend using WinZip version 7.0 and higher.

When installing, specify the "`-r`" option on `SETUP` to save your input. For example, to install for the first time, enter:

```
[drive:]\SETUP -r
```

where "`[drive:]`" is the drive where the install package is located. This produces the response file named `setup.iss`, which can then be used on subsequent silent installations. The **`setup.iss`** answer file is created in your Windows directory (which is usually `c:\windows`).

If you want to create a response file without actually installing an endpoint, enter:

```
[drive:]\SETUP noinst -r
```

To perform a silent installation, specify the "`-s`" option on `SETUP`. Make sure the answers documented in the answer file **`setup.iss`** are appropriate for the silent installation. If the `setup.iss` file is not in the same directory as **`setup.exe`**, then specify the path and filename with the "`-f1`" option. For example, here's how to install using the `setup.iss` file we placed in the `\Program Files\Ixia\Endpoint` directory on our `n:` LAN drive:

```
SETUP -s -f1n:\Program Files\Ixia\Endpoint\setup.iss
```

Don't mix the `.iss` files among different Windows operating systems because their endpoint installations require slightly different input.

The results of the silent installation are recorded in a file named `setup.log`, which is created in your `Windows` directory.

It's common to use unattended install from a LAN drive. Be sure you've copied all of the files for each type of endpoint into a single directory (rather than into separate diskette images), and you've created your initial **`setup.iss`** file from that directory.

## What We Do During Installation

Here's what happens during the installation steps. Let's say you install the endpoint into the directory `\Program Files\Ixia\Endpoint`. A directory is created with the following contents:

- the executable programs;
- the `README` file;
- the directory `Cmpfiles`. This directory contains files with the `.CMP` file extension. These are files containing data of different types, such as typical text or binary data. These files are used by the endpoint as data on `SEND` commands. The different data types can be used to vary the data compression performance of your network hardware and software.
- the file `endpoint.ini`

  See Chapter 2, *Endpoint Initialization File* for information about tailoring this file for individual endpoints.

The endpoint is installed as a service, which means there's nothing visible while it's running. During installation, the endpoint is configured to automatically start when the system reboots. Controlling the endpoint from the Services dialog box is described in *Running 64-bit Windows XP/2003 Endpoints* on page 12-7.

## Removing the Endpoint Package (Uninstall)

To remove the endpoint package from your hard disk, follow these steps:

1. Click **Start** > **Settings** > **Control Panel**.
2. Click **Add or Remove Programs**. The Add or Remove Programs Properties dialog box is shown.
3. Highlight **Ixia Endpoint** and press **Change/Remove**. The un-installation program begins. After the program is completed, the endpoint should be uninstalled.

# Configuring 64-bit Windows XP/ 2003 Endpoints

The endpoint program uses network application programming interfaces such as WinSock for all of its communications. The endpoint dynamically configures its own programs, so you do not have to update the configuration files for your communications software. However, your communications software must be configured and running correctly. The following steps guide you through this verification process.

1. Determine the network addresses of the computers to be used in tests.
2. Select a service quality.
3. Verify the network connections.

The following topics describe how to accomplish these steps for XP/2003.

## Windows XP/2003 Configuration for TCP/IP

The RTP, TCP, and UDP protocols use TCP/IP software for network communications. TCP/IP offers two forms of network addresses: IP addresses and domain names. An IP address is a 32-bit (IPv4) or 128-bit (IPv6) numeric address. IPv4 addresses are represented in dotted notation as a set of four numbers separated by periods, such as `199.72.46.202`. IPv6 addresses are represented by up to 8 colon separated hex digit pairs, such as 0::FF. An alternative, domain names are in a format that is easier to recognize and remember, such as www.ixiacom.com. To use domain names, you need either a Domain Name Server (DNS) set up in your network or an `/etc/hosts` file on each computer.

## Determining Your IP Network Address

To determine an XP/2003 computer's local IP address, enter the following at a command prompt:

```
IPCONFIG
```

If your TCP/IP stack is configured correctly, your output will look like the following:

```
Windows IP Configuration

Ethernet adapter Local Area Connection:
Connection-specific DNS Suffix . :
IP Address. . . . . . . . . . . . : 10.41.2.19
Subnet Mask . . . . . . . . . . . : 255.255.0.0
Default Gateway . . . . . . . . . : 10.41.1.254
```

The local IP address is shown in the first row; here it's 10.41.2.19.

For IP addresses not configured by DHCP, you can also find your IP address using the graphical user interface. Select **Start - Settings - Control Panel**, then double-click on the **Network Connections** icon. Select Local Area Connection and click **Properties**. In the Local Area Connection Properties dialog box, double-click **Internet Protocol (TCP/IP)** in the list. Your IP address and subnet mask are shown.

To determine a Windows XP/2003 computer's local hostname, enter the following at a command prompt:

```
HOSTNAME
```

The current hostname is shown in the first row.

From the graphical user interface, return to **Internet Protocol (TCP/IP)** configuration. Press **Advanced** and then select the **DNS** tab to see or change your DNS servers. If the DNS tab is empty, avoid using domain names as network addresses; use numeric IP addresses instead.

The default location for the `/etc/hosts` file is the following:

```
c:\WINDOWS\SYSTEM32\DRIVERS\ETC\HOSTS
```

### Trying Out the TCP/IP Connection

*Ping* and *ping6* are simple utility programs, included in all TCP/IP implementations. They are used to check the connection from one computer to another using either IPv4 or IPv6 addresses. For *ping*, enter the following at a command prompt:

```
ping xx.xx.xx.xx
```

Replace the x's with the IP address of the target computer. If Ping returns a message that says "`Reply from xx.xx.xx.xx ...`," the Ping worked. If it says "`Request timed out`," the Ping failed, and you have a configuration problem.

For *ping6* enter an address in standard IPv6 format.

Make sure that you can run *ping/ping6* successfully from the IxChariot or Ixia Qcheck Console to each computer serving as Endpoint 1, and between each pair of endpoints involved in a test, before starting your testing with TCP/IP. When using alternate networks, the alternate networks need to be tested as well.

### Sockets Port Number

TCP/IP applications use their network address (as described above) to decide which computer to connect to in a network. They use a Sockets *port number* to decide which application program to connect to within a computer.

The TCP/IP sockets port used by IxChariot endpoints is **10115**. This port number is used during the initialization of a test; during the actual running of the test, other port numbers are used. If the script specifies "`port_number=AUTO`" on the `CONNECT_ACCEPT` command, additional ports are dynamically acquired from the protocol stack. Otherwise, the endpoint issuing the `CONNECT_ACCEPT` commands (usually Endpoint 2) uses the port number specified in the script.

## Running 64-bit Windows XP/2003 Endpoints

The following topics describe starting and stopping an endpoint running on the Windows XP/2003 operating system, as well as some of the messages and information that become available during testing with this endpoint. The Windows XP/2003 64-bit endpoint is controlled from the Services dialog box, which you access by selecting **Start - Settings - Administrative Tools - Services** from the Start menu. The Services dialog box lets you start or stop the endpoint, listed as **Ixia Endpoint**.

Only a user with Administrator authority is permitted to start or stop Windows XP/2003 services.

### Starting a Windows XP/2003 Endpoint

By default, the endpoint program is configured to start automatically, which means that you will not see a window for the program when it is running. Because the endpoint runs as a service, you do not have to be logged into your workstation for the endpoint to run.

If you stop the endpoint service, you can restart it without restarting XP/2003. There are two ways to restart the endpoint service:

1. At a command prompt, enter:

   ```
   net start IxiaEndpoint
   ```

2. In the Services dialog box, double-click **Ixia Endpoint** and press **Start**. The status changes to “started” when the endpoint is successfully started.

A single running copy of the endpoint service handles one or multiple concurrent tests.

## Stopping a Windows XP/2003 Endpoint

There are two ways to stop the endpoint service:

1. At a command prompt, enter the following:

   ```
   net stop IxiaEndpoint
   ```

2. In the Services dialog box, double-click **Ixia Endpoint** and click **Stop**. The status is blank when the endpoint program has stopped.

## Disable Your Screen Saver

Screen savers can significantly lower the throughput that’s measured by an endpoint. We recommend disabling your screen saver at endpoint computers while running tests.

## Disable NIC Power Save Mode

If your NIC is configured to power down after some period of non-traffic, this might cause your test to fail.

## Disabling Automatic Startup

To disable the automatic starting of the XP/2003 endpoint, take the following steps:

1. From the Windows Start menu, select **Programs\Administrative Tools\Services**. The Services dialog is shown.
2. Double-click **Ixia Endpoint**.
3. From the Startup type menu, select **Manual**.
4. Press **OK** to save the new setting and exit the dialog. The endpoint will no longer start automatically when you restart the computer. However, you can manually start the endpoint.

## How to Tell If a Windows XP/2003 Endpoint Is Active

The status field in the Services dialog box shows whether the Ixia Endpoint service has started.

## The SetAddr Utility for 64-bit Windows

Endpoints for Windows operating systems now ship with a utility that helps you quickly create virtual IP addresses on 64-bit Windows endpoint computers. Virtual addresses are chiefly useful when you’re testing hundreds or even thousands of endpoint pairs using only a few computers as endpoints. To all intents and purposes, the traffic on the network is identical, whether you’re using “real” or virtual addresses.

- For more information about creating virtual addresses, consult “Configuring Virtual Addresses on Endpoint Computers” in the *User Guide* for IxChariot.

- When you install a Windows endpoint, `Setaddr.exe` for 64-bit Windows is automatically installed in the same directory. The usage is as follows:

```
setaddr [-dr] -a N -f Addr -t Addr -i Addr -s Addr

  | -l[a]

  | -da

  | -ds -f Addr -s Addr
```

(where “`N`” indicates the adapter number of the NIC card you’re assigning virtual addresses to, and “`Addr`” indicates the virtual addresses or subnet mask you're assigning to it).

```
Options:

-l      List all network adapters

-la     List all network adapters and their IP addresses

-a      Adapter to modify (number given by -l options)

-dr     Delete a range of addresses

-da     Delete all addresses

-ds     Delete a single address

-f      From address

-t      To address

-i      Increment by

-s      Subnet Mask
```

The `-d` flags cannot be used to delete a computer’s primary IP address.

The `-i` flag lets you determine how the range of addresses will be created. This is an optional field; by default, `SetAddr` increments the range by one in the final byte only. This “increment by” value is represented as “`0.0.0.1`”. Enter a value (0-255) for each byte of the 4-byte IP address. A value of 1 specifies that the address values in that byte will be incremented by one when SetAddr creates the range. For example, enter

```
setaddr -f 10.40.1.1 -t 10.40.4.250 -i 0.0.1.1 -s
255.255.0.0
```

SetAddr creates 1,000 virtual addresses.

### Known Limitations

- A version of `SetAddr` is also available for Windows NT, Windows 2000, and Windows XP/2003 32-bit computers. This 64-bit Windows version of `SetAddr` does not work on 32-bit systems.
- `SetAddr` only works on computers with fixed IP addresses. DHCP-enabled adapters can’t be used.
- You must restart the computer to whose NIC you've assigned virtual IP addresses before you begin testing with that computer. `SetAddr` modifies

some Windows Registry keys, and restarting is required for the changes to take effect.

- The number of virtual addresses you can assign to a single adapter depends on the protocol stack and the size of the Windows Registry. We have benchmarked measurements using computers running up to 2500 virtual addresses, which is a recommended limit.
- No checking is done to ensure that thousands of addresses are not being created. Be careful! More TCP/IP stack resources are required to manage virtual addresses.

## Logging and Messages

While most endpoint error messages are returned to the IxChariot or Ixia Qcheck Console, some may be logged to disk. Errors are saved in a file named ENDPOINT.LOG, in the directory where you installed the endpoint. To view an error log, use the command-line program named FMTLOG.EXE. Program FMTLOG.EXE reads from a binary log file, and writes its formatted output to stdout. Use the following FMTLOG command:

```
FMTLOG log_filename > output_file
```

This endpoint has extensive internal cross-checking to catch unexpected conditions early. If an assertion failure occurs, the file assert.err is written to the directory where you installed the endpoint.

## Getting the Latest Fixes and Service Updates

We've found that communications software is often fragile. Its developers are constantly working to make it more robust, as the software gets used in an ever-wider set of situations.

We therefore recommend working with the very latest software for the underlying operating system and communications software. Following are the best sources we've found for information and upgrades for 64-bit Windows XP/2003.

### Updates and Information for Windows XP/2003

In order to keep your Windows XP/2003 system up-to-date, you should use the Windows Update function available from your Start Menu.

# 13 Sun Solaris

This chapter explains the installation, configuration, and operation of the Performance Endpoint software for Sun Solaris version 2.4 (or later). The endpoints operate on the "SPARC" and "x86" versions of Solaris.

- SPARC computers contain CPUs made by Sun Microsystems and others.
- x86 computers are commonly known as "Intel-compatible PCs"; they contain CPUs made by Intel, AMD, Cyrix, or others.

## Installation Requirements for Solaris Endpoints

Here's what you need to run the endpoint program with Sun Solaris:

- A computer capable of running Sun Solaris well.

  For SPARC computers, any system seems to give good performance.

  For x86 computers, this implies a CPU such as an Intel 80386, 80486, a member of the Pentium family, or equivalent. A Pentium or better is recommended.

- At least 32 MBytes of random access memory (RAM).

  The total RAM requirement depends on the RAM usage of the underlying protocol stack and the number of concurrent connection pairs. For large tests involving hundreds of connections through a single endpoint, additional memory may be required.

- A hard disk with at least 4 MBytes of space available.
- Sun Solaris version 2.4 or later, with TCP/IP networking and corresponding networking hardware installed and configured. This version also supports IP Multicast.
- An Acrobat Reader to view the `.PDF` files.

  Acrobat readers are loaded on most computers for viewing other documents, but if you do not have one, they are available at Adobe's Web site: www.adobe.com/prodindex/acrobat/readstep.html.

NOTE: in the following discussion, the name of the HP endpoint file is pewsun_*Mm*.tar or pes86_*Mm*.tar, where *Mm* is the major and minor IxChariot version number; for example *520* for IxChariot release 5.20

# Endpoint Installation for Sun Solaris

First, make sure that you are logged in as a "root" user. Also, remember that all the commands and parameters discussed here are case-sensitive; use the combination of uppercase and lowercase letters as shown. The following instructions explain how to install an endpoint from a CD-ROM and from the World Wide Web.

**Note**: To install version 4.4 of the Endpoint for Sun Solaris over a previous version of the endpoint, you need to modify the admin file to contain "`instance=overwrite`" and "`conflict=nocheck`."

**To install the endpoint from a CD-ROM, do the following:**

1. Put the CD-ROM in your CD-ROM drive.
2. Next, enter the `VOLCHECK` command, which tells Solaris that the CD-ROM is inserted in the drive and is readable. `VOLCHECK` returns quickly to the command prompt, without a message.

   ```
   volcheck
   ```

3. The CD-ROM contains an archive of the endpoint package. First use the `rm` command to ensure a clean temporary install directory. Then use the `tar` command to extract the archive contents from the CD-ROM.

   For SPARC systems, enter:

   ```
   cd /tmp
   rm -fr endpoint
   tar -xvf /cdrom/endpoint/solaris/pesun_Mm.tar
   ```

   For x86 systems, enter:

   ```
   cd /tmp
   rm -fr endpoint
   tar -xvf /cdrom/endpoint/s86/pes86_Mm.tar
   ```

4. Next, install the endpoint package using the `pkgadd` command:

   ```
   pkgadd -d /tmp endpoint
   ```

   The `pkgadd` command is not part of the endpoint installation. It is part of the standard Solaris installation and can be found in the `/usr/bin` directory.

5. You will see the license agreement, presented with the `pg` command. Press the spacebar until the end of the agreement is displayed. You are asked whether you accept the terms and conditions of the agreement. If you do, enter "`accept_license`" and press Return.
6. Next, you are asked the following question:

```
This package contains scripts which will be executed
with super user permission during the process of
installing this package.

Do you want to continue with the installation of this
package [y,n,?]
```

Enter a lowercase “y” to complete the installation script. About 20 lines of text give the status of the installation. When it’s finished, the last line reads:

```
Installation of <endpoint> was successful.
```

You may instead see the following message:

```
Notice! There were potential problems with migrating
from $oldInstallPath to $installPath. Review the
warnings displayed above for further explanation.
```

If you see this message, please review the entire output from the install script for an explanation of the warnings and further instructions.

**To delete the archive contents from the temporary working directory:**

```
cd /tmp
rm -fr endpoint
```

Remove the CD-ROM by entering `eject` at a command prompt.

This is a good time to read the `README` file, installed with the endpoint in `/opt/ixia`, for the latest information about the endpoint program.

When you’ve completed installation, refer to *Configuring Solaris Endpoints* on page 13-6 to make sure your endpoint is ready to be used in testing and monitoring.

**To install an endpoint you’ve downloaded from the World Wide Web, do the following:**

1. First, use the `rm` command to ensure a clean temporary install directory (we’ll use `tmp` in this example).

   **For SPARC systems:**

   - Download the `pesun_Mm.tar.Z` file to the `/tmp` directory.
   - Uncompress the endpoint file by using the `uncompress` command:

   ```
   cd /tmp
   uncompress pesun_Mm.tar
   tar -xvf pesun_Mm.tar
   ```

   **For x86 systems:**

   - Download the `pes86_Mm.tar.Z` file to the `/tmp` directory.
   - Uncompress the endpoint file by using the `uncompress` command:

   ```
   cd /tmp
   uncompress pes86_Mm.tar
   tar -xvf pes86_Mm.tar
   ```

2. Next, install the endpoint package using the `pkgadd` command:

   ```
   pkgadd -d /tmp endpoint
   ```

The pkgadd command is not part of the endpoint installation. It is part of the standard Solaris installation and can be found in the /usr/bin directory.

3. You will see the license agreement, presented with the pg command. Press the spacebar until the end of the agreement is displayed. You are asked whether you accept the terms and conditions of the agreement. If you do, enter “accept_license.”

4. You are next asked the following question:

   ```
   This package contains scripts which will be executed
   with super user permission during the process of
   installing this package. Do you want to continue with
   the installation of this package [y,n,?]
   ```

   Enter a lowercase “y” to complete the installation script. About 20 lines of text give the status of the installation. When it’s finished, the last line reads, “Installation of <endpoint> was successful.”

   You may instead see the following message:

   ```
   Notice! There were potential problems with migrating
   from $oldInstallPath to $installPath. Review the
   warnings displayed above for further explanation.
   ```

   If you see this message, please review the entire output from the install script for an explanation of the warnings and further instructions.

5. Use the following commands to delete the archive contents from the temporary working directory:

   ```
   cd /tmp
   rm -fr endpoint
   rm pes86_Mm.tar
   ```

This is a good time to read the README file, installed with the endpoint in /opt/ixia, for the latest information about the endpoint program.

When you’ve completed installation, refer to *Configuring Solaris Endpoints* on page 13-6 to make sure your endpoint is ready to be used in testing and monitoring.

## Installation Defaults File for Solaris

The admin file defines default installation actions to be taken when administrative input is required during install, for example, whether to allow a new package to overwrite an older version, whether an installation can be run with super user authority, and so on. The admin file is found in /var/sadm/install/admin/default. The man pages (“man -s 4 admin”) describe its format and content; please read the man pages if you are unfamiliar with the admin file.

To install version 4.4 of the Endpoint for Sun Solaris over a previous version of the endpoint, you need to modify the admin file to contain “instance=overwrite” and “conflict=nocheck.”

If you want non-interactive install capability, modify the admin file to contain “action=nocheck” so that the endpoint package scripts can be run with super user authority.

## Unattended Installation for Solaris

Unattended installation is available for the Sun Solaris endpoint. You install an endpoint once, manually, while the install facility saves your input in a *response* file. You can then install that same endpoint silently on other computers, that is, without providing input other than the response file.

First, complete the steps described in *Endpoint Installation for Sun Solaris* on page 13-2, using the `tar` command. Next create a response file, using the `pkgask` command:

```
pkgask -r /tmp/endpoint.response -d /tmp endpoint
```

The endpoint license agreement is displayed with the `pg` command. Press the spacebar until the end of the agreement is displayed. Next, you are asked whether you accept the terms and conditions of the agreement. If you do, enter "`accept_license.`"

You should see the following displayed:

```
Response file </tmp/endpoint.response> was created.
Processing of request script was successful.
```

Use the following command to install other Solaris endpoints in unattended mode (this single command is split over two lines):

```
pkgadd -n -a /tmp/endpoint/root/opt/ixia/admin
       -r /tmp/endpoint.response -d /tmp endpoint
```

The `pkgadd` command is not part of the endpoint installation. It is part of the standard Solaris installation and can be found in the `/usr/bin` directory.

When `pkgadd` is finished, the last line reads, "`Installation of <endpoint> was successful.`"

You may instead see the following message:

```
Notice! There were potential problems with migrating from
$oldInstallPath to $installPath. Review the warnings
displayed above for further explanation.
```

If you see this message, please review the entire output from the install script for an explanation of the warnings and further instructions.

The response file may be used to install the endpoint on each of your Sun Solaris computers.

## What Happens During Installation

Here's what happens during the installation steps. The endpoint is installed into the directory `/opt/ixia`. A directory is created with the following contents:

- The executable programs
- The `README` file
- Various install and uninstall programs
- The directory `cmpfiles`. This directory contains files with the `.cmp` file extension. These are files containing data of different types, such as typical

text or binary data. These files are used by the endpoint as data on SEND commands. The different data types can be used to vary the data compression performance of your network hardware and software.

- The file endpoint.ini. See Chapter 2, *Endpoint Initialization File* for information about tailoring this file for individual endpoints.

The installation program stops any copy of the endpoint program that may currently be running and starts a copy of the newly installed endpoint. You can run tests immediately, without a reboot.

Our software copies an S81endpoint initialization script to the /etc/rc2.d directory so the endpoint is started every time your system boots.

No changes are made to the PATH environment variable of the root user.

Should you have reason to install an older endpoint, you should delete any safestore files using the following steps:

1. Stop the endpoint.
2. Delete the safestore files from the endpoint directory (or from the directory specified by the SAFESTORE_DIRECTORY keyword in endpoint.ini). Safestore files have an extension of .q*; you may delete them using the command:

   ```
   rm *.q*.
   ```
3. Uninstall the current endpoint.
4. Install the desired endpoint.

### Removing the Endpoint Package (Uninstall)

Use the following command to remove the endpoint package (you must be logged in as root to run pkgrm):

```
pkgrm endpoint
```

Enter a lowercase “y” when you’re asked if you want to remove this package. About 10 lines of text give the status of the uninstallation. When it’s finished, the last line reads, “Removal of <endpoint> was successful.”

This removes the files from /opt/ixia, except for any files that were added to this directory that were not present at installation, such as the endpoint.ini file, and does not delete the directory. The removal program does not automatically delete files that have been added to the directory that you may need if you reinstall the product.

## Configuring Solaris Endpoints

The endpoint dynamically configures its own programs, so you do not have to update the configuration files for your communications software. However, your communications software must be configured and running correctly. The following steps guide you through this verification.

1. Determine the network addresses of the computers to be used in tests.

2. Verify the network connections.

The following sections discuss how to accomplish these tasks.

## Configuration for TCP/IP

The RTP, TCP, and UDP protocols use TCP/IP software for network communications. TCP/IP offers two forms of network addresses: IP addresses and domain names. An IP address is a 32-bit numeric address. It is represented in dotted notation as a set of four numbers separated by periods, such as `199.72.46.202`. The alternative, domain names are in a format that is easier to recognize and remember, such as www.ixiacom.com. To use domain names, you need either a Domain Name Server (DNS) set up in your network or an `/etc/hosts` file on each computer.

## Determining Your IP Network Address

Here are two ways to determine the IP address of the local computer you're using:

1. If you're using Sun's OpenWindows graphical user interface, right-click on the outer desktop background. One of the options in this Workspace menu that pops up is Workstation Info. Click on it to display Workstation Information about your computer, including your local Internet address.

```
netstat -in
```

2. As an alternative, enter the following at a command prompt:

Your local IP address is shown in the left-hand column, if there are active connections.

## Testing the TCP/IP Connection

Ping is a simple utility program, included in all TCP/IP implementations. To check the connection from one computer to another, enter:

```
ping xx.xx.xx.xx
```

Replace the `x`'s with the IP address of the target computer. If Ping returns a message that says "`xx.xx.xx.xx is alive`," the Ping worked.

Otherwise, there will be a delay, and then you'll see "`no answer from xx.xx.xx.xx`." This means that the Ping failed, and you can't reach the target computer.

Make sure that you can run Ping successfully from the IxChariot or Qcheck Console to each computer serving as Endpoint 1, and between each pair of endpoints involved in a test, before starting your testing with TCP/IP.

## Sockets Port Number

TCP/IP applications use their network address to decide which computer to connect to in a network. They use a Sockets *port number* to decide which application program to connect to within a computer.

The TCP/IP sockets port for endpoints is **10115**. This port number is used during the initialization of a test; during the actual running of the test, other port numbers are used. If the script specifies "`port_number=AUTO`" on the `CONNECT_ACCEPT` command, additional ports are dynamically acquired from the protocol stack. Otherwise, the endpoint issuing the `CONNECT_ACCEPT` commands (usually Endpoint 2) uses the port number specified in the script.

# Running Solaris Endpoints

The following sections describe how to manually start and stop the endpoint program, and how to examine error log files if a problem occurs.

## Starting a Solaris Endpoint

The endpoint program is installed so it will start automatically each time Solaris is rebooted. It sends its screen output to file `/var/adm/endpoint.console`. If you want to see any error messages generated at this endpoint, enter the following command:

tail -f /var/adm/`endpoint.console`

The detailed information about the start and stop of each individual connection pair is written to file `endpoint.aud`. The contents of this file vary depending on how you've set the `SECURITY_AUDITING` keyword in your `endpoint.ini` file.

See Chapter 2, *Endpoint Initialization File* for more information about `endpoint.aud` and `SECURITY_AUDIT` settings.

Instead of automatic startup, you can choose to manually start the endpoint program at a command prompt. Ensure that you are logged in as a "root" user. To start the endpoint, enter:

```
/opt/ixia/endpoint &
```

The "`&`" parameter indicates to Solaris that the endpoint program should run in the background. The screen output from the endpoint program is interleaved with other UNIX commands. Just press Return to enter more commands.

If you choose to manually start the endpoint, consider redirecting its output to the `endpoint.console` file. You can tell by the time stamp of the file when the endpoint program was started and stopped.

If the endpoint program is already running, you get the following message, "`CHR0183: The endpoint program is already running. Only one copy is allowed at a time.`"

## Stopping a Solaris Endpoint

The endpoint program has a special command-line option, `-k`. If you have an endpoint program you'd like to kill, go to a command prompt on the same computer and enter the following (you must be logged in as root to run this program):

```
/opt/ixia/endpoint -k
```

The `-k` command-line option has the purpose of killing any endpoint program running on that computer. You should see the message "`Sent exit request to the running endpoint`," which indicates that the endpoint program has been sent a request to stop.

If for some reason the request to stop is not handled by the running endpoint program correctly, you may need to use the UNIX "`kill -TERM`" command.

### Cleanup after Unexpected Errors

If the endpoint should fail or be killed abnormally (or encounter assertion conditions), you may also need to do additional cleanup. If the endpoint is still running, try to stop it using the command "`endpoint -k`". If that does not stop the endpoint, kill the endpoint using the UNIX `KILL` command.

Next, enter the following command:

```
rm /var/adm/.IXIA.ENDPOINT.PID
```

### How to Tell If a Solaris Endpoint Is Active

You can use traditional UNIX commands to determine if the endpoint program is active. At a command prompt, enter:

```
ps -ef | grep endpoint
```

If the endpoint program is running, it shows up with the following string in the right-most column of the output, "`/opt/ixia/endpoint`."

### Disabling Automatic Startup

To disable automatic startup, remove the `/etc/rc2.d/S81` endpoint file.

## Logging and Messages

While most error messages encountered on an endpoint are returned to the IxChariot or Qcheck Console, some may be logged to disk. Errors are saved in a file named `endpoint.log`, in the `/var/adm` directory. To view an error log, use the Ixia program named `FMTLOG`. `FMTLOG` reads from a binary log file, and writes its formatted output to `stdout`. Use the following `FMTLOG` command:

```
/opt/ixia/fmtlog log_filename >output_filename
```

The endpoint code does a lot of internal checking on itself. Our software captures details related to the problem in an ASCII text file named `assert.err` in the `/var/adm` directory. Save a copy of the file and send it to us via email for problem determination.

### Known Problems

You might see some operating-system problems during streaming tests. With test scripts running at a very fast rate or with many pairs using small datagram buffer sizes, the operating system may lock up.

Specifically, we have seen lock-up problems with Solaris version 2.6 and later when running certain kinds of streaming tests. We ran a 35-pair IxChariot test in which each pair used the Voice over IP Send script (`Voips.scr`). This script specifies small buffers (40 bytes each) at 64 kbps. Running this test to a Sun Ultra 5 computer (as the Endpoint 2) caused Solaris to completely lock up; the computer did not respond to network, keyboard, or mouse input.

We determined that the Endpoint 2 computer was overwhelmed with thousands of small datagrams, which the TCP/IP network stack could not process quickly enough. Either the RAM (in our case, the computer had 64 MB of RAM) or CPU power needs to be increased to handle the load.

We've also seen a recurring problem with Sun Solaris x86, version 2.4. The endpoint may stop, and a core dump may occur during testing. We have traced this problem to a Solaris software bug, which is solved with the latest OS patch. Download the patch from one of the following Web sites:

http://access1.sun.com/patch.public/cgi-bin/readme2html.cgi?patch=101946&type=rec

http://access1.sun.com/patch.public/cgi-bin/show_list.cgi/rec/Solaris_Intel_2.4

### Message CHR0181

You may receive message CHR0181 while running a test. If the error was detected at the Sun Solaris computer, it says that the endpoint program on Sun Solaris has run out of system semaphores. Each instance of Endpoint 1 requires a system semaphore. The maximum number of semaphores is not configurable on Sun Solaris; it is hard-coded to a large value. To avoid this problem, stop other programs that use semaphores or decrease the number of tests that use the computer as Endpoint 1.

## Updates for Sun Solaris

We've found that communications software is often fragile. Its developers are constantly working to make it more robust, as the software gets used in an ever-wider set of situations.

We therefore recommend working with the very latest software for the underlying operating system and communications software.

Sun posts code and driver updates directly to the following Web sites:

- www.sun.com/
- Anonymous FTP to ftp://ftp.sun.com/

# 14 Web-Based Performance Endpoint

Ixia Performance Endpoints are lightweight software agents that allow for testing and monitoring of computers and computer networks. Endpoints are available for more than 20 operating systems and are continually updated to support new features in IxChariot and Qcheck.

Unlike the endpoints for all of the other supported platforms, the Web-Based Performance Endpoint was not designed to be installed on a computer. Users of the Web-Based endpoint can either run it from the World Wide Web or save it to a local hard disk, but as soon as they restart, or log out of, the computer where it is running, the endpoint stops running.

The Web-Based endpoint runs on the following operating systems:

- Windows 98
- Windows NT 4.0
- Windows 2000
- Windows XP (32-bit version only)
- Windows Server 2003.

While endpoints for other operating systems still run as long as the computer where they're installed is powered on, the Web-Based endpoint stops running as soon as the user logs out or restarts. Nothing has been written to the Registry on the computer where it ran.

The Web-Based endpoint supports most IxChariot and Qcheck functions. A few features are not supported. The following table summarizes the IxChariot and Qcheck features that are not supported:

Table 14-1. Supported/Unsupported Features

| Function | Comment |
|---|---|
| APPC protocol | |
| SPX, IPX protocols | |
| `Endpoint.ini` file | Default settings cannot be changed. |

Table 14-1. Supported/Unsupported Features (Continued)

| Function | Comment |
|---|---|
| Application script datatypes (other than `ZEROES` or `NOCOMPRESS`) | IxChariot scripts that use a `send_datatype` parameter will fail. |
| Traceroute testing | |

## Running the Web-Based Endpoint

To run the Web-Based endpoint on your local computer, use the Web browser on that computer to navigate to www.ixiacom.com/support/ixchariot. Click the link labeled **Web-Based Endpoint**.

Unless you are using a utility like RealDownload to download files from the Web, you are then asked if you want to run the software from its present location or save it to disk. Click to select one of these options.

- Run from Location:

  If the download is successful, you'll see a message stating that the endpoint has been started. Click **OK** to close the message.

  There's nothing else you need to do. The endpoint is ready for testing with IxChariot or Qcheck.

- Save to Disk:

  If you want to save it to disk, save it to the folder where you save your temporary files, such as `Temp`.

  Navigate to the folder where you've saved the endpoint. Double-click the file `endpoint.exe` to start the endpoint. You'll see a message stating that the endpoint has been started. Click **OK** to close the message.

  After you start the endpoint, there's nothing else you need to do. The endpoint is ready for testing with IxChariot or Qcheck.

**Note**: When you save the endpoint to a local hard disk, it makes no difference where you save it. When you restart the computer, a copy of the executable `endpoint.exe` will still be on your hard drive, but it will no longer run until you restart it. Restart the executable by double-clicking it in the Windows Explorer.

## Error Handling

Unlike endpoints for other operating systems, the Web-Based endpoint doesn't log errors it encounters. However, it does report errors to IxChariot and Qcheck.

In the case of a connection failure or other failure during testing, the endpoint vanishes silently. The Console will notify you that it can no longer reach the endpoint. You should return to the Web and re-enable the endpoint in the case of such a failure.

## Compatibility with Other Endpoints

The Web-Based endpoint cannot run on a computer where another endpoint is already running. For example, you cannot run the Web-Based endpoint on a computer where you have the endpoint for Windows NT/2000/XP installed and running. When you attempt to download it, you'll receive an error message.

Correspondingly, if the Web-Based endpoint is running on a computer, you can install one of the conventional endpoints on that computer, but the conventional endpoint will not start running automatically once the installation completes. If the Web-Based endpoint executable is running, you must therefore either stop it before installing another endpoint, or else restart the computer after you complete the installation. The new endpoint will then start running automatically (and the Web-Based endpoint will no longer be present).

## Stopping the Web-Based Endpoint

The Web-Based endpoint stops automatically as soon as you restart your computer or log out. However, you can also stop the endpoint manually.

**To stop the Web-Based endpoint:**

1. Click **Ctrl+Alt+Delete** to access the Windows Task Manager.
2. Click the **Processes** tab. In Windows 98, you'll already see the list of processes.
3. Highlight the process `endpoint.exe`. Click **End Process** to stop the endpoint and remove it from your computer.

# Index

## Numerics

10/100/1000 TXS4 3-2

1000 SFPS4 3-2

1600T 3-1, 3-3

250 3-1

400T 3-1

## A

add multiple interfaces 3-16

AIX 6-1

ALLOW keyword 2-2

ALM1000T8 3-2

APPC LU aliases 11-7
  for unattended installation 11-7

application monitoring support 11-5, 11-6, 11-16

archived endpoints 1-11

AUDIT_FILENAME keyword 2-4

automatic upgrade
  self-install 2-5

## C

calculating memory requirements 1-9

capacities of endpoints 1-10

Chassis and Port Configuration tab 3-11

chassis base address 3-14

Chassis Chain 3-11

chassis name 3-11

CheckPoint VPN client
  known issue with application monitoring support 11-16

cleanup after errors 5-8
  HP-UX endpoint 5-8
  IBM AIX endpoint 6-7
  Linux (Cobalt) endpoint 7-7
  Linux (RPM) endpoint 7-4, 9-2
  Sun Solaris endpoint 13-10

clear all ownership 3-15

clear my ownership 3-15

CMPFILES directory 11-7
  HP-UX 5-4
  IBM AIX 6-4
  Linux 7-10, 9-2
  Sun Solaris 13-5
  Windows NT/2000/XP 11-7

Cobalt 7-11
  determining IP network address 7-11
  disabling automatic startup 7-15
  endpoint installation for Linux 7-4
  logging 7-15, 9-5
  support 7-16, 9-5

Cobalt endpoint
  install limitation 7-5

Communications Server
  for Windows NT and 2000 11-9

create desktop shortcut 3-10

## D

DNS 3-5

## E

ENABLE_PROTOCOL keyword 2-5

endpoint capabilities
  IxChariot 1-6

endpoint capacities 1-10

endpoint initialization file 2-1
  default keywords 2-1

endpoint versions 1-11

endpoint.aud 2-4

endpoint.console 5-7, 6-6

endpoint.ini 2-1
  ALLOW keyword 2-2
  AUDIT_FILENAME keyword 2-4
  ENABLE_PROTOCOL keyword 2-5
  SECURITY_AUDITING keyword 2-3

endpoint.log 6-8

endpoints
  installing with SMS 4-1
  uninstalling with SMS 4-3

error messages
  Windows CE endpoint 10-5

F

failed assertion
  HP-UX endpoint 5-9
  IBM AIX endpoint 6-8
  Linux endpoint 7-15, 9-5
  Sun Solaris endpoint 13-10
  Windows NT/2000/XP endpoint 11-16

filters 3-5, 3-21

Filters tab 3-21

G

gsendw32.exe 2-5

H

hardware performance pair
  port numbers 3-24

hardware requirements 1-1

HP-UX endpoint 5-1
  cleanup after errors 5-8
  configuring 5-6
  core dump 5-9
  determining IP network address 5-6
  disabling automatic startup 5-8
  installing 5-1, 5-2, 5-4
  messages 5-8, 5-9
  running 5-7
  safestore files 5-5
  starting 5-7
  stopping 5-8
  support 5-10
  TCP/IP 5-6, 5-7
  unattended installation 5-4
  uninstall 5-5

I

IBM AIX endpoint 6-1
  cleanup 6-7
  configuring 6-5, 6-6, 7-12, 9-3
  determining if active 6-7
  disabling automatic startup 6-8
  installing 6-1, 6-4
  messages 6-8
  README 6-4
  removing 6-5
  running 6-6
  starting 6-6
  stopping 6-7
  support 6-8
  TCP 6-5

IBM CommServer
  for Windows NT and 2000 11-9

IBM MVS endpoint
  TCP 1-4

IBM PCOMM
  for Windows NT and 2000 11-9

ICMP Types 3-22

installation key 7-10

installation requirements 11-2
  HP-UX endpoint 5-1
  IBM AIX endpoint 6-1
  Linux endpoint 7-2, 9-1
  Linux IA-64 endpoint 8-1
  Sun Solaris endpoint 13-1
  Windows CE endpoint 10-1
  Windows NT/2000/XP endpoint 11-2
  Windows XP 64-bit endpoint 12-1

installing
  HP-UX endpoint 5-2
  IBM AIX endpoints 6-1
  Linux endpoint 7-2, 7-3, 9-1
  Sun Solaris endpoints 13-2
  Windows CE endpoint 10-2
  Windows NT endpoint 11-8
  Windows NT/2000 11-3

installing endpoints using SMS 4-1

interface addresses 3-16

interfaces 3-16

IP address 3-5

IP addresses 3-8

IP Protocols 3-22

IPv6 Test Module
  support for 1-7

IxApplifier 3-5

Ixia chassis 3-11

Ixia IxChariot endpoint software 3-3

Ixia IxChariot Management Station 3-2

IxOS 3-3

L

Linux endpoint 7-1, 9-1
  cleanup 7-14, 9-4
  configuring 7-11, 7-12, 9-2, 9-3
  determining if active 7-15, 9-4
  endpoint.ini 7-10, 9-2
  installing 7-2, 7-4, 7-8, 7-10, 9-1, 9-2
  logging 3-27, 7-15, 9-5
  running 7-13, 7-15, 7-16, 9-3
  starting 7-13
  stopping 7-14, 9-4
  support 7-16, 9-5
  supported processors 7-1, 9-1
  uninstall 7-4, 7-7, 9-2
  upgrading for Cobalt 7-4

Linux IA-64 endpoint 8-1
  autostarting 8-6
  cleanup 8-7
  configuring 8-5
  disabling automatic startup 8-7
  increasing number concurrent connections 8-8
  installation requirements 8-1
  installing 8-2, 8-3, 8-4
  logging and messages 8-8
  running 3-27, 8-6
  starting 8-6
  stopping 8-7
  support 8-8
  uninstalling 8-4
  updates 8-8

LM100TXS8 3-2

login dialog 3-9

loopback
  Windows NT APPC 11-10

M

MAC address 3-5

MAC types 3-22

management address 3-14

management network 3-4

merge filters 3-22

messages
  HP-UX endpoint 5-9
  IBM AIX endpoint 6-8
  Linux endpoint 7-15, 9-5
  Sun Solaris endpoint 13-10
  Windows NT/2000/XP endpoint 11-16

Microsoft overlapped I/O 1-8

Microsoft SNA Server 11-17
  for Windows NT 11-2, 11-9

O

ownership 3-9, 3-15

P

Personal Communications
  for Windows NT and 2000 11-9

PFS 5-2

PKGADD command (Solaris) 13-4

port management address 3-14

port management network address 3-14

port ownership 3-9

Port Routes tab 3-20

Portable File System 5-2

R

Red Hat 7-12
  autostart commands 7-12
  disabling automatic startup 7-15
  logging and messages 7-15, 9-5
  Web site 7-16, 9-5

requirements 1-8

reset existing filters 3-22

response file 2-6, 4-1

restricting access to endpoints 2-2

retrieve settings 3-10

RISC System 6-1

routing 3-20

routing table 3-5, 3-20

RPM 7-11, 9-3
  determining IP network address 7-11, 9-3
  endpoint installation for Linux 7-3
  removing endpoint 7-4, 9-2

S

save settings 3-10

SECURITY_AUDITING keyword 2-3

select Ixia ports 3-11

SetAddr utility 11-14, 12-8

setup.iss file 4-1
  Windows NT 11-7

shortcut 3-10

Slackware 7-12
  autostart commands 7-12
  logging and messages 7-15, 9-5
  Web site 7-16, 9-5

SMS installation 4-1
  Windows NT endpoint 11-8

SNA Server 11-17
  configuring for Windows NT 11-9
  for Windows NT 11-2

software requirements
  protocol support 1-1

SPX II
support on Windows NT 11-2
Starting and Stopping Ixia Endpoints 3-27
Sun Solaris endpoint 13-1, 13-10
cleanup 13-10
configuring 13-6, 13-8
core dump 13-10
determining if active 13-10
installing 13-1, 13-2, 13-4, 13-5
IP network address 13-8
messages 13-10, 13-11
OS lockup 13-10
removing 13-6
running 13-9, 13-10
starting 13-9
stopping 13-9
support for OS 5-10
HP-UX 5-10
IBM AIX 6-8
Linux 7-16, 9-5
Sun Solaris 13-11
Windows NT/2000/XP 11-17
Windows XP 64-bit 12-10
Systems Management Server (SMS) 4-1

T

TAR
determining IP network address 7-11, 9-3
endpoint installation for Linux 7-8
unattended Linux install 7-10
TCP Ports 3-22
TCPaccess
versions supported for MVS endpoint 1-4
TcpWindowSize 1-8
test network 3-4
throughput 1-8
generating maximum 1-8
two networks 3-4

U

UDP Ports 3-22
uninstall
HP-UX endpoint 5-5
IBM AIX 6-5
Linux (Cobalt) endpoint 7-7
Linux (RPM) endpoint 7-4, 9-2
Linux (TAR) 7-10
Sun Solaris 13-6
via SMS 4-3
Windows CE endpoint 10-3
Windows NT/2000/XP 11-8

V

version
of endpoint 1-11
virtual addresses in Windows 11-14, 12-8
VLAN 3-5
VoIP Test Module
support for 1-6

W

Web-Based endpoint 14-1
compatibility with other endpoints 14-3
downloading 14-2
error handling 14-2
features not supported 14-1
restarting 14-2
running 14-2
running from location 14-2
saving 14-2
supported OSs 14-1
Windows 2000 3-1
Windows 3.1 endpoint
TCP/IP stacks 1-4
Windows CE .NET operating system 10-1
Windows CE endpoint 10-1
error messages 10-5
installation prerequisites 10-1
installing 10-2
IP address 10-3
limitations 10-4
running 10-4
Sockets port number 10-4
starting 10-5
stopping 10-5
TCP and UDP 10-3
uninstall 10-3
Windows NT Administrator Authority 11-3
Windows NT/2000/XP Endpoint
IP address 11-12
support 11-17
Windows NT/2000/XP endpoint 11-1
APPC 11-9, 11-10
configuring 11-9
disabling automatic startup 11-15
installing 11-3, 11-7
IPX address 11-11
IPX/SPX 11-10
messages 11-16
running 11-13
SetAddr utility 11-14
starting 11-13
stopping 11-13
support 11-17
support for OS 11-17

TCP/IP 11-12
uninstall 11-8, 11-9
with SPX 11-11

Windows XP 64-bit endpoint
configuring 12-5
directories 12-5
disabling automatic startup 12-8
installation 12-2
messages 12-10
removing 12-5
requirements 12-1
running 12-7
SetAddr utility 12-8
starting 12-7
stopping 12-8
unattended installation 12-4
updates 12-10
with TCP/IP 12-6