IxChariot® Archived Endpoints

Release 7.10

913-0955 Rev. A
December 2009

Corporate Headquarters	**Ixia Worldwide 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 **Training:** training@ixiacom.com Support: support@ixiacom.com +1 818 595 2599 For the online support form, go to: http://www.ixiacom.com/support/inquiry/
EMEA	**Ixia Europe Limited** One Globeside, Fieldhouse Lane Marlow, SL7 1HZ United Kingdom +44 1628 405750 FAX +44 1628 405790 salesemea@ixiacom.com	**Support:** eurosupport@ixiacom.com +44 1628 405797 For the online support form, go to: http://www.ixiacom.com/support/inquiry/?location=emea
Asia Pacific	**Ixia Pte Ltd** 210 Middle Road #08-01 IOI Plaza Singapore 188994	**Support:** Support-AsiaPac@ixiacom.com +65 6332125 For the online support form, go to: http://www.ixiacom.com/support/inquiry/
Japan	**Ixia KK** Aioi Sampo Shinjuku Building, 16th Floor 3-25-3 Yoyogi Shibuya-Ku Tokyo 151-0053 Japan	**Support:** Support-Japan@ixiacom.com +81 3 5365 4690 For the online support form, go to: http://www.ixiacom.com/support/inquiry/
India	**Ixia Technologies Pvt Ltd** 2nd Floor, 19/1, Vithall Malya Road, Bangalore 560 001 India	**Support:** Support-India@ixiacom.com +91 80 22161000 For the online support form, go to: http://www.ixiacom.com/support/inquiry/?location=india

913-0955 Rev. A
December 14, 2009 12:47 pm

Table of Contents

1 Introduction

About This Book

The *IxChariot Archived Endpoints* online help provides a description of Performance Endpoint software that is in an archived state: that is, it is not longer updated and maintained.

Intended Audience

This book provides information for individuals who are using any of the archived Performance Endpoints.

Conventions

The library uses consistent conventions to help you identify items throughout the documentation. The following table summarizes these conventions.

Table 1-1. Conventions

Convention	Use
Bold	Window and menu items Technical terms, when introduced
Italics	Book and CD-ROM titles Variable names and values Emphasized words
`Fixed Font`	File and folder names Commands and code examples Text you must type Text (output) displayed in the command-line interface
Brackets, such as `[value]`	Optional parameters of a command

Table 1-1. Conventions (Continued)

Braces, such as `{value}`	Required parameters of a command
Logical OR, such as `value1 | value2`	Exclusive parameters. Choose one parameter.

2 Endpoint Initialization File

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 or End2End servers.
- 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 filename of the End2End safestore file.
- 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. Default Contents

ALLOW	**ALL**
SECURITY_AUDITING	NONE
AUDIT_FILENAME	ENDPOINT.AUD
ENABLE_PROTOCOL	ALL
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 "Customizing endpoint.ini for Windows Endpoints" in the *Performance Endpoints* guide for more information.

3 Distributing Endpoints using SMS

Related Topics

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

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.
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 3-1. SETUP Command Parameters

Parameter	Comment
-noinst	No install: create the `setup.iss` file, but don't really install the endpoint right now. This is an 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.

4 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).

Topics in this chapter:

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.

Endpoint Installation for AIX

This section provides instructions for installing the AIX Performance Endpoint.

Performance Endpoint File Name

The name of the AIX Performance Endpoint file is peaix_*Mm*.tar.Z, where *Mm* is the major and minor IxChariot version number; for example *640* for IxChariot release 6.40.

Installation Procedures

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**.

Installation from CD-ROM

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 4-6 for information about your network connections.

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

Installation from the Web

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.

Removing Temporary Files

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 4-6 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 4-2 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 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`

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

IP networks use *network addresses* to forward traffic across a network to a specific device, and they use *port numbers* to deliver traffic to a specific application running on the selected device.

IxChariot uses a designated *management port* to transport test management traffic between the console and the endpoints. The management port is one of the following:

- SPX transport: port 10117
- TCP transport: either port 10115 (the default) or a user-selected port. Use the MANAGEMENT_PORT option in endpoint.ini to select a port for management traffic.)

IxChariot uses other ports for test traffic. If an IxChariot script specifies "`port_number=AUTO`" on the `CONNECT_ACCEPT` command, 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.

Maximum Value for the MSS Option

The IBM AIX Performance Endpoint supports the use of the Transmit Maximum Segment Size (MSS) option in testing. Note, however, that the highest valid MSS value is 1448 (12 bytes lower than that of the other operating systems). If you set the MSS value higher than 1448, the test will report a CHR0208 error.

Refer to "Setting the Transmit MSS Option" in the *IxChariot Scripts Development and Editing Guide* for additional information.

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.

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

5 HP-UX

This chapter explains the installation, configuration, and operation of the Performance Endpoint software for Hewlett-Packard's HP-UX 11.0 or later.

Topics in this chapter:

Note: 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 11.0 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.

Endpoint Installation for HP-UX

Performance Endpoint File Name

The name of the HP-UX Performance Endpoint file is pehpx_*Mm*.tar.Z, where *Mm* is the major and minor IxChariot version number; for example *620* for IxChariot release 6.20.

Installation Procedures

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**.

Installation from CD-ROM

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.

Installation from the Web

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`

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 11.0 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

IP networks use *network addresses* to forward traffic across a network to a specific device, and they use *port numbers* to deliver traffic to a specific application running on the selected device.

IxChariot uses a designated *management port* to transport test management traffic between the console and the endpoints. The management port is one of the following:

- SPX transport: port 10117
- TCP transport: either port 10115 (the default) or a user-selected port. Use the MANAGMENT_PORT endpoint.ini option to select a port for management traffic.

IxChariot uses other ports for test traffic. If an IxChariot script specifies "`port_number=AUTO`" on the `CONNECT_ACCEPT` command, 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 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.

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 5-9).

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 5-8 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 OS/2

This chapter explains the installation, configuration, and operation of the Performance Endpoint software for IBM's OS/2 operating system.

The endpoint for OS/2 has been archived at version 4.0. Therefore, it does not support features first offered in IxChariot 4.1 and higher.

Installation Requirements for OS/2 Endpoints

Here's what you need to run the endpoint program with OS/2:

- A computer capable of running OS/2 well. 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 8 MBytes of random access memory (RAM); 16 MBytes of RAM is recommended.
- 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 4 MBytes of space available.
- IBM OS/2 Warp 4, Warp 3 Connect, or Warp 3. TCP/IP v4.1 for OS/2 is required for 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: http://www.adobe.com/prodindex/acrobat/readstep.html.
- One or more compatible network protocol stacks, as described in *Network Protocol Stacks*.

We strongly recommend that you get up-to-date with the latest OS/2 service levels.

Getting the Latest Fixes and Service Updates on page 6-14 discusses a variety of ways to get the latest corrective service diskettes (CSDs) or Fixpaks.

Network Protocol Stacks

We recommend configuring your networking software--and making sure that it is working correctly--before installing our software.

See the online help for your networking software, and see *Configuring OS/2 Endpoints* on page 6-5 for more assistance.

- **for APPC:**

  Get IBM Communications Server version 5 or 4.1.

  IBM's newest APPC software for OS/2 is found in Communications Server for OS/2 Warp, Version 5. It's an update of Communications Manager/2, with added features like HPR and support for up to 5 adapters. Throughout this guide, we'll refer to any of these products simply as Communications Server/2 or CS/2.

  Be sure to visit IBM's Web site and download the latest CSDs for whatever version of Communications Server or Communications Manager/2 you're using.

- **for IPX and SPX:**

  The IPX/SPX software supplied by IBM in OS/2 Warp is out of date; we do not support it.

  To use the OS/2 endpoint with IPX or SPX, get the latest Client for OS/2 package from Novell. We've tested with version 2.21 of this package. You can download Client for OS/2 package directly from Novell's Web site: http://support.novell.com/products/

  Here are the bugs we've encountered in this package:

  - Our software does not support connections between OS/2 and Windows NT or 2000, using IPX or SPX.
  - The largest IPX datagram that can be sent to an OS/2 endpoint is 537 bytes. Larger sizes cause a trap in file `TLI_SPX.DLL`.
  - Using more than 20 connections with IPX or SPX causes the OS/2 endpoint to lock up. It must be stopped and restarted to use it again.
  - We have also encountered some problems running SPX between NetWare and OS/2 endpoints. For some tests, some of the pairs report that the data received is not the same as the data that was sent. We have determined that there is an error in the Novell stack for OS/2.

  We're pursuing these bugs with Microsoft and Novell.

- **for TCP and UDP:**

  Get TCP/IP for OS/2 version 4.1, OS/2 Warp 4, or OS/2 Warp Connect 3.

  IBM's newest TCP/IP for OS/2, version 4.1, is required for IP Multicast support. It is available for download from the *IBM Software Choice* Web site (see *Updates for IBM TCP/IP for OS/2* on page 6-14). The TCP/IP V4.1 for OS/2 upgrade requires prerequisite software, all available from IBM's Software Choice Web site:

- We have also encountered some problems running SPX between NetWare and OS/2 endpoints. For some tests, some of the pairs report that the data received is not the same as the data that was sent. We have determined that there is an error in the Novell stack for OS/2.
- Netscape Navigator for OS/2 (free)
- Java 1.1.1 (JDK 1.1.4 for OS/2 Warp, free from Web site)

TCP/IP version 3.0 (shipped with OS/2 Warp Connect 3), offers significant improvements in stack performance, as well as reliability and usability improvements, over previous standalone versions. TCP/IP version 4.0 (shipped with OS/2 Warp 4) continues those improvements.

TCP/IP version 4.0 (in OS/2 Warp 4) has a bug that easily causes a trap when running multiple connections. See the Ixia technical support Web site to download IBM's fixed version of file `SO32DLL.DLL`.

Endpoint Installation for OS/2

Following are instructions for installing **from a CD-ROM** and **from the World Wide Web**.

To install the endpoint from the endpoint CD-ROM:

Put the CD-ROM in your CD-ROM drive. Enter the following at an OS/2 command prompt:

```
[drive:]    (change to your CD-ROM drive)
CD\ENDPOINT\OS2
SETUP
```

Now skip to *Completing Installation*.

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

1. Download the endpoint zip file (`os235.zip`) to a temporary directory (such as `c:\Temp`).
2. Unzip the files into the `Temp` directory. If you don't have IBM's unzip utility for OS/2, you can download it from ftp://ftp-os2.cdrom.com/pub/os2/archiver/unzip532.exe.
3. From the directory where you've unzipped the setup files, enter the following at an OS/2 command prompt:

   ```
   SETUP
   ```

Now read *Completing Installation*.

Completing Installation

The first screen lets you select where to install the endpoint. 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 `\NetIQ\Endpoint` on your boot drive.

The endpoint installation installs the common files and then the endpoint.

After installing the files, the installation program creates a program folder for our software. Inside that folder are two icons: **Endpoint** and **Readme**.

Last, you are asked if you want to update your `CONFIG.SYS` and `STARTUP.CMD` files. We recommend that you answer `YES`, letting the installation program update your system. The copying of files is now complete. After completing installation, you need to shut down and restart your computer, so the changes to `CONFIG.SYS` will take effect.

Refer to *Configuring OS/2 Endpoints* on page 6-5 for more information about your network connections.

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

What We Do During Installation

Here's what we do during the installation steps. Let's say you install our software into the directory named `\NetIQ\Endpoint`. It contains the following:

- the executable programs and dynamic link libraries (DLLs);
- the message file used by the endpoint program;
- the `README` file and icons;
- 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 a discussion of the `endpoint.aud` file.

The installation adds a folder to the OS/2 desktop, and inserts the appropriate icons.

After you reboot, you are ready to start creating and running tests at this computer.

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.

Updates to CONFIG.SYS

When you install the endpoint, the installation adds `\NetIQ\Endpoint` (or your equivalent) to the beginning of your `PATH` and `DPATH` environment variables.

It also adds `\NetIQ\Endpoint` (or your equivalent) to your `LIBPATH`, putting our `DLL` directory at the beginning of its list. Changes in `LIBPATH` only take effect after rebooting OS/2, so you are asked to shut down your computer and reboot after installing.

Before making any changes, we copy your existing `CONFIG.SYS` file to `CONFIG.nnn`, where "`nnn`" is the first number we can find that isn't used already.

When running tests, our software creates one or more threads for each connection pair, in addition to the threads created by the underlying network software. The endpoint installation modifies the `THREADS` environment variable in your `CONFIG.SYS`, setting your `THREADS` value to a minimum of 512. Set it higher if you are running tests with many hundreds of connections (the largest possible `THREADS` value in OS/2 Warp 4 is 4095).

Updates to STARTUP.CMD

Our software adds a statement at the beginning of `STARTUP.CMD` to start the endpoint program when OS/2 is started:

```
DETACH [d:]\path\ENDPOINT.EXE
```

where `d:` and `path` are the drive and path where you installed our software.

This statement starts the endpoint program as an OS/2 background process. The endpoint program, when started this way, runs invisibly with no screen I/O.

As with `CONFIG.SYS`, the endpoint's installation first copies your existing `STARTUP.CMD` file to `STARTUP.nnn`, where "nnn" is the first number that isn't used already.

Configuring OS/2 Endpoints

The endpoint program runs as an application, using 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.

1. Determine the network addresses of the computers to be used in tests
2. Select a service quality (for APPC)
3. Verify the network connections

These topics describe how to accomplish these steps for OS/2:

OS/2 Configuration for APPC

This section provides selected information about configuring APPC. If you are new to configuring APPC, start with the guidance provided by the APPC network software you're using, that is, IBM's Communications Server for OS/2 Warp (or the obsolete Communications Manager/2—we use the term *CS/2* throughout to refer to either IBM product).

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. The URL is: http://www.redbooks.ibm.com/pubs/pdfs/redbooks/gg244485.pdf

Determining the APPC Network Address

This section describes the steps for setting up APPC at the endpoint, for use with IBM's Communications Server for OS/2 Warp.

You can easily determine the LU name of any computer running Communications Server. At an OS/2 command prompt, enter:

```
DISPLAY -SNA
```

This instructs CS/2 to list information about SNA definitions at that computer. You should get something like this:

```
*       SNA Global Information       *
Network name                                NETIQ
Control point (CP) name                     SJOYCE
Physical unit (PU) name                     SJOYCE
Node ID (for XID)                           X'05D00000'
CP alias                                    MYLU
Node type                                   End node
CP local address                            Not used
(independent LU)
Workstation serial number                   00-0000000
Machine type                                0000
Machine model number                        000
Communications Manager version              1.2
```

A fully-qualified LU name is the easiest network address to use with our software. It is constructed by concatenating the network name, a period, and a control point (CP) or LU name. Although you can define multiple LUs at one computer, one always serves the role of the control point LU. This CP name is returned by the `DISPLAY` command you entered. In the above example, the fully-qualified LU name is "`NETIQ.SJOYCE`," constructed by appending the CP name to the network name. You will need to know the LU names of each endpoint you use for APPC testing.

Selecting a Service Quality (APPC Mode Name)

Most networking protocols have some mechanism to allow applications to tell the network what kind of service it requires. APPC does this through the mode definition. Several modes come preconfigured on most APPC products. These include the following:

Table 6-1. Selecting a Service Quality

Mode Name	Description
#INTER	Interactive data, high priority, no security
#INTERSC	Interactive data, high priority, secure connections only
#BATCH	Batch data, low priority, no security
#BATCHSC	Batch data, low priority, secure connections only

For many tests, these modes are sufficient. However, if you are trying to emulate a particular APPC application, you should select the same mode name that it uses.

These pre-defined modes are defined with session limits of 8. This means that you can only have 8 APPC sessions at a time, between a pair of computers, using the same mode name. If you're attempting to run more than 8 sessions using the same mode between a pair of computers, we recommend creating a new mode on both computers, with a session limit larger than 8.

Reaching APPC Session Limits

There is a limit on the number of LU 6.2 sessions that can use an APPC mode at the same time, between a single pair of computers. When you reach this limit, attempts to start new sessions will fail, with the sense data value `X'FFFE0016'`.

Using Secure Modes

If you use modes that are defined as "secure," such as `#BATCHSC` or `#INTERSC`, you can only get a connection over links that are defined as secure. CS/2's links default to "non-secure," and can't be used for secure sessions.

Use secure modes only when you know that your network is configured to provide that level of service.

Using APPC Compression with CS/2

In OS/2, some APPC modes are defined so that they compress the data being sent and received. This compression requires a lot of CPU processing and can affect your performance results. To tell whether a mode uses compression, enter the following at an OS/2 command prompt:

```
DISPLAY -MD
```

Find the mode in question, and look for a line that says "Compression need," with a value of Requested or Prohibited.

Trying Out 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 that is included with most APPC software packages. 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, enter the following at a command prompt:

```
APING lu_name -M mode_name -N
```

Substitute the lu_name and mode_name with the actual names you determined in the steps above. (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.

Although APING is packaged with Windows NT SNA Server and OS/2 Communications Manager, it is not automatically configured. If you run APING and get a CPI-C return code of CM_TP_NOT_AVAILABLE or CM_TP_NOT_RECOGNIZED, it means that APING is not configured on the other computer. The good news is that you did get a connection to the other computer, which means that our software should be able to use this endpoint for APPC testing.

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.

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.

OS/2 Configuration for IPX and SPX

For OS/2 and NetWare, our software makes calls to the TLI API when using IPX or SPX.

To use the IPX or SPX protocol, IPX addresses must be supplied as the network address at the IxChariot or Qcheck Console 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). The network number and node ID are separated by a colon. 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.

If you already know the IP address of a computer—and thus can Ping that computer—it's easy find its MAC address. First, Ping to the target computer, 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.

If the IPX/SPX protocol stack is configured and started, the endpoint shows the local IPX address when it starts. You may need to briefly comment out the `DETACH` command in your OS/2 `STARTUP.CMD` file, and start the endpoint manually to see this address. You'll probably want to run the endpoint `DETACH`ed, though, in normal operation.

Alternatively, to determine an OS/2 computer's local IPX address, view the `LANTRAN.LOG` file. This ASCII file is recreated each time the computer is rebooted; by default, it's placed in directory `C:\IBMCOM`.

This provides the computer's 6-byte node ID in a line that looks like the following:

```
Adapter 0 is using node address 0207011A3082...
```

You can ask your Novell NetWare administrator for your current 4-byte network number. They can load the `MONITOR` program at the server and look in the "Connection Information" section, under the entry for the OS/2 computer (presuming it's connected to the NetWare server). You'll see an address that looks like this: `00000002:0207011A3082:0004`. The full address is everything preceding the second colon.

In IxChariot, it's tedious to enter IPX addresses when adding new endpoint pairs. When using the IPX or SPX protocol, 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.

Sockets Port Number

IPX and SPX applications use their network address (as described above) to decide which computer to connect to in a network. They use a *port number* to decide which application program to connect to within a computer.

The IPX/SPX sockets port for endpoints is **10117**. 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.

OS/2 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

You can find a computer's local IP address in OS/2 by running the **TCP/IP Configuration** program. Here's how to run that program:

1. Double-click on the **OS/2 System** folder.
2. Double-click on the **TCP/IP** folder.
3. Double-click on the **TCP/IP Configuration** icon.

Alternatively, you can enter the following at an OS/2 command prompt:

```
TCPCFG
```

The `NETSTAT` command lets you determine *active* IP connections (that is, `NETSTAT` isn't helpful if the TCP/IP software itself isn't loaded and running). To determine a computer's local IP address, enter the following OS/2 command:

```
NETSTAT -a
```

This prints out a line for each active interface. The local IP address should be in the second column of each row.

To determine the domain name, enter the following OS/2 command, substituting the IP address for the x's.

```
HOST xx.xx.xx.xx
```

If the `HOST` command cannot resolve the IP address into a domain name, avoid using domain names as network addresses; use numeric IP addresses instead.

There may be more than one `HOSTS` file on your computer. You can find which is being used by looking at the `ETC` environment variable. For example, enter:

```
SET ETC
```

to find the path to the directory containing your HOSTS file. The default location for the /etc/hosts file on OS/2 is:

```
d:\MPTN\ETC\HOSTS
```

where d: is the drive where you installed OS/2.

Be sure there is a "newline" or "end-of-line" character on the last line of your HOSTS file in OS/2. Without a newline there, it appears that TCP/IP in OS/2 ignores the last line.

Trying Out the TCP/IP Connection

Ping is a simple utility program included in all TCP/IP implementations. To try out the connection from OS/2 to another computer, enter the following:

```
ping xx.xx.xx.xx 100 1
```

Replace the x's with the IP address of the target computer, that is, the computer you're trying to reach. The final two parameters tell OS/2 to send a 100-byte packet, 1 time (otherwise, its default is to send 56-byte packets continuously). On OS/2, if the Ping returns a message that says, "1 packet(s) received," the Ping worked. If it hangs for an extended period or says, "0 packet(s) received," the Ping failed, and you may have a configuration problem, a network problem, or the target computer may not even be powered on.

For more details about the Ping command, enter:

```
ping -?
```

If you're unable to reach the target computer using Ping, OS/2's TRACERTE command may help you determine how far packets can get through the network. The TRACERTE command tries to find whether each hop in the IP network can be reached, on the way to the target computer. Be aware that TRACERTE's results aren't necessarily repeatable, because a different route can be taken by each packet sent.

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 (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 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 OS/2 Endpoints

The following sections describe starting and stopping an endpoint, as well as some of the messages and information that become available during testing with this endpoint.

Starting an OS/2 Endpoint

By default, the endpoint installation program updates your `STARTUP.CMD` file. This update causes `ENDPOINT.EXE` to be started automatically when OS/2 is started. `ENDPOINT.EXE` will run as a detached OS/2 process.

Alternatively, you can run `ENDPOINT.EXE` in an OS/2 window. This lets you watch the endpoint as it runs, with only a tiny performance penalty. To do this, change the line in your `STARTUP.CMD` file that starts the endpoint, from

```
DETACH ENDPOINT.EXE
```

to

```
START ENDPOINT.EXE
```

You should see a message like the following at an endpoint when it is active and waiting to run a test:

```
Endpoint, Version 4.0
Copyright NetIQ Corp., 1995-2000.
Build level: xxx

Processing INI file (d:\Ganymede\Endpoint\endpoint.ini).
Endpoint INI information in use:

All available protocols are enabled.
All consoles may run tests on this endpoint.
Security Auditing: NONE
Audit filename: d:\Ganymede\Endpoint\endpoint.aud

Support for APPC has been started.
Support for SPX has been started.
Support for TCP has been started.
```

The order in which the protocol support messages appear may differ, depending on which network protocol completes initialization first. “Support for TCP” means that the endpoint is ready to run tests that use the RTP, TCP, and UDP protocols; “Support for SPX” means that it’s ready for both IPX and SPX tests. 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.

If you stop `ENDPOINT.EXE` and need to restart it without restarting OS/2, you may either start the program from the endpoint icon in the folder for our software or enter

```
ENDPOINT
```

at an OS/2 command prompt.

A single running copy of ENDPOINT.EXE can handle one or multiple concurrent tests. If the endpoint program is already running and you try to start another copy, you get the following message, "**CHR0183**: The endpoint program is already running. Only one copy is allowed at a time."

Stopping an OS/2 Endpoint

Like all applications involved with networks and communications, endpoint programs can sometimes get "hung," with no way to terminate them. The endpoint program in OS/2 has a special command-line option, /K or -K. If you have an endpoint program you'd like to kill, go to another OS/2 command prompt on the same computer and enter:

```
ENDPOINT /K
```

This special option has the sole purpose of killing any endpoint program running on that computer. This is especially helpful for terminating an endpoint program running as a detached process, which you can't see in the OS/2 Window List.

Use the **Ctrl+Break** or **Ctrl+C** key combinations to stop a visible running endpoint program (that is, one that was not started with DETACH). The program will ask you if you really want to stop. Enter "y" to stop the program; enter "n" to keep running.

Disable Your Screen Saver

Screen savers in OS/2 can significantly lower the throughput that's measured by an endpoint. We recommend disabling your screen saver at endpoint computers while running tests.

How to Tell If an OS/2 Endpoint Is Active

If the OS/2 endpoint is running in a detached process, you can see if it's active by trying to start it again. At an OS/2 command prompt, go to the directory where you installed our software. Enter ENDPOINT. If the endpoint is already running, message **CHR0183** is returned.

Disabling Automatic Startup

To disable the automatic starting of the endpoint, edit the your STARTUP.CMD file. You should see a line like the following:

```
DETACH ENDPOINT.EXE
```

Comment out this line (using REM). To start the endpoint at any time, click on the endpoint icon in the folder for our software.

Logging and Messages

If you are running ENDPOINT.EXE in an OS/2 window, you will see information about the running tests and error messages (when appropriate). 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 called ENDPOINT.LOG. To view an error log, use the command-line program named FMTLOG. FMTLOG 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 directory where you installed the endpoint.

If `ENDPOINT.EXE` was started from `STARTUP.CMD`, error log files are written to the root directory of the boot drive. If `ENDPOINT.EXE` was started from an icon, the log files go to the same directory as `ENDPOINT.EXE`.

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.

See the Technical Support section of the Ixia Web site for links to the latest software updates.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 the OS/2 software used by the endpoints.

Updates for IBM OS/2

The IBM Web site http://www.software.ibm.com/network/pcomm/support/ has links to the latest Fixpaks and driver updates. OS/2 Warp 4 Fixpak 10 was released in February 1999.

Updates for IBM Communications Server and Communications Manager/2

The IBM Web site http://www-4.ibm.com/software/network/commserver/support/fixes/csos2.html has the latest fixes.

Updates for IBM TCP/IP for OS/2

We're testing with IBM's TCP/IP for OS/2, version 4.1 530. This is newer than the TCP/IP stack shipped in OS/2 Warp 4. It's available for download from the IBM Software Choice Web site:

http://service.software.ibm.com/asd-bin/doc/index.htm.

For more information about IBM Software Choice, call 800-513-7043 in the USA.

Updates for Novell Client Software

Novell posts code and driver updates directly to their Web site: http://support.novell.com/products/.

7 Microsoft Windows 3.1

The following topics explain the installation, configuration, and operation of the Performance Endpoint software for Microsoft Windows 3.1. These steps also apply to Windows for Workgroups 3.11. For simplicity, we refer to both of these as Windows 3.1 throughout.

The Windows 3.1 endpoint, without true multitasking, can perform up to 50% worse in the role of Endpoint 1 than as Endpoint 2. We recommend using it only as Endpoint 2, when possible.

The Windows 3.1 endpoint is archived at version 3.5. Therefore, it will not support the latest functionality offered in recent releases of IxChariot. Other Windows endpoints are available in later versions, including the Windows 95, Windows 98, Windows Me, Windows NT/2000/XP, and Windows XP (64-bit) endpoints.

Our documentation uses the generic term “Windows” to refer to Win32 operating systems. Other Windows operating systems are referred to by name.

Installation Requirements for Windows 3.1 Endpoints

Here’s what you need to run the endpoint program with Microsoft Windows 3.1:

- A computer capable of running Windows 3.1 in Enhanced mode. This implies a CPU such as an Intel 80386, 80486, a member of the Pentium family, or equivalent. An x486 or better is recommended.
- 8 MBytes of random access memory (RAM). Your total RAM requirement depends on the RAM usage of your underlying TCP/IP protocol stack, and the number of active DOS and Windows applications.
- A hard disk with at least 4 MBytes of space available.
- Microsoft Windows 3.1, 3.11, or Windows for Workgroups 3.11. The Chameleon TCP/IP protocol stack version 7.0 supports Endpoint 2 as an IP Multicast receiver. Chameleon 7.0 is the only Windows 3.X TCP/IP protocol stack we have found that supports IP Multicast. We have tested extensively

with it. Because of the lack of thread support, you cannot use Endpoint 1 as an IP Multicast sender.

- 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: http://www.adobe.com/prodindex/acrobat/readstep.html.

- A TCP/IP protocol stack that is installed, configured, and loaded.

  This endpoint uses TCP/IP by making calls to the WinSock version 1.1 programming interface. We have tested with the TCP/IP protocol stacks listed in the `README` file. Aside from limitations in the implementations of the protocol stacks, the endpoint for Windows 3.1 should work with any TCP/IP stack that offers an API compliant with WinSock 1.1.

  We recommend configuring your TCP/IP networking support--and making sure that it is working correctly--before installing our software.

See the online help for your TCP/IP networking software, and see *Configuration for TCP/IP* on page 7-6 for more assistance.

Endpoint Installation for Windows 3.1

You can install from CD-ROM, diskettes, or from the World Wide Web:

- **from CD-ROM:**

  Put the CD-ROM in your CD-ROM drive. From **File** on the Program Manager or File Manager menu, select **Run**. The Run dialog box asks you to type the name of a program; enter the following in the **Command Line:** field:

  ```
  [drive:]\endpoint\win31\setup.exe
  ```

- **from diskette:**

  Put the first installation diskette in your diskette drive. From **File** on the Program Manager or File Manager menu, select **Run**. The Run dialog box asks you to type the name of a program; enter the following in the **Command Line** field:

  ```
  [drive:]\setup.exe
  ```

  where "`[drive:]`" is your 3.5 inch diskette drive.

- **from the World Wide Web:**

  Save the `win3135.zip` file for the endpoint to a local directory, such as `Temp`. Use the File Manager to navigate to the zip file and unzip it. Then double-click on the icon for `Setup.exe`.

  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 `\Program Files\NetIQ\Endpoint` on your boot drive.

  The endpoint installation next goes through the following steps:

1. Installing the endpoint;
2. Creating a Program folder with **Endpoint** and **Readme** icons;
3. Updating `win.ini` to automatically start the endpoint, if you choose;
4. Showing the `README` file, if you choose.

The copying of files is now complete; you can remove the CD-ROM or diskette from its drive.

The endpoint program is not running when the installation completes. You can either restart Windows, or start the endpoint program manually by double-clicking on the new folder, then double-clicking on the endpoint icon. By default, the Windows 3.1 endpoint starts minimized, that is, as an icon.

You should see a message like the following when the endpoint is active and waiting to run a test:

```
Endpoint, Version 3.5
Copyright NetIQ Corporation, 1995-2000.
Build level: xxx

Processing INI file (d:\Ganymede\Endpoint\endpoint.ini).
Endpoint INI information in use:

All available protocols are enabled.
All consoles may run tests on this endpoint.
Security Auditing: NONE
Audit filename: d:\Ganymede\Endpoint\endpoint.aud

Support for TCP has been started.
```

"Support for TCP" means that the endpoint is ready to run tests that use the RTP, TCP, and UDP protocols. Endpoints only listen for incoming tests on connection-oriented protocols, like TCP. Datagram tests are set up and results are returned using their associated connection-oriented protocol; thus, UDP tests are set up using TCP, and IPX tests are set up using SPX.

When you've completed installation, refer to *Configuring Windows 3.1 Endpoints* on page 7-6 to make sure your endpoint is ready to be used in testing and monitoring.

Unattended Installation for Windows 3.1

Unattended installation (also called *silent installation*) is available for the endpoints for Windows 3.1 and 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.

When installing, specify the "`-r`" option on `SETUP` to save your input. For example, to install from diskette the first time, enter the following:

```
[drive:]\SETUP -r
```

where "[drive:]" indicates your 3.5 inch diskette drive. This produces the answer file named setup.iss, which can then be used on subsequent silent installations. The setup.iss answer file is created in your Windows directory.

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\NetIQ\Endpoint directory on our n: LAN drive:

```
setup -s -f1n:\Ganymede\Endpoint\setup.iss
```

Don't mix the .iss files among 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 setup.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.

Installing the Windows 3.1 Endpoint with SMS

See "Distributing Endpoints Using SMS" in the *Performance Endpoints* guide for information on automatically installing (and uninstalling) endpoints, using Microsoft's Systems Management Server (SMS).

What We Do During Installation

Here's what happens during the installation steps. Let's say you install the endpoint software into the directory \Ganymede\Endpoint. A directory is created with the following contents:

- the executable programs;
- the README file and icons;
- 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 process for the Windows 3.1 endpoint program makes no changes to CONFIG.SYS or AUTOEXEC.BAT.

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 following command:

   ```
   delete *.q*.
   ```

3. Uninstall the current endpoint.
4. Install the desired endpoint.

Windows Resources Consumed by the Endpoint

There are three resources important in the resource management of Windows 3.1: RAM, USER, and GDI.

Table 7-1. Windows Resource Consumption

Resource	Description
RAM	This is the amount of virtual RAM available to Windows. Let's say you have 16 MBytes of RAM installed in a computer, with a 1 MByte RAM disk carved from that. For this example, you also have 20 MBytes of swap space on the disk, reserved for Windows. The total RAM that's managed by Windows is 35 MBytes (that is, (16 - 1) + 20). A negligible amount is consumed from the RAM resource when an endpoint is started. Little measurable RAM is consumed when running tests with long connections. However, we've seen up to 2.6 MBytes consumed when running TCP tests with short connections. Almost all of this memory is consumed by the TCP/IP stack. Long and short connections are discussed in the Application Scripts guide. RAM is also consumed when using named .CMP files for the datatype parameter of SEND script commands. The entire CMP file is loaded into RAM, for use on SEND commands. The largest .CMP file shipped with our software is about 800 KBytes. User files can be up to 1,000 KBytes
USER	A 64 KByte resource, with handles to windows and menu objects. When the endpoint is started, it consumes about 2 KBytes from the USER resource. Nothing additional is consumed when tests are run.
GDI	A 64 KByte resource, with handles to brushes, pens, bitmaps, and so on. Nothing is consumed from the GDI resource when an endpoint is started or a test is run

Removing the Endpoint Package (Uninstall)

If you need to remove the endpoint package from your hard disk, take these steps:

1. Click on the **Control Panel** icon.
2. Click the **Add/Remove Programs** icon. The Add/Remove Programs Properties is shown.
3. Highlight **NetIQ Endpoint** 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 the endpoint. For detailed instructions on manually removing the endpoints, see our Web site at http://www.ixiacom.com/support/ixchariot.

Configuring Windows 3.1 Endpoints

The endpoint program runs as a 16-bit application in Windows 3.1, using the WinSock version 1.1 programming interface for all of its TCP and UDP 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.

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

Let's look at TCP 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

This information may vary based on a particular vendor's TCP/IP implementation of the programs described here.

The `CONFIG` or `IPCONFIG` program, shipped with many TCP/IP protocol stacks, lets you determine a Windows 3.1 computer's IP address. It's generally used by typing the following at an MS-DOS command prompt:

```
CONFIG
```

or

```
IPCONFIG
```

If this command is available on your system, it returns your local IP address.

You may also be able to find your IP address using the protocol's graphical user interface. See the manual for the stack to find more information.

Some TCP/IP protocol stacks also offer the `NETSTAT` program. The "`-n`" command line option will often show you the local IP address of a computer, if any connections are active:

```
NETSTAT -N
```

Trying Out 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.

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 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.

Using RTP or UDP with the Chameleon TCP/IP Stack

Message CHR0216 indicates a datagram timeout, and can occur because of a bug in the NetManage Inc. Chameleon stack when the following conditions hold true:

- a Chameleon TCP/IP stack is installed on the Windows 3.1 endpoint computer,
- the network protocol specified is RTP or UDP, and
- the `send_buffer_size` specified in the application script is more than 1490 bytes.

The workaround is to either modify the script to send less than 1490 bytes at a time or use a different TCP/IP stack. We've seen this problem with versions 4.6.5, 4.6.6, and 7.0 of the Chameleon TCP/IP stack.

Running Windows 3.1 Endpoints

The following sections describe starting and stopping an endpoint, as well as the messages and information that are available for problem determination.

Using Our Software with Windows 3.1

Only one connection pair at a time can use a Windows 3.1 endpoint. If you try to test with multiple connections using the same Windows 3.1 endpoint program, the error message **CHR0180** is returned when the test is run. If you try to test with Endpoint 1 and Endpoint 2 as the same Windows 3.1 computer (that is, a test in loopback), you'll see error message **CHR0182**.

The Windows 3.1 endpoint, without true multitasking, can perform up to 50% worse in the role of Endpoint 1 than as Endpoint 2. We recommend using it only as Endpoint 2, when possible.

Starting a Windows 3.1 Endpoint

If you so chose during installation, `WIN.INI` was updated to start the endpoint program. This means the program `endpoint.exe` is started automatically when Windows 3.1 is started. By default, the endpoint program starts as an icon.

Anytime the endpoint program is not running, you can start it by double-clicking on the folder where you've saved it, then double-clicking on the Endpoint icon. Again, the endpoint program starts, by default, as an icon.

A single running copy of `endpoint.exe` handles only one connection. If the endpoint program is already running and you try starting another copy, you get the following error message: "**CHR0183**: `The endpoint program is already running. Only one copy is allowed at a time.`"

Stopping a Windows 3.1 Endpoint

Stop the endpoint program by double-clicking in the upper left-hand corner, or by using the Windows Close key combination: **Alt+F4**.

We've observed some odd behavior with some Windows 3.1 protocol stacks. Stop the endpoint, then attempt to start it again. You see the following error message at the endpoint computer: "`CHR0206: The TCP port number is already in use.`" This message occurs because the protocol stack did not clean up its resources correctly. The workaround is a reboot of the computer; merely restarting Windows does not clean up the condition. We've identified the protocol stacks with this problem in the `README` file for the Windows 3.1 endpoint.

Disable Your Screen Saver

Screen savers in Windows 3.1 can significantly lower the throughput that's measured by an endpoint. We recommend disabling your screen saver at endpoint computers while running tests.

Disabling Automatic Startup

To disable automatic startup, edit your `WIN.INI` file. In the Windows section, comment out the following line:

```
load: ganymede\endpoint\endpoint.exe
```

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 our software. To view an error log, use the program named `FMTLOG`. The version of `FMTLOG` shipped with the Windows 3.1 endpoint runs as a command from an MS-DOS prompt. `FMTLOG` reads from a binary log file, and writes its formatted output to `stdout`. Use the following `FMTLOG` command:

```
FMTLOG \ganymede\endpoint\endpoint.log >
your_output_filename
```

In addition, the endpoint code does a lot of internal checking on itself. You may see a "Failed assertion" message in the endpoint window. If you encounter an assertion failure, please write down the sequence of things you were doing when it occurred. Our software captures details related to the problem in an ASCII text file named `ASSERT.ERR` in the directory where you installed the endpoint. Save a copy of the file, and send it back to us via e-mail, for problem determination.

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.

Therefore, we recommend working with the very latest software for the underlying operating system and communications software. Updates for the Windows 3.1 software used by the endpoint program are hard to come by; the following sections offer suggestions.

Updates for TCP/IP Protocol Stacks

Contact the vendor of the Windows 3.1 TCP/IP protocol stack you are using to obtain the latest updates. We have tested with the protocol stacks described in the `README` file that accompanies this endpoint.

Updates for Microsoft Windows 3.1

Microsoft posts code and driver updates directly to their Web site. However, at this writing, Windows 3.1 and Windows for Workgroups don't appear on their Web site anymore.

Go to the Microsoft home page at http://www.microsoft.com to check for information on upgrades.

8 Windows 95

This chapter explains the installation, configuration, and operation of the Performance Endpoint software for Microsoft Windows 95.

The endpoint for Windows 95 is archived at version 4.1. Therefore, it will not support the latest functionality offered in new releases of IxChariot. Endpoints for other Windows 32-bit and 64-bit operating systems are available in later versions, with functionality not offered in the endpoint for Windows 95.

See "Microsoft Windows 98," "Microsoft Windows Me," or "Microsoft Windows NT, Windows 2000, and Windows XP" in the *Performance Endpoints* guide for more information about other Win32 operating systems.

Installation Requirements for Windows 95 Endpoints

Here's what you need to run the endpoint software with Microsoft Windows 95:

- A computer capable of running Windows 95 well. This implies a CPU such as an Intel 80386, 80486, a member of the Pentium family, or equivalent. A Pentium or better is recommended.
- 8 MBytes of random access memory (RAM); 16 MBytes of RAM is recommended.

  The total RAM requirement depends on the RAM usage of the underlying protocol stack and the number of concurrent connection pairs. Since only about 20 connections are possible with a Windows 95 endpoint, additional memory is probably not required.
- A hard disk with at least 4 MBytes of space available.
- Microsoft Windows 95 with the latest service packs applied. We strongly recommend that you get up-to-date with the latest Windows 95 service levels. *Getting the Latest Fixes and Service Updates* on page 8-14 discusses how to get the latest software.
- One or more compatible network protocol stacks, as described in the following section, *Network Protocol Stacks*.

Network Protocol Stacks

We recommend configuring your networking software -- and making sure that it is working correctly -- before installing our software.

See the Help for your networking software, and see *Configuring Windows 95 Endpoints* on page 8-7 for more assistance.

- **for APPC:**

  Use IBM Personal Communications AS/400 and 3270 for Windows 95 version 4.3 (called PCOMM for Windows 95, eNetworks, or SecureWay) -- or later. It is limited in capacity to about 50 sessions at a time.

- **for IPX and SPX:**

  The IPX/SPX protocol stack supplied by Microsoft in release 1.00 of Windows 95 (or Service Pack 1) is not sufficiently robust for use with our software. Thus, we do not support it. Choose one of these options for improved network software for use with IPX and SPX:

  - Use Windows 98 or Windows Me.

    Microsoft significantly improved the network protocol stacks for TCP/IP and IPX/SPX in Windows 98 and Windows Me. For best performance and robustness, use Windows 98 or Windows Me, if possible.

  - Get the *Client 3.2 for Windows 95* package from Novell.

    This is the best solution today for running IPX and SPX on Windows 95 with our software. The software package contains an IPX/SPX protocol stack that replaces the one shipped with Windows 95. It performs well, supporting about 40 concurrent connections. We've tested with version 3.21 of this package. Download this software package directly from Novell: http://support.novell.com/products/.

  - Get Microsoft's "Windows Sockets 2 for Windows 95" stack.

    Microsoft's WinSock 2 stack for Windows 95 improves the protocol support for both IPX/SPX and TCP/UDP. To download this stack, see *Updates for WinSock 2* on page 8-14.

  - Use Microsoft's "Windows 95 OEM Service Release 2" (OSR2).

    Microsoft's latest version of Windows 95, OEM Service Release 2, known as OSR2, contains an improved IPX/SPX protocol stack, which performs pretty well and supports about 8 concurrent connection pairs.

    OSR2 is not separately available. It's only shipped as part of the OEM packaging on new computers. If you have one of these new computers, this is a simple solution (but not as good as Novell's Client 3.2 package or Windows 98 or Me).

  To see if OSR2 is installed, check the System Properties for the version number of Windows 95. If you have version 4.00.950b, OSR2 is installed.

- **for RTP, TCP, and UDP:**

  TCP/IP software is provided as part of the network support in the Windows 95 operating system. For several reasons, Windows 98 and Windows Me offer the preferred TCP/IP stack.

- **Windows 98 and Windows Me** come with Microsoft's latest WinSock 2 protocol stack integrated, which supports IP Multicast, QoS, and about 50 connections.
- **Windows 95** shipped with a TCP/IP stack that is limited in capacity to about 10 connections at a time. It does not support IP Multicast or QoS.

A WinSock 2 stack for Windows 95 is available for download from Microsoft; it supports IP Multicast, but not QoS. To download this stack, see *Updates for WinSock 2* on page 8-14.

Endpoint Installation for Windows 95

Following are instructions for installing the endpoint **from a CD-ROM** or **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. From the Start menu, click **Run**.

   `[drive:]\Endpoint\archive\win95\gsendw95.exe`
3. The Run dialog box asks you to enter the name of a program. Enter the following in the **Open** field:
4. If a previous version of the endpoint is present, you are asked if you want to remove it.
5. The next dialog box lets you select the directory for the endpoint. We recommend installing the endpoint on a local hard disk of the computer you're using (if you install on a LAN drive, the additional network traffic will influence your performance results).

   If you have previously installed these endpoints, the default directory is where you previously installed them. If you have not previously installed endpoints but have installed other NetIQ products, the default directory is the same level as the one where your other products are installed. For example, if you installed IxChariot in `C:\Program Files\NetIQ\Chariot`, the default directory is `C:\Program Files\NetIQ\Endpoint`. If you have not previously installed other NetIQ products, the default drive is the first drive with enough disk space for the endpoints.
6. Next, you are asked whether to install the pre-built data files, which are used by application scripts to generate the payload data in the frames they send; this is important if there are software or hardware in your network that do data compression. These files take a small amount of hard disk space; we recommend always installing them.

   You are also given the option to start the endpoint after it is installed. If you choose **No**, it is started the next time you restart Windows 95. The endpoint installation next copies the necessary files to your hard disk.
7. Finally, you are asked if you want to install application monitoring support. This option is **not** recommended.

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 at startup, take the following steps:

1. Open the Registry edit utility using the `REGEDIT` command.
2. Navigate to `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServices` and open it.
3. Write down the value for the variable **NetIQ Performance Endpoint**, so that you can restart the endpoint as a service later. Delete the NetIQ Performance Endpoint key from the Registry.

   If you want to restore the setting later, you must do so manually.

When you've completed installation, refer to *Configuring Windows 95 Endpoints* on page 8-7 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. Save the endpoint to a directory on a local hard drive.
2. Use Windows Explorer to navigate to the endpoint file, `gsendw95.exe`, and double-click to unzip it and activate Setup. Refer to *Using WinZip* on page 9-5 for instructions.

   If a previous version of the endpoint is present, you are asked whether you want to remove it.
3. The next screen after the Software License Agreement lets you select the directory for the endpoint. We recommend installing the endpoint on a local hard disk of the computer you're using (if you install on a LAN drive, the additional network traffic will influence your performance results).
4. If you have previously installed these endpoints, the default directory is where you previously installed them. If you have not previously installed endpoints but have installed other NetIQ products, the default directory is the same level as the one where your other products are installed. For example, if you installed IxChariot in `C:\Program Files\NetIQ\Chariot`, the default directory is `C:\Program Files\NetIQ\Endpoint`. If you have not previously installed other NetIQ products, the default drive is the first drive with enough disk space for the endpoints.

   If an endpoint is already installed, you will be prompted to remove the previous installation.
5. Next, you are asked whether to install the pre-built data files, which are used by application scripts to generate the payload data in the frames they send; this is important if there are software or hardware in your network that do data compression. These files take a small amount of hard disk space; we recommend always installing them. You are also given the option to start the

endpoint after it is installed. If you choose **No**, it is started when you reboot Windows 95.

6. Finally, you are asked if 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. 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 at startup, take the following steps:

1. Open the Registry edit utility using the command `REGEDIT`.
2. Navigate to `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServices` and open it.
3. Write down the value for the variable NetIQ Performance Endpoint, so that you can restart the endpoint as a service later. Delete the NetIQ Performance Endpoint key from the Registry.

   If you want to restore the setting later, you must do so manually.

When you've completed installation, refer to *Configuring Windows 95 Endpoints* on page 8-7 to make sure your endpoint is ready to be used in testing and monitoring.

Unattended Installation for Windows 95

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 "`gsendw95.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\NetIQ\Endpoint` directory on our n: LAN drive:

```
SETUP -s -f1n:\Program Files\NetIQ\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.

Installing the Windows 95 Endpoint with SMS

See the *Performance Endpoints* guide for information on automatically installing (and uninstalling) endpoints, using Microsoft's Systems Management Server (SMS).

What Happens During Installation

Here's what happens during the installation steps. Let's say you install the endpoint into the directory `\Program Files\NetIQ\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 installation process for a Windows 95 endpoint makes no changes to `CONFIG.SYS` or `AUTOEXEC.BAT`. The installation process does, however, involve adding the endpoint to the Registry so that the endpoint starts automatically when you start a Windows 95 computer.

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

1. Stop the endpoint. See *Stopping a Windows 95 Endpoint* on page 8-13 for instructions.
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.

Removing the Endpoint Package (Uninstall)

If you need 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 the **Add/Remove Programs** icon. The Add/Remove Programs Properties dialog box is shown.
3. Highlight **NetIQ Endpoint** and click **Add/Remove**. The uninstallation program begins. After the program is completed, the endpoint should now be uninstalled.

Removing the Endpoint Manually

If the uninstallation program is unable to uninstall the endpoint, you will need to manually uninstall the endpoint.

For detailed instructions on manually removing the endpoints, see the Performance Endpoints FAQ page in the Knowledge Base on our Web site at http://www.netiq.com/support/pe/default.asp.

Configuring Windows 95 Endpoints

The Performance Endpoint program is a 32-bit application for Windows 95, using the 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.

- Determine the network addresses of the computers to be used in tests
- Verify the network connections

The following topics describe how to accomplish these steps for Windows 95:

- *Windows 95 Configuration for APPC*
- *Windows 95 Configuration for IPX and SPX* on page 8-9
- *Windows 95 Configuration for TCP/IP* on page 8-10

Windows 95 Configuration for APPC

The following sections describe the steps for setting up APPC at the endpoint, for use with IBM's Personal Communications (PCOMM) software for Windows 95. If you are new to configuring APPC, start with the guidance provided by the APPC network software you're using.

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. The URL is: www.redbooks.ibm.com/pubs/pdfs/redbooks/gg244485.pdf.

Determining the APPC Network Address

To determine the fully qualified LU name of any computer running PCOMM, do the following:

1. Start the SNA Node Operations program by either running `pcsnops.exe` from a command prompt or by clicking the icon.

2. If the node is not currently started, select **Operations...Start Node**.
3. The first panel shown should be the Node panel, which shows a value titled FQCP Name. If this is not visible, select **View...Select Resource Attributes** and select it for viewing. A default fully qualified LU name is automatically configured and it has the exact same name as the FQCP Name shown in this panel.

A fully qualified LU name is the easiest network address to use with our software. Although you can define multiple LUs at one computer, the default LU name is the one on which the endpoint "listens" for a connection.

Automatically Starting APPC

The PCOMM software, as its default installation proceeds, does not automatically restart its SNA stack after a reboot. Thus, in the default installation, our software won't be able to use APPC until you manually start the stack.

IBM does provide a way to start the stack automatically; place a shortcut to `autostrt.exe` in the Startup folder.

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 the **Start** icon, 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 the endpoint.

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.

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.

Windows 95 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 thereafter. The underlying file, named `SPXDIR.dat`, is like the `HOSTS` file used in TCP/IP or the LU alias definitions offered with APPC.For Windows, our software makes WinSock version 1.1 Sockets-compatible calls when using the IPX or SPX network protocol.

Determining the IPX Network Address

Previous versions of the endpoint showed the Windows 95 endpoint icon in the System Tray at startup. You could click on the icon and see the IPX/SPX address information for this endpoint computer. Because the endpoint for Windows 95 now runs as a background service and therefore doesn't appear in the system tray, NetIQ now ships the executable program `ipxaddr.exe` as part of its endpoint software.

To determine the IPX/SPX address using `ipxaddr.exe`, type `IPXADDR` at a command prompt, starting from the directory in which you installed the endpoint. (The default is `Program Files\NetIQ\Endpoint`.)

While `ipxaddr.exe` will also work with Windows NT, Windows 2000, and Windows XP, these operating systems already include a utility for determining IPX/SPX addresses. See "Determining Your IPX Network Address" in *Performance Endpoints* for more information.

You can also ask your network administrator for your current 4-byte network number. He or she can load the Monitor program at the server and look in the "Connection Information" section, under the entry for the Windows 95 computer

(presuming it's connected to the NetWare server). You'll see an address that looks like this: `00000002:0207011A308A:0004`. The full IPX address is everything preceding the second colon.

If you already know the IP address of a computer—and thus can Ping to that computer—you can 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.

Sockets Port Number

IPX and SPX applications use their network address to decide which computer to connect to in a network. They use a *port number* to decide which application program to connect to within a computer.

The Sockets port for IPX and SPX is **10117**. 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 issuing the `CONNECT_ACCEPT` commands (usually Endpoint 2) uses the port number specified in the script.

IPX/SPX Limitations with Windows 95

In *Installation Requirements for Windows 95 Endpoints* on page 8-1, we stated that the preferred protocol stack for IPX/SPX on Windows 95 is Novell's latest *Client 3.2 for Windows 95*.

When running 20 (or more) connections repeatedly, we've seen version 2.11 of this stack fail in the following way:

```
A fatal exception 0E has occurred at 0028:C105D82A in VXD
WSIPX(01) + 0000127E. The current application will be
terminated.
```

Expect to see errors like this when you overload the stack.

Windows 95 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.netiq.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

The easiest way to find the local IP address on a Windows 95 computer is to enter the following at an MS-DOS command prompt:

```
WINIPCFG
```

Users of TCP/IP on other operating systems may be familiar with the `NETSTAT` command:

```
NETSTAT -N
```

This shows a line of text for each active connection. The local IP address is in the second column of each row.

You can also find and change your IP address using the graphical user interface. From the Start icon, select **Settings**. Select the **Control Panel** folder, and double-click the **Network** icon. The installed network components are shown.

Double-click **TCP/IP** to get to the **TCP/IP Properties**. Select the **IP Address** page to see or change your local IP address. Select the **DNS Configuration** page 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 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.

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.

See the following topic for more information on limitations.

TCP/IP Limitations with Windows 95

The TCP/IP software shipped by Microsoft in Windows 95 is intended by its designers "for client computers." Microsoft has some limits in its Winsock.dll that make large numbers of connections impractical.

In our testing, we've run Windows 95 computers with 10 TCP connections successfully. We've had problems running 20 connections. We recommend restricting your TCP/IP testing with Windows 95 to about 10 connections per computer. Additional simultaneous connections offer the potential for protection faults, lock-ups, or unpredictable results.

If you're not running WinSock 2 or "Windows 95 OEM service release 2" (OSR2):

- Microsoft has fixed an important bug in Windows 95 that affects applications that use the TCP Sockets API. The fix involves updating the file `Kernel32.dll` with a new version dated 2/2/1996. You must apply this fix to get reliable behavior when using TCP/IP.
- To get the Kernel32 Update, visit the following page at Microsoft's Web site: www.microsoft.com/windows95/downloads/.

To see whether OSR2 is installed, check the System Properties for the version number of Windows 95. If you have version 4.00.950b, OSR2 is installed.

To see whether WinSock 2 is installed, search the computer and see whether the file `WS232.dll` is installed. If the file is installed, then WinSock 2 is installed on the computer.

Running Windows 95 Endpoints

The following topics describe some of the limitations associated with the Windows 95 operating system, starting and stopping an endpoint, and configuring the endpoint for IP multicast and QoS support in Windows operating systems. A final topic describes how the endpoint logs error messages.

Starting a Windows 95 Endpoint

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. This causes `endpoint.exe` to be started automatically when Windows 95 is started.

If you stop `endpoint.exe` and need to restart it without restarting Windows 95, enter

```
ENDPOINT
```

at a command prompt (from the directory where you installed our software).

A single running copy of `endpoint.exe` handles all concurrent tests. If the endpoint program is already running and you try to start another copy, you see a popup error dialog box, "`Endpoint is already running`."

Stopping a Windows 95 Endpoint

To stop the endpoint program, use the command-line option -k. Invoke this command from the directory where you've installed the endpoint:

```
endpoint -k
```

Disabling Automatic Startup

To prevent the endpoint from running automatically at startup, take the following steps:

1. Open the Registry edit utility using the command `REGEDIT`.
2. Navigate to `Hkey_Local_Machine\Software\Microsoft\Windows \CurrentVersion\RunServices` and open it.
3. Write down the value for the variable NetIQ Performance Endpoint, so that you can restart the endpoint as a service later. Delete the **NetIQ Performance Endpoint** key from the Registry.

If you want to restore the setting later, you must do so manually.

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 the Suspend Program

The Suspend program is a power management program. If you run IxChariot or Qcheck tests to an endpoint with Suspend enabled, the test will not complete. Disabling the Suspend program should eliminate these problems.

Logging and Messages

While most error messages encountered by 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 our software. To view an error log, use `FMTLOG`. The version of `FMTLOG` shipped with the Windows 95 endpoint runs as a command from an MS-DOS prompt. `FMTLOG` reads from a binary log file, and writes its formatted output to `stdout`. Use the following `FMTLOG` command:

```
FMTLOG log_filename > output_file
```

For example, to format the error log and write the formatted output to a file named `log.txt`, enter the following at an MS-DOS prompt:

```
FMTLOG d:\Program Files\NetIQ\endpoint\endpoint.log
>log.txt
```

In addition, 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 directory where you installed the endpoint. Save a copy of the file and send it to us via email for problem determination.

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 95 software used by the endpoint program.

See the Support section of the NetIQ Web site for links to the latest software updates.

Updates for Microsoft Windows 95

Microsoft posts code and driver updates directly to the following Web site: www.microsoft.com/windows/downloads/default.asp.

Updates for WinSock 2

Download the WinSock 2 stack for Windows 95 (it's not necessary for Windows 98 or Windows Me) from the following Web site: www.microsoft.com/windows95/downloads/.

Updates for Novell Client Software

Novell posts code and driver updates directly to the following Web site: http://support.novell.com/.

Updates for IBM SNA Software for Windows 95

For information on IBM's Personal Communications (PCOMM) family of software, see the following Web site: www.software.ibm.com/network/pcomm/support/.

9 Windows 98

This chapter explains the installation, configuration, and operation of the Performance Endpoint software for Microsoft Windows 98. This endpoint used to be identical to the endpoint for Windows 95. However, endpoint support for Windows 95 has been archived at version 4.1.

The endpoint for Windows 98 is archived at version 4.3. Therefore, it will not support the latest functionality offered in new releases of IxChariot. Endpoints for other Windows 32-bit and 64-bit operating systems are available in later versions, with functionality not offered in the endpoint for Windows 98.

See "Microsoft Windows NT, Windows 2000, and Windows XP" in the *Performance Endpoints* guide for information about endpoints for other Win32 operating systems.

Installation Requirements for Windows 98 Endpoints

Here's what you need to run the endpoint software with Microsoft Windows 98:

- A computer capable of running Windows 98 well. This implies a CPU such as an Intel 80386, 80486, a member of the Pentium family, or equivalent. A Pentium or better is recommended.
- 8 MBytes of random access memory (RAM); 16 MBytes of RAM is recommended.

  The total RAM requirement depends on the RAM usage of the underlying protocol stack and the number of concurrent connection pairs. Because only about 20 connections are possible with a Windows 98 endpoint, additional memory is probably not required.
- A hard disk with at least 4 MBytes of space available.
- Microsoft Windows 98 with the latest service packs applied. We strongly recommend that you get up-to-date with the latest Windows 98 service levels. *Getting the Latest Fixes and Service Updates* on page 9-13 discusses how to get the latest software.

- One or more compatible network protocol stacks, as described in the following section, *Network Protocol Stacks*.

Network Protocol Stacks

We recommend configuring your networking software -- and making sure that it is working correctly -- before installing our software.

See the online help for your networking software, and see *Configuring Windows 98 Endpoints* on page 9-7 for more assistance.

- **for APPC:**

  Use IBM Personal Communications AS/400 and 3270 for Windows 95 version 4.3 (called PCOMM for Windows 95, eNetworks, or SecureWay) -- or later. It is limited in capacity to about 50 sessions at a time.

- **for IPX and SPX:**

  Microsoft significantly improved the network protocol stacks for TCP/IP and IPX/SPX in Windows 98.

- **for RTP, TCP, and UDP:**

  TCP/IP software is provided as part of the network support in the Windows 98 operating system. Windows 98 comes with Microsoft's latest WinSock 2 protocol stack integrated, which supports IP Multicast, QoS, and about 50 connections.

Endpoint Installation for Windows 98

Following are instructions for installing the endpoint **from a CD-ROM** or **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. From the Start menu, click **Run**.
3. The Run dialog box asks you to enter the name of a program; enter the following in the **Open** field:

   ```
   [drive:]\Endpoint\Win32\gsendw32.exe
   ```

4. If a previous version of the endpoint is present, you are asked if you want to remove it.
5. The next dialog box lets you select the directory for the endpoint. We recommend installing the endpoint on a local hard disk of the computer you're using (if you install on a LAN drive, the additional network traffic will influence your performance results).

   If you have previously installed these endpoints, the default directory is where you previously installed them. If you have not previously installed endpoints but have installed other NetIQ products, the default directory is the same level as the one where your other products are installed. For example, if you installed IxChariot in `C:\Program Files\NetIQ\Chariot`, the default directory is `C:\Program Files\NetIQ`. If you have not previously

installed other NetIQ products, the default drive is the first drive with enough disk space for the endpoints; the default directory is `\Program Files\NetIQ\Endpoint`.

6. Next, you are asked whether to install the pre-built data files, which are used by application scripts to generate the payload data in the frames they send; this is important if there are software or hardware in your network that do data compression. These files take a small amount of hard disk space; we recommend always installing them.

   You are also given the option to start the endpoint after it is installed. If you choose **No**, it is started the next time you restart Windows 98. The endpoint installation next copies the necessary files to your hard disk.

7. Finally, you are asked if 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. 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 at startup, take the following steps:

1. Open the Registry edit utility using the `REGEDIT` command.
2. Navigate to `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServices` and open it.
3. Write down the value for the variable **NetIQ Performance Endpoint**, so that you can restart the endpoint as a service later. Delete the `NetIQ Performance Endpoint` key from the Registry.

   If you want to restore the setting later, you must do so manually.

When you've completed installation, refer to *Configuring Windows 98 Endpoints* on page 9-7 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. Save the endpoint to a directory on a local hard drive.
2. Use Windows Explorer to navigate to the endpoint file, `gsendw32.exe`, and double-click to unzip it and activate Setup. Refer to *Using WinZip* on page 9-5 for instructions.

   If a previous version of the endpoint is present, you are asked whether you want to remove it.

3. The next screen after the Software License Agreement lets you select the directory for the endpoint. We recommend installing the endpoint on a local hard disk of the computer you're using (if you install on a LAN drive, the additional network traffic will influence your performance results).

4. If you have previously installed these endpoints, the default directory is where you previously installed them. If you have not previously installed endpoints but have installed other NetIQ products, the default directory is the same level as the one where your other products are installed. For example, if you installed IxChariot in `C:\Program Files\NetIQ\Chariot`, the default directory is `C:\Program Files\NetIQ`. If you have not previously installed other NetIQ products, the default drive is the first drive with enough disk space for the endpoints; the default directory `is \Program Files\NetIQ\Endpoint`.

   If an endpoint is already installed, you will be prompted to remove the previous installation.

5. Next, you are asked whether to install the pre-built data files, which are used by application scripts to generate the payload data in the frames they send; this is important if there are software or hardware in your network that do data compression. These files take a small amount of hard disk space; we recommend always installing them. You are also given the option to start the endpoint after it is installed. If you choose **No**, it is started when you reboot Windows 98.

6. You are then asked if 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. 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 at startup, take the following steps:

1. Open the Registry edit utility using the command `REGEDIT`.
2. Navigate to `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServices` and open it.
3. Write down the value for the variable NetIQ Performance Endpoint, so that you can restart the endpoint as a service later. Delete the NetIQ Performance Endpoint key from the Registry.

   If you want to restore the setting later, you must do so manually.

When you've completed installation, refer to *Configuring Windows 98 Endpoints* on page 9-7 to make sure your endpoint is ready to be used in testing and monitoring.

Using WinZip

If you are installing endpoints on Windows, you first need to unzip the `gsendw32.exe` file from the CD. For Windows 3.1, the zip file is called `win31.zip`. We recommend using WinZip version 7.0. Follow these steps to unzip the file:

1. Open the WinZip program.
2. On the File menu, click **Open Archive**.
3. In the Open Archive dialog box, browse to the `Endpoint\Win32` directory on the endpoint CD-ROM and select the executable endpoint file `gsendw32.exe`.
4. Click **Open** to unzip the files. The files that were unzipped are shown in the Window.
5. On the Action menu, click **Extract**.
6. In the Extract dialog box, browse to the directory where you want to save the files. This location should be accessible by users who need to install the endpoint.
7. Click **Extract**. The files are extracted to the directory you selected.

Unattended Installation for Windows 98

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 "`gsendw32.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\NetIQ\Endpoint` directory on our `n:` LAN drive:

```
SETUP -s -f1n:\Program Files\NetIQ\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.

Installing the Windows 98 Endpoint with SMS

See "Distributing Endpoints Using SMS" in the *Performance Endpoints* guide for information on automatically installing (and uninstalling) endpoints, using Microsoft's Systems Management Server (SMS).

What Happens During Installation

Here's what happens during the installation steps. Let's say you install the endpoint into the directory `\Program Files\NetIQ\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 installation process for a Windows 98 endpoint makes no changes to `CONFIG.SYS` or `AUTOEXEC.BAT`. The installation process does, however, involve adding the endpoint to the Registry so that the endpoint starts automatically when you start a Windows 98 computer.

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

1. Stop the endpoint. See *Stopping a Windows 98 Endpoint* on page 9-12 for instructions.
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.

Removing the Endpoint Package (Uninstall)

If you need 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 the **Add/Remove Programs** icon. The Add/Remove Programs Properties dialog box is shown.
3. Highlight **NetIQ Endpoint** and click **Add/Remove**. The uninstallation program begins. After the program is completed, the endpoint should now be uninstalled.

Removing the Endpoint Manually

If the uninstallation program is unable to uninstall the endpoint, you will need to manually uninstall the endpoint.

For detailed instructions on manually removing the endpoints, see the Performance Endpoints FAQ page in the Knowledge Base on our Web site at www.netiq.com/support/pe/default.asp.

Configuring Windows 98 Endpoints

The Performance Endpoint program is a 32-bit application for Windows 98, using the 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.

- Determine the network addresses of the computers to be used in tests
- Verify the network connections

The following topics describe how to accomplish these steps for Windows 98:

- *Windows 98 Configuration for APPC*
- *Windows 98 Configuration for IPX and SPX* on page 9-9
- *Windows 98 Configuration for TCP/IP* on page 9-10

Windows 98 Configuration for APPC

The following sections describe the steps for setting up APPC at the endpoint, for use with IBM's Personal Communications (PCOMM) software for Windows 95. If you are new to configuring APPC, start with the guidance provided by the APPC network software you're using.

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. The URL is: www.redbooks.ibm.com/pubs/pdfs/redbooks/gg244485.pdf.

Determining the APPC Network Address

To determine the fully qualified LU name of any computer running PCOMM, do the following:

1. Start the SNA Node Operations program by either running `pcsnops.exe` from a command prompt or by clicking the icon.
2. If the node is not currently started, select **Operations...Start Node**.
3. The first panel shown should be the Node panel, which shows a value titled FQCP Name. If this is not visible, select **View...Select Resource Attributes** and select it for viewing. A default fully qualified LU name is automatically configured and it has the exact same name as the FQCP Name shown in this panel.

A fully qualified LU name is the easiest network address to use with our software. Although you can define multiple LUs at one computer, the default LU name is the one on which the endpoint "listens" for a connection.

Automatically Starting APPC

The PCOMM software, as its default installation proceeds, does not automatically restart its SNA stack after a reboot. Thus, in the default installation, our software won't be able to use APPC until you manually start the stack.

IBM does provide a way to start the stack automatically; place a shortcut to `autostrt.exe` in the Startup folder.

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 the **Start** icon, 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 the endpoint.

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.

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.

Windows 98 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.

For Windows, our software makes WinSock version 1.1 Sockets-compatible calls when using the IPX or SPX network protocol.

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 thereafter. The underlying file, named `SPXDIR.dat`, is like the `HOSTS` file used in TCP/IP or the LU alias definitions offered with APPC.

Determining the IPX Network Address

Previous versions of the endpoint showed the Windows 98 endpoint icon in the System Tray at startup. You could click on the icon and see the IPX/SPX address information for this endpoint computer. Because the endpoint for Windows 98 now runs as a background service and therefore doesn't appear in the system tray, NetIQ now ships the executable program `ipxaddr.exe` as part of its endpoint software.

To determine the IPX/SPX address using `ipxaddr.exe`, type `IPXADDR` at a command prompt, starting from the directory in which you installed the endpoint. (The default is `Program Files\NetIQ\Endpoint`.)

While `ipxaddr.exe` will also work with Windows NT, Windows 2000, and Windows XP, these operating systems already include a utility for determining IPX/SPX addresses. See "Determining You IPX Network Address" in the *Performance Endpoints* guide for more information.

You can also ask your network administrator for your current 4-byte network number. He or she can load the Monitor program at the server and look in the "Connection Information" section, under the entry for the Windows 98 computer (presuming it's connected to the NetWare server). You'll see an address that looks like this: `00000002:0207011A308A:0004`. The full IPX address is everything preceding the second colon.

If you already know the IP address of a computer -- and thus can Ping to that computer -- you can 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.

Sockets Port Number

IPX and SPX applications use their network address to decide which computer to connect to in a network. They use a *port number* to decide which application program to connect to within a computer.

The Sockets port for IPX and SPX is **10117**. 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 issuing the `CONNECT_ACCEPT` commands (usually Endpoint 2) uses the port number specified in the script.

Windows 98 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.netiq.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

The easiest way to find the local IP address on a Windows 98 computer is to enter the following at an MS-DOS command prompt:

```
WINIPCFG
```

Users of TCP/IP on other operating systems may be familiar with the `NETSTAT` command:

```
NETSTAT -N
```

This shows a line of text for each active connection. The local IP address is in the second column of each row.

You can also find and change your IP address using the graphical user interface. From the Start icon, select **Settings**. Select the **Control Panel** folder, and double-click the **Network** icon. The installed network components are shown.

Double-click **TCP/IP** to get to the **TCP/IP Properties**. Select the **IP Address** page to see or change your local IP address. Select the **DNS Configuration** page 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 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.

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.

See the following topic for more information on limitations.

Running Windows 98 Endpoints

The following topics describe starting and stopping the endpoint and configuring the endpoint for IP multicast and QoS support in the Windows 98 operating system. A final topic describes how the endpoint logs error messages.

Starting a Windows 98 Endpoint

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. This causes `endpoint.exe` to be started automatically when Windows 98 is started.

If you stop `endpoint.exe` and need to restart it without restarting Windows 98, enter

```
ENDPOINT
```

at a command prompt (from the directory where you installed our software).

A single running copy of `endpoint.exe` handles all concurrent tests. If the endpoint program is already running and you try to start another copy, you see a popup error dialog box, "`Endpoint is already running`."

Stopping a Windows 98 Endpoint

To stop the endpoint program, use the command-line option `-k`. Invoke this command from the directory where you've installed the endpoint:

```
endpoint -k
```

Disabling Automatic Startup

To prevent the endpoint from running automatically at startup, take the following steps:

- Open the Registry edit utility using the command `REGEDIT`.
- Navigate to `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServices` and open it.
- Write down the value for the variable NetIQ Performance Endpoint, so that you can restart the endpoint as a service later. Delete the **NetIQ Performance Endpoint** key from the Registry.

If you want to restore the setting later, you must do so manually.

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 the Suspend Program

The Suspend program is a power management program. If you run IxChariot or Qcheck tests to an endpoint with Suspend enabled, the test will not complete. Disabling the Suspend program should eliminate these problems.

Logging and Messages

While most error messages encountered by 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 our software. To view an error log, use `FMTLOG`. The version of `FMTLOG` shipped with the Windows 98 endpoint runs as a command from an MS-DOS prompt. `FMTLOG` reads from a binary log file, and writes its formatted output to `stdout`. Use the following `FMTLOG` command:

```
FMTLOG log_filename > output_file
```

For example, to format the error log and write the formatted output to a file named `log.txt`, enter the following at an MS-DOS prompt:

```
FMTLOG d:\Program Files\NetIQ\endpoint\endpoint.log
>log.txt
```

In addition, 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 directory where you installed the endpoint. Save a copy of the file and send it to us via email for problem determination.

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 98 software used by the endpoint program.

See the Support section of the NetIQ Web site for links to the latest software updates.

Updates for Windows 98

Microsoft posts code and driver updates directly to the following Web site: www.microsoft.com/windows/downloads/

Updates for Novell Client Software

Novell posts code and driver updates directly to the following Web site: http://support.novell.com/.

Updates for IBM SNA Software for Windows 98

For information on IBM's Personal Communications (PCOMM) family of software, see the following Web site: www.software.ibm.com/network/pcomm/support/.

10 Microsoft Windows CE 4.X

This chapter describes the installation, configuration, and operation of the Performance Endpoint software for Microsoft Windows CE 4.X.

Topics in this chapter:

Available Performance Endpoints for Windows CE

Ixia provides four distinct Performance Endpoints for Windows CE:

- **pewcex86** – Performance Endpoint for Windows CE running on Intel x86 processors.
- **pewcearm** – Performance Endpoint for Windows CE running on Intel Strong Arm and XScale processors.
- **pewcearm_cl** – Command line version of the Windows CE Performance Endpoint running on Intel Strong Arm and XScale processors. This is the same as the wcearm endpoint minus the GUI.
- **pewcearm_disk** – Performance Endpoint for Windows CE running on Intel Strong Arm and XScale processors, with file storage support.

You can run both streaming and non-streaming tests using the Windows CE Performance Endpoints. You can also run IP Multicast tests that include these endpoint as part of a multicast group.

Most IxChariot testing parameters are supported, but note exceptions in *Limitations of the Windows CE Endpoint* on page 10-8.

Installation Requirements

Table 10-1 describes the requirements for installing and running the Microsoft Windows CE Performance Endpoint software.

Table 10-1. Windows CE Performance Endpoints

Windows CE Endpoint	Supported Processors	Operating System Version	RAM Required
pewcex86	Intel x86 compatible	Windows CE 4.2, 4.3	64 MB
pewcearm	Intel Strong Arm, Intel XScale	Windows CE 4.2, 4.3	64 MB
pewcearm_disk	Intel Strong Arm, Intel XScale	Windows CE 4.2, 4.3	64 MB
pewcearm_cl	Intel Strong Arm, Intel XScale	Windows CE 4.2, 4.3	64 MB

The Performance Endpoint file names include the product release number. For example, *pwecex86_640.exe* is version 6.40 of the Windows CE Intel x86-compatible Performance Endpoint.

The Windows CE Performance Endpoint supports Windows Mobile 5.0, a compact operating system that is packaged with a suite of basic applications for mobile devices. Windows Mobile is powered by Windows CE 5.0 and uses the .NET Compact Framework. It runs on devices such as Pocket PCs, Smartphones, and Portable Media Centers.

Network Protocol Stacks

We recommend that you configure your networking software–and make sure that it is working correctly–before installing the Performance 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.

The TCP/IP and UDP/RTP protocols are supported by the Performance Endpoint for Windows CE. The Windows CE Performance Endpoints run on any IP network, regardless of topology. For example, we have tested it with 802.11a/b/g wireless links and 10/100/1000 Ethernet links.

Endpoint Installation for Windows CE

Installing the *pewcearm* Performance Endpoint

The following installation instructions assume that the Windows CE device to be tested is already synched to your desktop computer:

Follow these steps to install the GUI version of the endpoint:

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, 640 for release 6.40.
4. Paste the file to the following directory:

   ```
   [Mobile Device]\My Pocket PC\Windows\Start Menu
   ```

The endpoint is now ready for use. Refer to *Running Windows CE Endpoints* on page 10-6 for additional instructions.

Installing the *pewcearm_cl* Performance Endpoint

Follow these steps to install the command line version of the endpoint:

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_cl_Mm.exe` to your Windows CE device, using the tools available on your device.

Once you have copied the endpoint, it is ready for use. Refer to *Running Windows CE Endpoints* on page 10-6 for additional instructions.

Installing the *pewcearm_disk* Performance Endpoint

Follow these steps to install the pewcearm_disk endpoint:

1. From your Windows CE device, use your Web browser to navigate to the Windows CE endpoint at www.ixiacom.com/support/ixchariot.
2. Download the **pewcearm-disk-*Mm*.exe** self-extracting archive file to a disk drive on your Windows CE device.
3. Double-click the pewcearm-disk-Mm.exe file to extract the contents.

   You can place these files anywhere on the storage device (whether in the root directory, or in a user-defined directory). The self-extracting archive includes all the files you need to run the Performance Endpoint, including:

   - pewcearm_disk.exe (the Performance Endpoint executable)
   - endpoint.ini
   - echr.msg
   - the cmpfiles directory

4. Modify the endpoint.ini, as required for your testing.

Once you have copied the endpoint, it is ready for use. Refer to *Running Windows CE Endpoints* on page 10-6 for additional instructions.

Installing the *pewcex86* Performance Endpoint

The following installation instructions assume that the Windows CE device to be tested is already synched to your desktop computer:

Follow these steps to install the endpoint:

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, 630 for release 6.30.
4. Paste the file to the following directory:

```
[Mobile Device]\My Pocket PC\Windows\Start Menu
```

The endpoint is now ready for use. Refer to *Running Windows CE Endpoints* on page 10-6 for additional instructions.

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, 640 for release 6.40. 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, 640 for release 6.40.

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.

If you are using the command line version of the endpoint, the procedure for determining your IP address depends on the tools available on the device.

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

IP networks use *network addresses* to forward traffic across a network to a specific device, and they use *port numbers* to deliver traffic to a specific application running on the selected device.

IxChariot uses a designated *management port* to transport test management traffic between the console and the endpoints. The management port is one of the following:

- SPX transport: port 10117
- TCP transport: either port 10115 (the default) or a user-selected port.

IxChariot uses other ports for test traffic. If an IxChariot script specifies "`port_number=AUTO`" on the `CONNECT_ACCEPT` command, 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 how to start and stop an endpoint, and how to check the version of an endpoint. A final section describes how the endpoint handles error messages.

Intel Strong Arm and XScale Processor Based Operation

Following are instructions for starting and stopping the three versions of the pewcearm Performance Endpoint.

Starting the *pewcearm* Performance Endpoint

On your Windows CE device, tap **Start > pewcearm_*Mm*.exe**. *Mm* is the endpoint release number; for example, 640 for release 6.40.

Starting the *pewcearm_cl* Performance Endpoint

Procedures for starting and stopping the command line version of the Performance Endpoint depend on the tools available on the device. For example, for some devices you will enter **endpoint** at the command line to start the endpoint and use CTRL-C to stop the endpoint.

Starting the *pewcearm_disk* Performance Endpoint

On your Windows CE device, navigate to the directory where you have installed the Performance Endpoint files, then tap the executable (pewcearm_disk.exe) to start the endpoint.

Stopping the *pewcearm* and *pewcearm_disk* Performance 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**.

Stopping the *pewcearm_cl* Performance Endpoint

Procedures for starting and stopping the command line version of the Performance Endpoint depend on the tools available on the device. For example, for some devices you will enter **endpoint** at the command line to start the endpoint and use CTRL-C to stop the endpoint.

Intel x86 Processor Based Operation

Starting the *pewcex86* 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, 640 for release 6.40.

Stopping the *pewcex86* Endpoint

To stop the pewcex86 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.

If you are using the command line version of the endpoint, the procedure for displaying the endpoint version depends on the tools available on the device.

Logging and Messages

All error messages encountered on an endpoint are returned to the IxChariot or Qcheck Console.

For the `pewcearm_disk` Performance Endpoint, some error messages are logged to disk. These messages 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
```

In addition, if an assertion failure occurs, the `pewcearm_disk` Performance Endpoint writes a file named `assert.err` to the directory where you installed the endpoint.

Note that only the `pewcearm_disk` Performance Endpoint provides support for disk storage. The other Windows CE Performance Endpoints (pewcex86, pewcearm, and pewcearm_cl) do not provide disk support.

Limitations of the Windows CE Endpoint

The Windows CE Performance Endpoints do not support the following IxChariot test parameters:

- Disabling the UDP checksum.
- DiffServ QoS templates.
- Traceroute testing.
- Application scripts with `.cmp` data files as the datatype.

  Scripts that use `.cmp` files by default, such as the Internet scripts, will run only on the *pewcearm_disk* version of this endpoint.

  As a work-around on the other versions of the endpoint, edit the scripts to use `NOCOMPRESS` as the `send_datatype` instead of a `.cmp` file.

Additional Limitations:

- Support for CPU Utilization on Windows CE is device-dependent. For more information, see http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wcemain4/html/cerefGetIdleTime.asp.
- 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.

11 Microsoft Windows ME

This chapter explains the installation, configuration, and operation of the Performance Endpoint software for Microsoft Windows Millennium Edition (Me). This endpoint used to be identical to the endpoint for Windows 95 and the endpoint for Windows 98. However, endpoint support for Windows 95 has been archived at version 4.1, while endpoint support for Windows 98 has been archived at version 4.3.

The endpoint software for Windows Me is installed from the same file as the endpoint software for Windows NT/2000/2003/XP. However, some code, hardware and software requirements, installation instructions, and endpoint capabilities are different for the endpoint actually installed on the Windows Me operating system.

Note: The maximum number of pairs that can be licensed to run on Windows ME is limited to 50.

Installation Requirements for Windows Me Endpoints

Here's what you need to run the endpoint software with Microsoft Windows Me:

- A computer capable of running Windows Me well. This implies a CPU such as an Intel 80386, 80486, a member of the Pentium family, or equivalent. A Pentium or better is recommended.
- 8 MBytes of random access memory (RAM); 16 MBytes of RAM is recommended.
- The total RAM requirement depends on the RAM usage of the underlying protocol stack and the number of concurrent connection pairs. Since only about 20 connections are possible with a Windows Me endpoint, additional memory is probably not required.
- A hard disk with at least 4 MBytes of space available.

- Microsoft Windows Me with the latest service packs applied. We strongly recommend that you get up-to-date with the latest Windows Me service levels. *Getting the Latest Fixes and Service Updates* on page 11-13 discusses how to get the latest software.
- One or more compatible network protocol stacks, as described in the following section, *Network Protocol Stacks*.

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

Network Protocol Stacks

We recommend configuring your networking software -- and making sure that it is working correctly -- before installing our software.

See the online help for your networking software, and see *Configuring Windows Me Endpoints* on page 11-7 for more assistance.

- for APPC:

  Use IBM Personal Communications AS/400 and 3270 for Windows 95 version 4.3 (called PCOMM for Windows 95, eNetworks, or SecureWay) -- or later. It is limited in capacity to about 50 sessions at a time.

- for IPX and SPX:

  Microsoft significantly improved the network protocol stacks for TCP/IP and IPX/SPX in Windows Me.

- for RTP, TCP, and UDP:

  TCP/IP software is provided as part of the network support in the Windows Me operating system. Windows Me comes with Microsoft's latest WinSock 2 protocol stack integrated, which supports IP Multicast, QoS, and about 50 connections.

Endpoint Installation for Windows Me

Following are instructions for installing the endpoint from a CD-ROM or 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. From the Start menu, click **Run**.
3. The Run dialog box asks you to enter the name of a program; enter the following in the **Open** field:

   ```
   [drive:]\Endpoint\Win32\pew32_Mm.exe
   ```

4. If a previous version of the endpoint is present, you are asked if you want to remove it.

5. The next dialog box lets you select the directory for the endpoint. We recommend installing the endpoint on a local hard disk of the computer you're using (if you install on a LAN drive, the additional network traffic will influence your performance results).

   If you have previously installed these endpoints, the default directory is where you previously installed them. If you have not previously installed endpoints but have installed other NetIQ products, the default directory is the same level as the one where your other products are installed. For example, if you installed IxChariot in `C:\Program Files\Ixia\IxChariot`, the default directory is `C:\Program Files\Ixia`.

6. Next, you are asked whether to install the pre-built data files, which are used by application scripts to generate the payload data in the frames they send; this is important if there are software or hardware in your network that do data compression. These files take a small amount of hard disk space; we recommend always installing them.

   You are also given the option to start the endpoint after it is installed. If you choose **No**, it is started the next time you restart Windows Me. The endpoint installation next copies the necessary files to your hard disk.

7. Finally, you are asked if 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 at startup, take the following steps:

1. Open the Registry edit utility using the `REGEDIT` command.
2. Navigate to `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServices` and open it.
3. Write down the value for the variable **NetIQ Performance Endpoint**, so that you can restart the endpoint as a service later. Delete the `NetIQ Performance Endpoint` key from the Registry.

   If you want to restore the setting later, you must do so manually.

When you've completed installation, refer to *Configuring Windows Me Endpoints* on page 11-7 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. Save the endpoint to a directory on a local hard drive.

2. Use Windows Explorer to navigate to the endpoint file, `pew32_Mm.exe`, and double-click to unzip it and activate Setup. Refer to *Using WinZip* on page 11-5 for instructions.

   If a previous version of the endpoint is present, you are asked whether you want to remove it.

3. The next screen after the Software License Agreement lets you select the directory for the endpoint. We recommend installing the endpoint on a local hard disk of the computer you're using (if you install on a LAN drive, the additional network traffic will influence your performance results).

4. If you have previously installed these endpoints, the default directory is where you previously installed them. If you have not previously installed endpoints but have installed other NetIQ products, the default directory is the same level as the one where your other products are installed. For example, if you installed IxChariot in `C:\Program Files\Ixia\IxChariot`, the default directory is `C:\Program Files\Ixia`.

   If an endpoint is already installed, you will be prompted to remove the previous installation.

5. Next, you are asked whether to install the pre-built data files, which are used by application scripts to generate the payload data in the frames they send; this is important if there are software or hardware in your network that do data compression. These files take a small amount of hard disk space; we recommend always installing them. You are also given the option to start the endpoint after it is installed. If you choose **No**, it is started when you reboot Windows Me.

6. You are then asked if 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 at startup, take the following steps:

1. Open the Registry edit utility using the command `REGEDIT`.

2. Navigate to `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServices` and open it.

3. Write down the value for the variable NetIQ Performance Endpoint, so that you can restart the endpoint as a service later. Delete the NetIQ Performance Endpoint key from the Registry.

   If you want to restore the setting later, you must do so manually.

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

Using WinZip

If you are installing endpoints on Windows, you first need to unzip the `pew32_Mm.exe` file from the CD. For Windows 3.1, the zip file is called `win31.zip`. We recommend using WinZip version 7.0. Follow these steps to unzip the file:

1. Open the WinZip program.
2. On the File menu, click **Open Archive**.
3. In the Open Archive dialog box, browse to the `Endpoint\Win32` directory on the endpoint CD-ROM and select the executable endpoint file `pew32_Mm.exe`.
4. Click **Open** to unzip the files. The files that were unzipped are shown in the Window.
5. On the Action menu, click **Extract**.
6. In the Extract dialog box, browse to the directory where you want to save the files. This location should be accessible by users who need to install the endpoint.
7. Click **Extract**. The files are extracted to the directory you selected.

Unattended Installation for Windows Me

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\NetIQ\Endpoint` directory on our `n:` LAN drive:

```
SETUP -s -f1n:\Program Files\NetIQ\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 prob-

ably don't want your unattended install to ask you for `n:\disk1\`, `n:\disk2\`, and so on.

Installing the Windows Me Endpoint with SMS

See Chapter 3, *Distributing Endpoints using SMS* for information on automatically installing (and uninstalling) endpoints, using Microsoft's Systems Management Server (SMS).

What Happens 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 installation process for a Windows Me endpoint makes no changes to `CONFIG.SYS` or `AUTOEXEC.BAT`. The installation process does, however, involve adding the endpoint to the Registry so that the endpoint starts automatically when you start a Windows Me computer.

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

1. Stop the endpoint. See *Stopping a Windows Me Endpoint* on page 11-12 for instructions.
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.

Removing the Endpoint Package (Uninstall)

If you need 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 the **Add/Remove Programs** icon. The Add/Remove Programs Properties dialog box is shown.
3. Highlight **Performance Endpoint** and click **Add/Remove**. The uninstallation program begins. After the program is completed, the endpoint should now be uninstalled.

Removing the Endpoint Manually

If the uninstallation program is unable to uninstall the endpoint, you will need to manually uninstall the endpoint.

For detailed instructions on manually removing the endpoints, see the Performance Endpoints FAQ page in the Knowledge Base on our Web site at www.netiq.com/support/pe/default.asp.

Configuring Windows Me Endpoints

The Performance Endpoint program is a 32-bit application for Windows Me, using the 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.

- Determine the network addresses of the computers to be used in tests
- Verify the network connections

The following topics describe how to accomplish these steps for Windows Me:

- *Windows Me Configuration for APPC*
- *Windows Me Configuration for IPX and SPX* on page 11-9
- *Windows Me Configuration for TCP/IP* on page 11-10

Windows Me Configuration for APPC

The following sections describe the steps for setting up APPC at the endpoint, for use with IBM's Personal Communications (PCOMM) software for Windows 95. If you are new to configuring APPC, start with the guidance provided by the APPC network software you're using.

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. The URL is: www.redbooks.ibm.com/pubs/pdfs/redbooks/gg244485.pdf.

Determining the APPC Network Address

To determine the fully qualified LU name of any computer running PCOMM, do the following:

1. Start the SNA Node Operations program by either running `pcsnops.exe` from a command prompt or by clicking the icon.
2. If the node is not currently started, select **Operations...Start Node**.
3. The first panel shown should be the Node panel, which shows a value titled FQCP Name. If this is not visible, select **View...Select Resource Attributes** and select it for viewing. A default fully qualified LU name is automatically configured and it has the exact same name as the FQCP Name shown in this panel.

A fully qualified LU name is the easiest network address to use with our software. Although you can define multiple LUs at one computer, the default LU name is the one on which the endpoint "listens" for a connection.

Automatically Starting APPC

The PCOMM software, as its default installation proceeds, does not automatically restart its SNA stack after a reboot. Thus, in the default installation, our software won't be able to use APPC until you manually start the stack.

IBM does provide a way to start the stack automatically; place a shortcut to `autostrt.exe` in the Startup folder.

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 the **Start** icon, 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 the endpoint.

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.

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.

Windows Me 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.

For Windows, our software makes WinSock version 1.1 Sockets-compatible calls when using the IPX or SPX network protocol.

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 thereafter. The underlying file, named `SPXDIR.dat`, is like the `HOSTS` file used in TCP/IP or the LU alias definitions offered with APPC.

Determining the IPX Network Address

Previous versions of the endpoint showed the Windows Me endpoint icon in the System Tray at startup. You could click on the icon and see the IPX/SPX address information for this endpoint computer. Because the endpoint for Windows Me now runs as a background service and therefore doesn't appear in the system tray, NetIQ now ships the executable program `ipxaddr.exe` as part of its endpoint software.

To determine the IPX/SPX address using `ipxaddr.exe`, type `IPXADDR` at a command prompt, starting from the directory in which you installed the endpoint. (The default is `Program Files\NetIQ\Endpoint`.)

While `ipxaddr.exe` will also work with Windows NT, Windows 2000, and Windows XP, these operating systems already include a utility for determining IPX/SPX addresses.

You can also ask your network administrator for your current 4-byte network number. He or she can load the Monitor program at the server and look in the "Connection Information" section, under the entry for the Windows Me computer (presuming it's connected to the NetWare server). You'll see an address that looks like this: `00000002:0207011A308A:0004`. The full IPX address is everything preceding the second colon.

If you already know the IP address of a computer -- and thus can Ping to that computer -- you can 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.

Sockets Port Number

IPX and SPX applications use their network address to decide which computer to connect to in a network. They use a *port number* to decide which application program to connect to within a computer.

The Sockets port for IPX and SPX is **10117**. 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 issuing the `CONNECT_ACCEPT` commands (usually Endpoint 2) uses the port number specified in the script.

Windows Me 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.netiq.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

The easiest way to find the local IP address on a Windows Me computer is to enter the following at an MS-DOS command prompt:

```
WINIPCFG

Users of TCP/IP on other operating systems may be
familiar with the NETSTAT command:

NETSTAT -N
```

This shows a line of text for each active connection. The local IP address is in the second column of each row.

You can also find and change your IP address using the graphical user interface. From the Start icon, select **Settings**. Select the **Control Panel** folder, and double-click the **Network** icon. The installed network components are shown.

Double-click **TCP/IP** to get to the **TCP/IP Properties**. Select the **IP Address** page to see or change your local IP address. Select the **DNS Configuration** page 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 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.

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.

See the following topic for more information on limitations.

Running Windows Me Endpoints

The following topics describe starting and stopping the endpoint and configuring the endpoint for IP multicast and QoS support in Windows Me operating systems. A final topic describes how the endpoint logs error messages.

Starting a Windows Me Endpoint

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. This causes `endpoint.exe` to be started automatically when Windows Me is started.

If you stop `endpoint.exe` and need to restart it without restarting Windows Me, enter

```
ENDPOINT
```

at a command prompt (from the directory where you installed our software).

A single running copy of `endpoint.exe` handles all concurrent tests. If the endpoint program is already running and you try to start another copy, you see a popup error dialog box, "`Endpoint is already running.`"

Stopping a Windows Me Endpoint

To stop the endpoint program, use the command-line option `-k`. Invoke this command from the directory where you've installed the endpoint:

```
endpoint -k
```

Disabling Automatic Startup

To prevent the endpoint from running automatically at startup, take the following steps:

- Open the Registry edit utility using the command `REGEDIT`.
- Navigate to `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows \CurrentVersion\RunServices` and open it.
- Write down the value for the variable NetIQ Performance Endpoint, so that you can restart the endpoint as a service later. Delete the **Performance Endpoint** key from the Registry.

  If you want to restore the setting later, you must do so manually.

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 the Suspend Program

The Suspend program is a power management program. If you run IxChariot or Qcheck tests to an endpoint with Suspend enabled, the test will not complete. Disabling the Suspend program should eliminate these problems.

Logging and Messages

While most error messages encountered by 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 our software. To view an error log, use `FMTLOG`. The version of `FMTLOG` shipped with the Windows Me endpoint runs as a command from an MS-DOS prompt. `FMTLOG` reads from a binary log file, and writes its formatted output to `stdout`. Use the following `FMTLOG` command:

```
FMTLOG log_filename > output_file
```

For example, to format the error log and write the formatted output to a file named `log.txt`, enter the following at an MS-DOS prompt:

```
FMTLOG d:\Program Files\Ixia\endpoint\endpoint.log
>log.txt
```

In addition, 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 directory where you installed the endpoint. Save a copy of the file and send it to us via email for problem determination.

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 Me software used by the endpoint program.

Updates for Microsoft Windows Me

Microsoft posts code and driver updates directly to the following Web site: www.microsoft.com/windows/downloads/

Updates for Novell Client Software

Novell posts code and driver updates directly to the following Web site: http://support.novell.com/.

Updates for IBM SNA Software for Windows Me

For information on IBM's Personal Communications (PCOMM) family of software, see the following Web site: www.software.ibm.com/network/pcomm/support/.

12 Compaq Tru64 UNIX

The following topics explain the installation, configuration, and operation of the Performance Endpoint software for UNIX, running onlTherefore, it will not support the latest functionality offered in new releases of IxChariot.

Installation Requirements for Compaq Tru64 UNIX Endpoints

Here's what you need to run the endpoint program with Compaq Tru64 or Digital UNIX:

- A computer capable of running Compaq Tru64 or Digital UNIX. This implies a computer such as the DEC AlphaStation.
- 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.
- Compaq Tru64 or Digital UNIX, Version 4.0B 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.

From this point forward in the documentation, we will refer to this endpoint as the Compaq Tru64 UNIX endpoint. This term refers to endpoint that was formerly called the Digital UNIX endpoint.

Endpoint Installation for Compaq Tru64 UNIX

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 instructions below explain how to install an endpoint from a CD-ROM and from the World Wide Web.

To install an 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 `rz4c` and you're able to create a temporary directory named `cdrom`:

   ```
   mkdir /cdrom
   mount -r -t cdfs -o noversion -r /dev/rz4c /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/archive/decunix/enddecr.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 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 `/opt/NetIQ`. 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 `umount` command to unmount the file system from the CD-ROM:

   ```
   umount /cdrom
   ```

   The installation script and temporary directory are not removed automatically.

   To remove the temp files, enter:

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

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

```
more /opt/NetIQ/README
```

When you've completed installation, refer to *Configuring Compaq Tru64 UNIX Endpoints* on page 12-5 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 `enddecr.tar.Z` file to the `/tmp` directory.
3. Uncompress the endpoint file by using the `uncompress` command:

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

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

   ```
   ./endpoint.install
   ```

The endpoint installs itself in `/opt/NetIQ`. 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.

The installation script and temporary directory are not removed automatically.

To remove the temp files, enter:

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

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

```
more /opt/NetIQ/README
```

When you've completed installation, refer to *Configuring Compaq Tru64 UNIX Endpoints* on page 12-5 to make sure your endpoint is ready to be used in testing and monitoring.

Unattended Installation for Compaq Tru64 UNIX

Unattended installation is available for the Compaq Tru64 UNIX endpoint. 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
```

Removing the Endpoint Package (Uninstall)

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

```
/opt/NetIQ/endpoint.remove
```

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

This removes the files from `/opt/NetIQ` 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.

What We Do During Installation

Here's what happens during the installation steps. The endpoint is installed into the directory `/opt/NetIQ`. 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.

Our software ends 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/rc3.d` directory so the endpoint is started every time your system starts.

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.

Configuring Compaq Tru64 UNIX Endpoints

The endpoint dynamically configures its own programs, so you do not have to edit 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's how to determine the IP address of the local computer you're using.

```
netstat -in
```

You may have several network interfaces. If you are using a LAN network, for example, look at the output for the tu0 interface; your local IP address is shown in the "Address" column.

Trying Out 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 -c 1 xx.xx.xx.xx
```

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

```
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 (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 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 Compaq Tru64 UNIX 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 Compaq Tru64 UNIX Endpoint

The endpoint program is installed so that it will start automatically each time Compaq Tru64 UNIX 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/NetIQ/endpoint &
```

The "&" parameter indicates to Compaq Tru64 UNIX 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 Compaq Tru64 UNIX 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/NetIQ/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`" (described above). If that does not stop the endpoint, kill the endpoint using the UNIX "`kill`" command.

Next, enter the following command:

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

Endpoint Dumps Core or Fails to Run Tests

If the endpoint dumps core or does not run tests, check the `/var/adm/endpoint.console` file for the following message:

If you see a similar message, you need to install the latest patch kit. Obtain patch kits from Compaq's Web site: http://gatekeeper.dec.com/pub/DEC/OSF1/patch/. For instructions on the patch, click on the link for OldStyleREADME-4-19970808.txt.You can also receive this message if you have run out of disk space on the computer where the endpoint program is installed. Verify that you have enough disk space.DECthreads bugcheck (version V3.13-435), terminating execution.

```
vpUpcallThreadUnblocked: (os/kern) invalid argument (4)
nxm_resched(213,0)
```

How to Tell If a Compaq Tru64 UNIX 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 rightmost column of the output, "`/opt/NetIQ/endpoint`."

Disabling Automatic Startup

To disable automatic startup, remove the initialization file named `/sbin/rc3.d/S81endpoint`.

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 NetIQ program named `FMTLOG`. `FMTLOG` reads from a binary log file, and writes its formatted output to `stdout`. Use the following `FMTLOG` command:

```
/opt/NetIQ/fmtlog /var/adm/endpoint.log >output_filename
```

The endpoint code does internal checking. Our software captures details related to the problem in an ASCII 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 Compaq Tru64 UNIX computer, it says that the endpoint program on Compaq Tru64 UNIX has run out of system semaphores. Each instance of Endpoint 1 requires a system semaphore. The maximum number of semaphores is not configurable on Compaq Tru64 UNIX; it is hard-coded to a large value. To avoid this problem, stop other programs that use semaphores or decrease the number of connection pairs 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 recommend working with the very latest software for the underlying operating system and communications software. Following are the best sources we've found for the Compaq Tru64 UNIX software used by the endpoint program.

Updates for Compaq Tru64 UNIX

Compaq posts code and driver updates directly to the following Web sites:

- www.unix.digital.com/
- www.service.digital.com/
- www.compaq.com/services/

13 FreeBSD Unix

The following topics explain the installation, configuration, and operation of the Performance Endpoint software for the FreeBSD operating system.

The endpoint for FreeBSD UNIX has been archived at endpoint version 4.2. Therefore, it will not support the latest functionality offered in new releases of IxChariot.

Installation Requirements for FreeBSD Endpoints

The FreeBSD Endpoint works with FreeBSD 4.x (or later), however it does require the file `libc_r.so.3` (version 3 of the standard C library) be installed. To do this on a FreeBSD 4.x computer, install the FreeBSD "3.x Compatibility" package. Or as an alternative step, create a link from the newer version of `libc_r.so` to `libc_r.so.3`. To do this, run the following commands as the "`root`" user:

```
cd /usr/lib
ln -s libc_r.so.3 libc_r.so
```

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

- A computer capable of running FreeBSD well. This implies a CPU such as Intel 80386, 80486, a member of the Pentium family, or equivalent. A Pentium or better is recommended.
- 32 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.
- 32 MBytes of random access memory (RAM).
- A hard disk with at least 8 MBytes of space available.

- FreeBSD version 3.1 (or higher), with TCP/IP networking and corresponding networking hardware installed and configured. This version supports IP Multicast and RTP.
- 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.

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 examples provided. The following instructions describe how to install the endpoint from a CD-ROM or from the World Wide Web.

Endpoint Installation for FreeBSD

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

1. Put the endpoint CD-ROM in your CD-ROM drive.
2. For this example, we assume that your CD-ROM drive name is `acd0a` and that you are able to create a temporary directory named `cdrom`. Enter the following commands:

   ```
   mkdir /cdrom
   mount_cd9660 /dev/acd0a /cdrom
   ```

3. The endpoint 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/archive/freebsd/endfbsdr.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`" and press Return.
6. 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
   ```

7. 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.

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
```

When you've completed installation, refer to *Configuring FreeBSD Endpoints* on page 13-5 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. Then save the endpoint to that directory (we'll use `/tmp` in this example).
2. Download the `endfbsdr.tar.Z` file to the `/tmp` directory.
3. Uncompress the endpoint file by using the `uncompress` command:

   ```
   cd /tmp
   uncompress endfbsdr.tar
   tar -xvf endfbsdr.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`" and press Return.

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
```

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.

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 endfbsdr.tar
```

When you've completed installation, refer to *Configuring FreeBSD Endpoints* on page 13-5 to make sure your endpoint is ready to be used in testing and monitoring.

Unattended Installation for FreeBSD

Unattended installation is available for the FreeBSD endpoint. 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
```

What We Do During Installation

Here is what happens during the installation steps. The endpoint is installed into the directory `/usr/local/NetIQ`. 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. They are used by the endpoint as data for `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`.

  For information on tailoring the `endpoint.aud` file for individual endpoints, see Chapter 2, *Endpoint Initialization File*.

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 a reboot.

Our software displays information about updating your system to have the endpoint start automatically upon reboot.

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 Endpoint Package (Uninstall)

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

```
/usr/local/NetIQ/endpoint.remove
```

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

This removes the files from `/usr/local/NetIQ`, except `endpoint.ini`, and does not delete the directory.

Configuring FreeBSD 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 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 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's 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 is one way to determine the IP address of the local computer you are using. Enter the following at a command prompt:

```
netstat -in
```

You may have several network interfaces. If you are using a LAN, for example, look at the output for the xl0 interface; your local IP address is shown in the "Address" column.

Trying Out 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 -c 1 xx.xx.xx.xx
```

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

```
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 (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 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 start automatically when your computer restarts, you must update your system `rc` scripts.

If your FreeBSD system uses `rc.local`, add the following line to the `rc.local` file:

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

Don't forget the ampersand (&) at the end of the line; without it, the boot process will not continue and you may not be able 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.

cp /usr/local/NetIQ/rc2exec.fbd /usr/local/rc.local.d/
81endpointIf your FreeBSD system is configured to use
local_startup (as specified in /etc/rc.conf), copy /usr/
```

```
local/NetIQ/rc2exec.fbd to where local_startup points.
For example, if local_startup points to /usr/local/
rc.local.d, run this command:
```

Running FreeBSD 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.

Starting a FreeBSD Endpoint

The endpoint program should be installed so that it starts automatically each time FreeBSD is rebooted. See the previous topic for information about configuring it properly. The endpoint sends its screen output to file `/var/log/endpoint.console`.

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

```
tail /var/log/endpoint.console
```

The detailed information about the start and stop of each individual connection pair is written to the file `endpoint.aud`. See "Endpoint Initialization File" for a discussion of the `endpoint.aud` file.

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

```
/usr/local/NetIQ/endpoint &
```

The "&" parameter indicates to FreeBSD 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 error message: "`CHR0183: The endpoint program is already running. Only one copy is allowed at a time.`"

Stopping a FreeBSD 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/local/NetIQ/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's been created (so you'll need to take the steps outlined in the following topic).

Cleanup after Unexpected Errors

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

Next, enter the following command:

```
rm /var/log/.NETIQ.ENDPOINT.PID
```

This command removes the file that indicates that the endpoint was active.

Mixed-Platform Performance Issues

Some performance degradation might occur when you are running the FreeBSD endpoint in tests with endpoints running on other operating systems.

We've seen cases where network throughput and response time between FreeBSD and other platforms slows noticeably due to retransmission of data. In one example, if the FreeBSD endpoint TCP/IP stack was required to retransmit data to a Windows 95, Windows 98, or Windows Me endpoint, the data would not be retransmitted for one minute. It appears to stem from an implementation issue within the FreeBSD TCP/IP stack.

Some investigation has revealed that the FreeBSD implementation of RFC 1323 may have incompatibilities with the Windows 95/98 RFC 1323 support. Performance improves if you turn off FreeBSD's RFC 1323 support. Run the command "`sysctl -w net.inet.tcp.rfc1323=0`."

RFC 1323 is described in www.cis.ohio-state.edu/htbin/rfc/rfc1323.html, which describes TCP extensions for high performance.

Unexpected Reboot of FreeBSD Endpoint

You may find that some FreeBSD network device drivers force the system to reboot when not enough mbufs are available. We encountered this problem when running a 200-pair IxChariot test using the xl0 adapter.

You can avoid this problem by increasing the amount of memory available for use as network buffers. To do this, you must recompile the kernel with the following option:

```
options NMBCLUSTERS=2048
```

The exact value depends on how much network traffic is handled by FreeBSD at any given time. 2048 was more than enough to run the 200-pair IxChariot test.

For information on configuring and recompiling the FreeBSD kernel, consult the FreeBSD handbook at the following Web site:

www.freebsd.org/handbook/kernelconfig.html.

How to Tell If a FreeBSD 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 -ax | grep endpoint
```

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

```
13791 v1 s+ 0:53:15 /usr/local/NetIQ/endpoint
```

Disabling Automatic Startup

To disable automatic startup, remove the invocation of the `/usr/local/NetIQ/endpoint` from the `rc.local` file if you are using a FreeBSD system that uses `rc.local`. If your FreeBSD system is configured to use `local_startup`, remove the endpoint script from that directory.

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`.

To view an error log, use the NetIQ program named `FMTLOG`. `FMTLOG` reads from a binary log file and writes its formatted output to `stdout`. Use the following `FMTLOG` command:

```
/usr/local/NetIQ/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: `/var/log/assert.err`

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

When you try to run the FreeBSD endpoint on a computer running the 4.2 or 4.3 revision of the FreeBSD OS, the following error may occur:

```
/usr/libexec/ld-elf.so.1: Shared object "libc_r.so.3" not
found
```

If you see this error, it indicates that the endpoint is looking for a library that is no longer found in the later versions FreeBSD. You need to install the “3.x compatibility” option when you install your 4.x FreeBSD system. See *Installation Requirements for FreeBSD Endpoints* on page 13-1 for more information.

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. See the following topics for information about software updates and support.

Updates for FreeBSD

Check the following Web site for code and driver updates:

www.freebsd.org/

14 SCO UnixWare

The following topics explain the installation, configuration, and operation of the Performance Endpoint software for SCO UnixWare.

The endpoint for SCO UnixWare has been archived at endpoint version 4.2. Therefore, it will not support the latest functionality offered in new releases of IxChariot.

Installation Requirements for SCO UnixWare Endpoints

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

- A computer capable of running SCO UnixWare well. 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 16 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.
- SCO UnixWare version 2.1 or later, with TCP/IP networking and corresponding networking hardware installed and configured. UnixWare version 7.0 is required for 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.

Endpoint Installation for SCO UnixWare

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 instructions below 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. Create a directory named `/cdrom` using the `MKDIR` command:

   ```
   mkdir /cdrom
   ```

3. Next, enter the `MOUNT` command, which tells SCO UnixWare to mount the CD-ROM on the file system (`cdrom`):

   ```
   mount -f cdfs -r /dev/cdrom/* /cdrom
   ```

4. 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 endpoint
   tar -xvf /cdrom/endpoint/archive/unixware/endscor.tar
   ```

5. 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 SCO installation and can be found in the `/usr/bin` directory.

6. You will see the license agreement, presented with the pg command. Press **Enter** 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`."

7. 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.

8. Use the following commands to delete the archive contents from the temporary working directory:

```
cd /tmp
rm -fr endpoint
```

9. Unmount the CD ROM:

```
umount /cdrom
```

This is a good time to read the `README` file, installed with the endpoint in `/opt/NetIQ`, for the latest information about the endpoint program.

When you've completed installation, refer to *Configuring SCO UnixWare Endpoints* on page 14-6 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 `endscor.tar.Z` file to the `/tmp` directory.
3. Uncompress the endpoint file by using the `uncompress` command:

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

4. Next, install the endpoint package using the `pkgadd` command:

```
pkgadd -d /tmp endpoint
```

5. From the directory where you've downloaded the endpoint, run the endpoint's installation script to install our software:

```
./endpoint.install
```

   The `pkgadd` command is not part of the endpoint installation. It is part of the standard SCO installation and can be found in the `/usr/bin` directory.

6. You will see the license agreement, presented with the `pg` command. Press **Enter** 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`."
7. 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 should read, "`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.

8. Use the following commands to delete the archive contents from the temporary working directory:

```
cd /tmp
rm -fr endpoint
rm endscor.tar
```

This is a good time to read the `README` file, installed with the endpoint in `/opt/NetIQ`, for the latest information about the endpoint program.

When you've completed installation, refer to *Configuring SCO UnixWare Endpoints* on page 14-6 to make sure your endpoint is ready to be used in testing and monitoring.

Installation Defaults File for SCO UnixWare

The admin file defines default installation actions to be taken when input is required during install. For example, use the admin file to specify whether to allow a new package to overwrite an older version, or whether an installation can be run with super user authority. 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 over a previous version of the endpoint for SCO UnixWare, 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 SCO UnixWare

Unattended installation is available for the SCO UnixWare endpoint. 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 response file.

First, complete the steps described above, through 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 Enter until the end of the agreement is shown. Next, you are asked whether you accept the terms and conditions of the agreement. If you do, enter "`accept_license`" and press Enter.

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 SCO UnixWare endpoints in unattended mode (this single command is split over two lines):

```
pkgadd -n -a /tmp/endpoint/root/opt/NetIQ/admin
       -r /tmp/endpoint.response -d /tmp endpoint
```

The `pkgadd` command is not part of the endpoint installation. It is part of the standard SCO installation and can be found in the `/usr/bin` directory.

When `pkgadd` is finished, the last line should read, "`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 SCO UnixWare computers.

What We Do During Installation

Here's what happens during the installation steps. The endpoint is installed into the directory `/opt/NetIQ`. 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 a discussion of this file.

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 that 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`). Saf-

estore 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 this program):

```
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 uninstall. When it's finished, the last line reads, "`Removal of <endpoint> was successful.`"

This removes the files from `/opt/NetIQ`, except `endpoint.ini`, and does not delete the directory.

Configuring SCO UnixWare 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 topics explain 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.NetIQ.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's how to determine the IP address of the local computer you're using:

```
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.

Trying Out 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 or domain name of the target computer. If Ping returns a message that says "xx.xx.xx.xx is alive," the Ping test 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 (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 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 SCO UnixWare 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.

Starting an SCO UnixWare Endpoint

The endpoint program is installed so it will start automatically each time SCO UnixWare 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_AUDITING` 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:

```
/opt/NetIQ/endpoint &
```

The "&" parameter indicates to SCO UnixWare 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 additional commands.

If you choose to manually start the endpoint, consider redirecting its output to the `endpoint.console` file, described above. 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 SCO UnixWare 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/NetIQ/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. Avoid using "`kill -9`" to stop the running endpoint program—it doesn't clean up what's been created (so you'll need to take the steps outlined in the following topic).

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.

Next, enter the following command:

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

How to Tell If an SCO UnixWare Endpoint Is Active

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

```
ps -ef | fgrep endpoint
```

If the endpoint program is running, it shows up with the following string in the rightmost column of the output: "`/opt/NetIQ/endpoint`."

Disabling Automatic Startup

To disable automatic startup, remove the `/etc/rc2.d/S81endpoint` 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 NetIQ program named `FMTLOG`. `FMTLOG` reads from a binary log file, and writes its formatted output to `stdout`. Use the following `FMTLOG` command:

```
/opt/NetIQ/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 SCO UnixWare computer, it says that the endpoint program on the SCO has run out of system semaphores. Each instance of endpoint 1 requires two semaphore IDs, with a total of three semaphores. The default system semaphore count is 10.

For example, to increase the number of available system semaphores to allow 100 endpoint pairs, enter the following two commands:

```
/etc/conf/bin/idtune SEMMNI 200
/etc/conf/bin/idtune SEMMNS 300
```

After changing values, you need to rebuild the kernel and reboot to have the changes take effect:

```
/etc/conf/bin/idbuild
cd /
shutdown -y -g0 -i6
```

See the SCO UnixWare answerbook or `MAN` pages for the definitions of these parameters. Specifically, enter the following:

```
man idbuild
man idtune
```

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. See the following topics for information about software updates and support.

Updates for SCO UnixWare

SCO posts code, driver updates, and support tips to the following locations:

- Internet
  - www.sco.com/support/
- Newsgroups
  - comp.unix.sco.misc
  - comp.unix.sco.programmer
  - comp.unix.unixware.misc

15 SGI IRIX

The following topics explain the installation, configuration, and operation of the Performance Endpoint software for the IRIX operating system, by Silicon Graphics, Inc.

The endpoint for SGI IRIX has been archived at endpoint version 4.2. Therefore, it will not support the latest functionality offered in new releases of IxChariot.

Installation Requirements for IRIX Endpoints

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

- A computer capable of running IRIX well. This implies a CPU from Silicon Graphics, Inc., such as the Indy R5000.
- 32 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

  IRIX version 6.2, with TCP/IP networking and corresponding networking hardware installed and configured. This version also supports IP Multicast.

  The following patches to version 6.2 from Silicon Graphics, Inc. are required. If you have not installed these patches, you will not be able to run the endpoint.

  - 1918
  - 2044
  - 2187
  - 2254

  To find out which patches you have installed, select the **System** menu item from the System menu. From the submenu, select the **System Manager** menu

item. The System Manager is shown. Select the **Software Manager** icon. The Software Manager is shown. In the Software Inventory list, all installed patches are shown.

- 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.

Endpoint Installation for IRIX

First, ensure that you are logged in as a "root" user. Also, remember all commands and parameters discussed here are case-sensitive. Use the combination of uppercase and lowercase letters as shown in the examples provided. The instructions below 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.

   We assume that your CD-ROM drive is mounted to the directory named `/CDROM`.

2. The endpoint 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/archive/irix/endirxr.tar
   ```

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

   ```
   ./endpoint.install
   ```

4. 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`" and press Return.

The endpoint installs itself in `/usr/local/NetIQ`. 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.

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 endirxr.tar
```

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 IRIX Endpoints* on page 15-5 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 `endirxr.tar.Z` file to the `/tmp` directory.
3. Uncompress the endpoint file by using the `uncompress` command:

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

4. **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`" and press Return.

The endpoint installs itself in `/usr/local/NetIQ`. 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.

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 endirxr.tar
```

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 IRIX Endpoints* on page 15-5 to make sure your endpoint is ready to be used in testing and monitoring.

Unattended Installation for IRIX

Unattended installation is available for the IRIX endpoint. 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
```

What We Do During Installation

Here is what happens during the installation steps. The endpoint is installed into the directory `/usr/local/NetIQ`. 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 a discussion of the `endpoint.aud` file and how to tailor it 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 a reboot.

Our software creates an S81endpoint initialization script to the `/etc/rc2.d` directory so that the endpoint is started automatically 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, 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 Endpoint Package (Uninstall)

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

```
/usr/local/NetIQ/endpoint.remove
```

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

This removes the files from `/usr/local/NetIQ`, except `endpoint.ini`, and does not delete the directory.

Configuring IRIX 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 for use in tests.
2. Verify the network connections.

The following discussion of TCP/IP configuration explains how to accomplish these tasks.

Determining Your IP Network Address

Here is one way to determine the IP address of the local computer you are using. Enter the following at a command prompt:

```
netstat -in
```

You may have several network interfaces. If you are using a LAN, for example, look at the output for the `ec0` interface; your local IP address is shown in the "Address" column.

Trying Out 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 -c 1 xx.xx.xx.xx
```

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

```
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 (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 Sockets port for TCP/IP 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 IRIX 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.

Starting an IRIX Endpoint

The endpoint program is installed so that it starts automatically each time IRIX 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 /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_AUDITING` 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/NetIQ/endpoint &
```

The "&" parameter indicates to IRIX 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 IRIX 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/local/NetIQ/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's been created (so you'll need to do the steps outlined in the following topic).

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.

Next, enter the following command:

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

How to Tell If an IRIX 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 (the first line, below, is wrapped):

```
root 1240 1239 0 16:23:34 pts/0 0:00 /usr/local/NetIQ/
endpoint
-M -G 1239 -C 1 -T -1
root 1239 1 0 16:23:34 pts/0 0:00 /usr/local/NetIQ/
endpoint
```

Disabling Automatic Startup

To disable automatic startup, remove /etc/rc2.d/S81endpoint.

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/adm/endpoint.log.

To view an error log, use the NetIQ program named FMTLOG. FMTLOG reads from a binary log file, and writes its formatted output to stdout. Use the following FMTLOG command:

```
/usr/local/NetIQ/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: /var/adm/assert.err

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

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. See the following topics for information about software updates and support.

Updates for IRIX

Check the following Web sites for code and driver updates:

www.sgi.com/support/

The required patches for IRIX version 6.2 can be found specifically at:

- http://support.sgi.com/surfzone/patches/patchset/6.2_indigo.rps.html

  You will be asked for an SGI user name and password; this page requires your membership in the Supportfolio Online service, by Silicon Graphics, Inc.)

16 Novell NetWare

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

The endpoint for Novell NetWare has been tested and approved by Novell for the following Novell products:

- NetWare v5.x
- NetWare v4.x
- NetWare v3.12

The endpoint for Novell NetWare has been archived at endpoint version 4.2. Therefore, it will not support the latest functionality offered in new releases of IxChariot.

Installation Requirements for Novell NetWare Endpoints

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

- A computer capable of running Novell NetWare well. This implies a CPU such as an Intel 80386, 80486, a member of the Pentium family, or equivalent. An x486 or better is recommended.
- 8 MBytes of random access memory (RAM); we recommend 16 MBytes of 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 8 MBytes of space available.
- 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.

- Novell NetWare version 6.0, 5.x, 4.x or 3.12. Version 6.0, 5.x or 4.x is required for IP Multicast.

You also need compatible network protocol software:

- **for IPX and SPX**

  IPX and SPX software is provided as part of the NetWare operating system. The files `tli.nlm`, `ipxs.nlm`, and `spxs.nlm` must be loaded for the endpoint to use this support.

  Novell has greatly improved the SPX support in NetWare version 4.x, 5.x, 6.0 and in later fix levels of NetWare version 3.12. Prior to these improvements the SPX stack only supported a window size of 1; buffers are limited to 576 bytes in size. Current versions of NetWare use "SPX II," which removes these limitations--yielding much better performance. We've seen a file transfer script on a 10 Mbps Ethernet give a throughput of about 9 Mbps with SPX II, but only about 6 Mbps with SPX.

- **for TCP and UDP**

  TCP/IP software is provided as part of the network support in the NetWare operating system. The file `tcpip.nlm` must be loaded for the endpoint to use this support.

  NetWare version 4.x, 5.x, 6.0 or later is required for IP Multicast. We recommend that you keep up-to-date with Novell's Minimum Patch List, available from the Novell Web site.

We recommend that you configure your networking software--and make sure that it is working correctly--before installing our software.

See the documentation for your networking software, and see *Configuring Novell NetWare Endpoints* on page 16-6 for more assistance.

We also recommend that you get up-to-date with the latest Novell NetWare service levels.

Updates for Novell NetWare and Clients on page 16-10 discusses how to get the latest service updates.

Endpoint Installation for NetWare

If you have a version of the endpoint already running, be sure to unload it from the NetWare system console before installing a new one.

See *Stopping a Novell NetWare Endpoint* on page 16-9 for details.

You must install the endpoint from the World Wide Web, from a CD-ROM, or from diskettes, from any connected Windows or IBM OS/2 NetWare client. The file `SETUP.BAT` is designed to run from a NetWare client computer, not directly from the actual NetWare server.

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

1. Download the endpoint file `nlm35.zip` to a local hard drive.
2. Unzip the file. For an IBM OS/2 install, use the OS/2 unzip utility, available at ftp://service.boulder.ibm.com/ps/products/os2/rsu/unzip.exe. For a Windows install, use the WinZip utility. See *Using WinZip* on page 9-5 for more information.
3. At a command prompt, go to the drive where you've saved the endpoint and enter `SETUP`, followed by the drive and path on the NetWare server where you want to install the endpoint:

   ```
   SETUP drive:path
   ```

   where "`drive:path`" indicates a drive and path on the NetWare server where you want the endpoint installed. Don't end the path with a slash or backslash.

   ```
   SETUP N:\NETIQ
   ```

Now skip to *Completing Installation* on page 16-4.

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

1. Put the Performance Endpoint CD-ROM in your CD-ROM drive. From a command prompt, go to the drive and path where the CD-ROM is located:

   ```
   drive:
   CD\ENDPOINT\ARCHIVE\NETWARE
   ```

   where "`drive:`" is your CD-ROM drive.
2. Next, enter `SETUP`, followed by the drive and path on the NetWare server where you want to install the endpoint:

   ```
   SETUP drive:path
   ```

   where "`drive:path`" indicates a drive and path on the NetWare server where you want the endpoint installed. Don't end the path with a slash or backslash.

   ```
   SETUP N:\NETIQ
   ```

Now skip to *Completing Installation* on page 16-4.

To install the endpoint from diskettes, do the following:

Installing from diskette is a little complicated because the endpoint file will not fit on one diskette. Diskettes can be made from the CD-ROM by copying the `disk1` through `diskX` directories to diskettes. Then carefully follow these instructions:

1. Put the first installation diskette in your diskette drive. From a command prompt, go to the drive where the diskette is located:

   ```
   drive:
   ```

   where "`drive:`" indicates your 3.5 inch diskette drive.
2. Next, enter `SETUP`, followed by the drive and path on the NetWare server where you want to install the endpoint:

```
SETUP drive:path
```

where "drive:path" indicates a drive and path on the NetWare server where you want the endpoint installed.

Now skip to *Completing Installation*.

Completing Installation

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" and press Return.

We recommend installing the endpoint in a directory named \NetIQ on the appropriate drive of your NetWare server. The copying of files is now complete; you can remove the CD-ROM or diskette from its drive.

The endpoint program is not running when the installation completes. You can start the endpoint manually:

- If you have installed the endpoint from a remote workstation using the NetWare Requester, enter:

  ```
  LOAD d:\path\ENDPOINT.NLM
  ```

  where d: and path are the full drive and path on NetWare where the endpoint is installed.

- If you have installed directly from the DOS command prompt on the NetWare server, enter:

  ```
  LOAD SYS:\path\ENDPOINT.NLM
  ```

  where path is the full path on NetWare where the endpoint is installed.

However, you'd probably prefer to have the endpoint start automatically each time NetWare is started. To do this, insert the preceding statement at the end of your AUTOEXEC.NCF file, so that the endpoint starts after all network drivers and devices are loaded.

You should see a message like the following when the endpoint is active and waiting to run a test:

```
Endpoint, Version 4.2
Copyright NetIQ Corp., 1995-2001
Build level: xxx

Processing INI file (SYS:\NETIQ\endpoint.ini).
Endpoint INI information in use:

All available protocols are enabled.
All consoles may run tests on this endpoint.
Security Auditing: NONE
Audit filename: endpoint.aud

Support for TCP has been started.
Support for SPX has been started.
The local SPX address is 330897F1:000000000001
```

The order in which these messages appear may differ depending on which network protocol completes initialization first. "`Support for TCP`" means that the endpoint is ready to run tests that use the RTP, TCP, and UDP protocols; "`Support for SPX`" means that it's ready for both IPX and SPX tests. 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.

See *Configuring Novell NetWare Endpoints* on page 16-6 for information about your network connections.

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

Unattended Installation for NetWare

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

Run the endpoint's installation, adding the "`accept_license`" parameter:

```
SETUP drive:path accept_license
```

What Happens During Installation

Here's what happens during the installation steps. Let's say you install the endpoint into the directory `\NetIQ`. 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 installation process for a Novell NetWare endpoint makes no changes to `AUTOEXEC.NCF`.

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 Endpoint Package (Uninstall)

If you need to remove the endpoint package from your hard disk, you must first stop the endpoint program (if it is running). To remove the endpoint, delete the directory where you installed the Novell NetWare endpoint. After you have deleted the directory, the endpoint is no longer installed on the computer.

Configuring Novell NetWare Endpoints

The endpoint program runs as an application, using the network application programming interfaces, such as TLI, for all of its communications. The endpoint dynamically configures its own programs, so you do not have to edit 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 describe how to accomplish these steps for Novell NetWare:

- *NetWare Configuration for IPX and SPX*
- *NetWare Configuration for TCP/IP* on page 16-7

NetWare Configuration for IPX and SPX

To use the IPX or SPX protocol in tests, IPX addresses must be supplied as the network address at the IxChariot or Qcheck Console 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.

A NetWare server has two 6-byte node addresses:

- An *internal* address, which is always `000000000001`.
- An *external* address, which is usually the same as the MAC address of the LAN adapter you are using. Our software cannot connect to a NetWare endpoint via its external address.

Our software makes calls to the TLI programming interface when using the IPX or SPX network protocol.

Determining the Local IPX Network Addresses

To connect to an endpoint running on a NetWare server with IPX or SPX, you need to know its internal IPX address. The internal address is set during startup, by the `IPX INTERNAL NET` command. This command can be found in one of the NCF files run during startup (usually it's in the `AUTOEXEC.NCF` file):

- 4-byte network address: follows **ipx internal net**
- 6-byte node address: always `000000000001`

For example, if the IPX Internal Net is `323DC670`, the internal address is `323DC670:000000000001`.

Another way to find this internal address is to load the `MONITOR` program at the server. Look in the "Connection Information" section, under the entry for the server itself. You'll see an address that looks like this: `323DC670:000000000001:0007`. The internal address is everything preceding the second colon.

When connecting to a NetWare endpoint, always use its internal address.

Be sure to read *Running Novell NetWare Endpoints* on page 16-8 to understand the limitations and environment of a NetWare endpoint.

Sockets Port Number for IPX and SPX

IPX and SPX applications use their network addresses 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 port for IPX and SPX is **10117**.

NetWare 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

The local IP address for a NetWare server is set when the TCP/IP protocol is bound to the adapter. This is done during startup, using the `BIND` command. The `BIND` command is most likely in the `AUTOEXEC.NCF` or `INITSYS.NCF` file on your server. View the appropriate NCF file to determine the TCP/IP address of the NetWare server.

The local IP address is also shown in the `\ETC\CONSOLE.LOG` file.

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 the following at an MS-DOS command prompt of a client Windows or OS/2 computer:

```
ping xx.xx.xx.xx
```

Replace the `x`'s with the IP address of the target computer. If Ping succeeds, there's a route between the computer you are using and the NetWare server.

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 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 Novell NetWare Endpoints

The following sections describe starting and stopping a Novell NetWare endpoint, as well as some of the messages and information that become available during testing with this endpoint.

Starting a Novell NetWare Endpoint

You should see a message like the following at an endpoint when it is active and waiting to run a test:

```
Support for SPX has been started.
The local SPX address is 330897F1:000000000001
Support for TCP has been started.
```

If you stop `ENDPOINT.NLM` and need to restart it without restarting Novell NetWare, you can load the endpoint manually.

- If you have installed the endpoint from a remote workstation using the NetWare Requester, enter:

  ```
  LOAD d:\path\ENDPOINT.NLM
  ```

  where `path` is the full path on NetWare where the endpoint is installed.

- If you have installed directly from the DOS command prompt on the NetWare Server, enter:

  ```
  LOAD SYS:\path\ENDPOINT.NLM
  ```

  where `d:` and `path` are the full drive and path on NetWare where the endpoint is installed.

A single running copy of `ENDPOINT.NLM` handles one or multiple concurrent tests. If the endpoint program is already running and you try to start another copy, you get the following message, "`The module is already loaded and cannot be loaded more than once.`"

Loading `ENDPOINT.NLM` on NetWare 3.12, without first loading `IPXS.NLM`, causes the NetWare server to crash. (Since you need `IPXS.NLM` loaded to do about just anything on a NetWare server, this condition should be rare.) NetWare 5.x and 4.x do not allow `IPXS.NLM` to be unloaded.

Stopping a Novell NetWare Endpoint

To stop the endpoint program, unload it. You can unload the NLM by entering the following at a NetWare system console:

```
UNLOAD ENDPOINT.NLM
```

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 directory where you installed our software. To view an error log, use `FMTLOG`. The version of `FMTLOG` installed on your Novell NetWare server runs as a DOS program. You can thus run it from any OS/2 or Windows computer. `FMTLOG` reads from a binary log file, and writes its formatted output to `stdout`. Use the following `FMTLOG` command:

```
FMTLOG \NetIQ\endpoint.log > your_output_filename
```

In addition, the endpoint code does a lot of internal checking on itself. You may see a `Failed assertion` message in the endpoint window. If you encounter an assertion failure, please write down the sequence of things you were doing when it occurred. Our software captures details related to the problem in an ASCII text file named `assert.err` in the directory where you installed the endpoint. Save a copy of the file and send it back to us via email for problem determination.

Limitations of the Novell NetWare Endpoint

The endpoint is limited by the amount of RAM installed in the NetWare server. If there is insufficient memory, tests with a large number of connection pairs can cause the endpoint NLM to fail:

- With 48 MBytes of RAM, up to 64 concurrent connection pairs are supported.
- With 16 MBytes of RAM, up to 16 concurrent connection pairs are supported.

Use the NetWare MONITOR program to keep on eye on the memory being consumed by our software and other concurrently running programs.

The shortest **SLEEP** period on NetWare endpoints is $1/18^{th}$ second, that is, 55 ms. Thus, if you're setting a constant **SLEEP** time for a script that will use NetWare, multiples of 55 are the most efficient due to the granularity of its clock.

- On NetWare 4.x, with a 10 Mbps Ethernet card:

  Examine your `STARTUP.NCF` file. Make sure it contains the following line:

  ```
  SET Maximum Physical Receive Packet Size = 1520
  ```

  Without that line, the default size is 4202 bytes. This seems to be the wrong value for use with Ethernet's buffering. Tests using IPX or UDP will fail with a timeout and the wrong error message when the scripts' `send_buffer_size` is less than 4202 bytes (but greater than 1520).

- On a NetWare server slower than a Pentium 100 MHz:

  Examine your `STARTUP.NCF` file. Make sure it contains the following line:

  ```
  SET Maximum Packet Receive Buffers = 2000
  ```

- Without that line, we've observed that the default size is 100. This seems to be the wrong value for use on slow computers, which have trouble keeping up with incoming packets. Running tests with extensive traffic can cause the NetWare server to lock up, requiring rebooting.

Novell's fix STRTL5 for NetWare 4.x has a bug that causes tests with short SPX connections, such as the `Credits` script, to hang. Get the latest fix, named STRTL6, at the following Web site: support.novell.com/misc/patlst.htm.

Novell's TCP/IP stack fix named TCPN04 has a limitation of 32 simultaneous connections. Get the latest stack fix, named TCPN05. See http://support.novell.com/misc/patlst.htm.

Updates for Novell NetWare and Clients

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 latest software for the underlying operating system and communications software.

Novell posts code and driver updates directly to the following Web site: http://support.novell.com/.

17 Spirent Communications TeraMetrics

The following topics explain the installation, configuration, and operation of the Performance Endpoint software for Spirent Communications' TeraMetrics™ product family.

The endpoint for Spirent TeraMetrics has been archived at endpoint version 4.3. Therefore, it will not support the latest functionality offered in new releases of IxChariot and End2End.

The TeraMetrics endpoint is designed specifically to work with Spirent Communications (formerly Netcom Systems) TeraMetrics modules. TeraMetrics modules, part of the SmartBits® suite of products, allow for traditional SmartBits testing, as well as providing access to third-party applications running on a subset of the Linux OS. Using IxChariot in conjunction with the Spirent Communications SmartBits suite of software and the TeraMetrics endpoint allows for unprecedented network analysis and component testing capability.

By default, the TeraMetrics Linux operating system allows read-only access to all volumes except the `/tmp` directory. Therefore, all dynamic content created by the endpoint (such as `endpoint.console`, `endpoint.log`, etc.) is redirected to the `/tmp` directory.

Installing TeraMetrics Endpoints

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

- A Spirent Communications LAN-3301A or LAN-3311A, or any TeraMetrics module
- A Spirent Communications SMB-600 or SMB-6000B Chassis.

RPM-Based Endpoint Installation for TeraMetrics

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. Install the Download Manager on the endpoint computer, if you have not already done so. Download Manager and its documentation can be obtained from Spirent at www.spirentcom.com/.
2. Put the CD-ROM in your CD-ROM drive.
3. Start the Download Manager software.
4. Configure your connection to the SmartBits chassis, and connect.
5. Select the **Software** view from the Download Manager window.
6. Reserve the TeraMetrics module(s) where you want to install the endpoint.
7. Find the `endteram.rpm` file in the `Endpoint\Teramet` directory. Download the file onto the TeraMetrics module(s). When the download completes, the TeraMetrics module(s) restart, and the endpoint starts running.

To install an endpoint you're downloading from the World Wide Web, do the following:

The endpoint must be installed using the Download Manager from the SmartBits software suite. To install the endpoint, take the following steps:

1. Install the Download Manager on the endpoint computer, if you have not already done so. Download Manager and its documentation can be obtained from Spirent at www.spirentcom.com/.
2. Download the endpoint RPM file (`endteram.rpm`) from the World Wide Web. The latest endpoint versions are always available at www.ixiacom.com/ixchariot/download/endpoints.
3. Start the Download Manager software.
4. Configure your connection to the SmartBits chassis, and connect.
5. Select the **Software** view from the Download Manager window.
6. Reserve the TeraMetrics module(s) where you want to install the endpoint.
7. Download `endteram.rpm` onto the TeraMetrics module(s). When the download completes, the TeraMetrics module(s) restart, and the endpoint starts running.

Removing the RPM Endpoint Package (Uninstall)

The endpoint must be removed using the Download Manager from the SmartBits software Suite. **To uninstall the endpoint, use the following steps:**

1. Start the Download Manager software.
2. Configure your connection to the SmartBits chassis, and connect.
3. Select the **Software** view from the Download Manager window.
4. Select the TeraMetrics module(s) from which you want to remove the endpoint.
5. Select the **On Board** tab to display the packages currently installed on the TeraMetrics module.
6. Select the endpoint package, then right-click and select **Erase RPM**. The TeraMetrics module(s) restart and the endpoint is removed.

What We Do During Installation

Here is what happens during the installation steps. The endpoint is installed into the directory `/usr/local/NetIQ` on the Linux OS of the TeraMetrics module. 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`. This file lets you configure the endpoint. See Chapter 2, *Endpoint Initialization File* for a full discussion.

The file `/usr/netcom/autorun/NetIQ` is created. This file autostarts the endpoint via a script added to the `usr/Netcom/autorun` directory.

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.

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

Configuring TeraMetrics 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. IP addresses for TeraMetrics module ports are assigned through the debug console of the SmartBits software application SmartWindows, or by telnetting to the port as described in the *TeraMetrics Software Developer's Guide*.
2. 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
```

Depending on your configuration and the number of ports per TeraMetrics module, you may see several IP addresses.

Trying Out 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 port.

Make sure that you can run Ping successfully from the IxChariot or Qcheck Console or the End2End server 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 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 TeraMetrics 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 TeraMetrics Endpoint

The endpoint program is installed so that it starts automatically each time TeraMetrics is restarted. The autostart file `/usr/netcom/autorun/NetIQ` is created during installation. This file is executed automatically during startup of the TeraMetrics module(s).

The endpoint program sends its screen output to file `/tmp/endpoint.console`.

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

```
tail -f /tmp/endpoint.console
```

or

```
cat /tmp/endpoint.console
```

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/NetIQ/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 titled *How to Tell If a TeraMetrics Endpoint Is Active* on page 17-6 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 `/tmp/NETIQ.ENDPOINT.PID` by using the `ls -a` command. This file should be manually removed.

Stopping a TeraMetrics 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/NetIQ/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 below).

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 /tmp/.NETIQ.ENDPOINT.PID
```

How to Tell If a TeraMetrics 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/NetIQ/endpoint
367 p0 S 0:00 | \_ /usr/local/NetIQ/endpoint
368 p0 S 0:00 | \_ /usr/local/NetIQ/endpoint
369 p0 S 0:00 | \_ /usr/local/NetIQ/endpoint
```

Disabling Automatic Startup

Disabling automatic startup requires modifying `/usr/Netcom/autorun`, a read-only file. Contact Spirent Communications for assistance.

Logging and Messages

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

```
/tmp/endpoint.log
```

```
/usr/local/NetIQ/fmtlog /tmp/endpoint.log >/tmp/
output_filenameTo 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(s):
```

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

```
/tmp/assert.err
```

Save a copy of the file and email it to NetIQ Technical Support for problem determination.

Message CHR0181

You may receive message CHR0181 while running a test. If the error was detected at the TeraMetrics computer, it says that the endpoint program on TeraMetrics has run out of system semaphores. Each instance of Endpoint 1 requires a system semaphore. The maximum number of semaphores is not configurable on TeraMetrics, 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 software for the underlying operating system and communications software.

Check the following Web site for code and driver updates: www.netcomsystems.com/support/softwareupdates.asp.

Contacting Spirent for TeraMetrics Module Support

To obtain technical support for any Spirent Communications, SmartBits Division product, please contact the Technical Support Department using any of the following methods:

Table 17-1. Spirent Contact Information

e-mail	support@spirentcom.com
telephone	800-886-8842 or 818-676-2589 (between 07:00 and 18:00 PST)
fax	818-880-9154

A Archived Endpoint Specifications

This appendix describes the IxChariot Performance Endpoints that have been archived. It contains the following topics:

Operating System and Protocol Stack Support

Table A-1 identifies the supported operating system and protocol stack software for the archived endpoints. The table lists the software with which we have tested the archived Performance Endpoints for each operating system.

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

Table A-1. Archived Endpoints - Operating System Compatibility

Archived Endpoint	OS version	TCP, UDP, RTP	IP Multicast version	IPX/SPX stack
Cobalt RaQ/RaQ2 (MIPS)	Linux v. 2.0 for MIPS	included	kernel 2.0.32	no
Cobalt RaQ3 (x86)	kernel 2.0.32	included	kernel 2.0.32	no
Compaq Tru64 UNIX	Digital UNIX 4.0B or Compaq Tru64 Unix for Alpha	included	v4.0B	no
FreeBSD UNIX	BSD v3.1	included	v3.1	no
HP-UX	HP-UX v11.0	included	v11.0	no
IBM AIX	AIX v4.1.4	included	v4.1.4	no

Table A-1. Archived Endpoints - Operating System Compatibility (Continued)

Archived Endpoint	OS version	TCP, UDP, RTP	IP Multicast version	IPX/SPX stack
IBM MVS	MVS/ESA SP v4R2.2	See "*MVS TCP/IP Stacks* on page A-3"	no	no
IBM OS/2	OS/2 Warp 4, Warp Connect 3	Download TCP 4.1	Download TCP 4.1	Download Novell Netware Client v2.12
Linux IA-64 (TurboLinux)	kernel 2.4.0test7-42	included	kernel 2.4.0test7-42	no
Microsoft Windows 3.1	Windows 3.1 or Windows for Workgroups 3.11	see "*Microsoft Windows 3.1 TCP/IP Stacks* on page A-2"	Chameleon 7.0, as E2	no
Microsoft Windows 95	Windows 95	included	no	Download Novell Netware Client v3.21
Microsoft Windows 95 with WinSock 2	Windows 95 with WinSock 2 installed	Download WinSock 2	included	included
Microsoft Windows 98	Windows 98	included	included	included
Microsoft Windows CE 4.X	Windows CE 4.2, 4.3	included	included	no
Microsoft Windows Millennium Edition (Me)	Windows Me	included	included	included
Microsoft Windows NT 4 for Alpha	Windows NT4 SP 3	included	SP3 (IGMPv1) SP4 (IGMPv2)	included
Microsoft Windows XP 64-bit Edition (IA-64 processors)	Windows XP (64-bit)	included	included	no
Novell NetWare	v3.12	included	v4.0	included
SCO UnixWare	UnixWare v2.1	included	v7.0	no
SGI IRIX	IRIX v6.2 with patches	included	v6.2	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.

Performance Endpoint Support for IxChariot Functions

The following table describes the capabilities of the archived Performance Endpoints. These endpoints may not support new functionality in the latest versions of IxChariot.

Table A-2. Archived Performance Endpoint Capabilities per OS

Endpoint OS	IP QoS (DiffServ, GQOS, TOS)	Trace-route	CPU Utiliti-zation	VoIP Tests	Video Pair Tests	IPv6 Tests	802.11 Statistics
Cobalt RaQ or RaQ2 (MIPS)	TOS	No	Yes	No	No	No	No
Cobalt RaQ3 (x86)	TOS	Yes	Yes	Yes	No	No	No
Compaq Tru64 UNIX	TOS	No	Yes	No	No	No	No
FreeBSD UNIX	TOS	No	Yes	No	No	No	No
HP-UX	DiffServ, TOS	Yes	No	Yes	No	No	No
IBM AIX	DiffServ, TOS	Yes	No	Yes	No	No	No
IBM MVS	No	No	No	No	No	No	No
IBM OS/2	TOS	No	Yes	No	No	No	No
Linux IA-64 (TurboLinux)	TOS	Yes	Yes	No	Yes	No	No
Microsoft Windows 3.1	No	No	No	No	No	No	No

Table A-2. Archived Performance Endpoint Capabilities per OS (Continued)

Endpoint OS	IP QoS (DiffServ, GQOS, TOS)	Trace-route	CPU Utiliti-zation	VoIP Tests	Video Pair Tests	IPv6 Tests	802.11 Statistics
Microsoft Windows 95	No	No	Yes	No	No	No	No
Microsoft Windows 95 with WinSock 2	TOS (UDP, RTP)	Yes	Yes	No	No	No	No
Microsoft Windows 98	GQOS (RSVP), TOS (UDP, RTP)	Yes	Yes	Yes	No	No	No
Microsoft Windows CE 4.X	No	No[a]	Yes	Yes	No	No	Yes[b]
Microsoft Windows Me	GQOS (RSVP)	Yes	Yes	Yes	No	No	No
Microsoft Windows NT 4 for Alpha	No	Yes	Yes	No	No	No	No
Microsoft Windows 98 (Web-Based)	Yes	No	Yes	Yes	No	No	No
Microsoft Windows XP 64-bit Edition (IA-64)	DiffServ, GQoS, TOS	No	No	Yes	Yes	No	No
Novell NetWare	No	No	No, v3.12; Yes, v4.0	No	No	No	No
SCO UnixWare	TOS (bits 3-5)	No	No	No	No	No	No
SGI IRIX	TOS	No	Yes	No	No	No	No

a.Support for CPU Utilization on Windows CE is device-dependent. For more information, see http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wcemain4/html/cerefGetIdleTime.asp.
b.Windows CE version 4.20 or later.

Endpoint Computer Resource Guidelines

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.

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 ("n/a" indicates that the protocol is not supported by the endpoint). The shaded rows list archived endpoints.

Table A-3. Calculating Memory Requirements

Operating System	Base RAM (in KB)	TCP KB/ pair	UDP KB/ pair	RTP KB/ pair	SPX KB/ pair	IPX KB/ pair
HP-UX	844	140-150	257-292	158-207	n/a	n/a
IBM AIX	1176	132-284	146-296	146-296	n/a	n/a
MVS	666	25-48	24-52	24-52	n/a	n/a
NetWare	1100	80-110	320-340	320-340	70-100	260-280
OS/2	1096	50-65	150-170	150-170	315-340	150-170
Windows 3.1	550	72-600	72-600	72-600	n/a	n/a
Windows 95/98/Me	1100	40-65	100-145	100-145	40-65	55-75
Windows CE 4.X	277	44-70	196-436	700-773	n/a	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 *IxChariot 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. The shaded rows list archived endpoints.

Table A-4. Endpoint Pair Capacity

Operating System	Installed RAM	TCP pairs	RTP or UDP pairs	SPX pairs	IPX pairs
HP-UX	1 GB	200	150	n/a	n/a

Table A-4. Endpoint Pair Capacity (Continued)

IBM AIX 4.1	1 GB	200	180	n/a	n/a
NetWare 4.12	64 MB	500	200	100	100
OS/2 4.0	32 MB	500	200	20	20
Windows 3.1	8 MB	1	1	n/a	n/a
Windows 95/98/Me	16 MB	18	100	40	175
Win64 (Itanium based)	768 MB	175	120	n/a	n/a
Windows CE 4.X	56 MB	85	35	n/a	n/a

Notes:

- 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.

Index

F

H

I

T

U

V

W