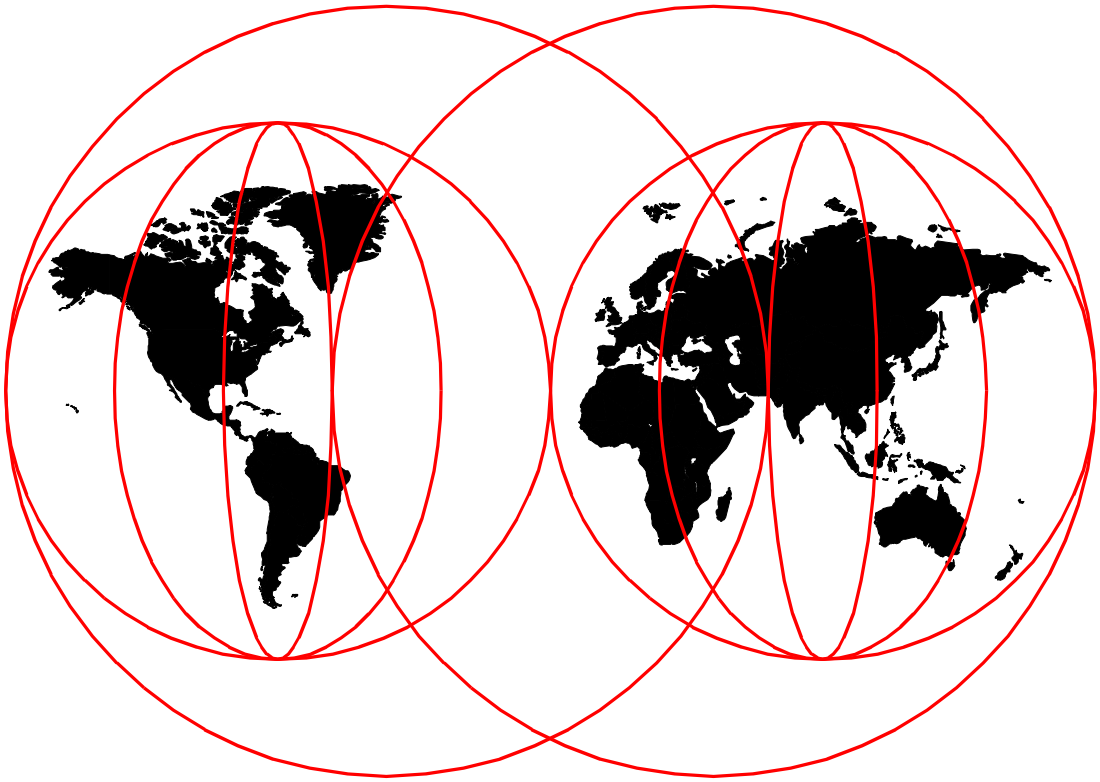
V4 TCP/IP for AS/400: More Cool Things Than Ever

Fant Steele, Brendan Kay, Palle Lyckegaard, Linda Morrison, Gerald (Jerry) Pape, Tom Vernaillen



International Technical Support Organization

www.redbooks.ibm.com

SG24-5190-00

International Technical Support Organization

SG24-5190-00

V4 TCP/IP for AS/400: More Cool Things Than Ever

March 2000

Take Note!

Before using this information and the product it supports, be sure to read the general information in Appendix B, “Special notices” on page 691.

First Edition (March 2000)

This edition applies to Version 4 of OS/400.

Comments may be addressed to:
IBM Corporation, International Technical Support Organization
Dept. JLU Building 107-2
3605 Highway 52N
Rochester, Minnesota 55901-7829

Contents

Figures

Tables

Preface

This redbook provides sample scenarios that demonstrate common solutions with example configurations. It covers all versions and releases of OS/400 V4 up to V4R4. In some cases, the examples are provided at lower-level versions. Where there are significant differences, multiple versions are presented. The information in this redbook helps you plan, install, tailor, configure, and troubleshoot TCP/IP on your AS/400 system. The intended audience for this redbook includes the analysts, consultants, and support people that will design, install, and configure the AS/400 system in a TCP/IP environment.

This redbook is a compendium of AS/400 TCP/IP information. It is built from information contained in other existing redbooks which are referenced throughout this book, as well as new information developed specifically for this book. We provide enough basic information about the common TCP/IP functions to allow the reader to setup the TCP/IP functions on an AS/400 system. In some cases, we provide more in-depth information, while, in other cases, we refer you to another source of more detailed documentation that goes beyond the basics. Some knowledge of the AS/400 platform and TCP/IP is assumed.

The team that wrote this redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization Rochester Center.

Fant Steele is a Senior I/T Technical Specialist on the e-business team in the AS/400 Advanced Technical Support Group located in Rochester, MN. Prior to joining this group, Fant spent two and a half years as an Advisory ITSO Specialist for AS/400 in the International Technical Support Organization (ITSO), Rochester Center. While in the ITSO, Fant wrote several redbooks on communication and e-business topics. Fant also taught classes and seminars worldwide on many areas of AS/400 communications technologies and e-business. He spent eight years as an instructor and developer for the AS/400 communications and programming curriculum of IBM Education and Training. Prior to joining IBM in 1989, he worked on S/36 to AS/400 code conversion, VM/MVS systems programming, and applications programming for the manufacturing industry.

Brendan Kay is a Communications Development Specialist for Look Software in Australia. He has 14 years of experience in S/390, AS/400, and PC communications development and support. He holds a degree in Computer Science from Monash University in Melbourne, Australia. His areas of expertise include TCP/IP communications and networking on PCs and the AS/400 system.

Palle Lyckegaard is a technical consultant for business partner EDB Gruppen Systems A/S in Denmark. He has eight years of experience in the AS/400, PC, and networking fields. His areas of expertise include AS/400, communications, networking, and especially TCP/IP across all platforms.

Gerald (Jerry) Pape is a I/T Specialist in Austria. He has six years of experience with the IBM AS/400 system. His areas of expertise include AS/400 network connectivity and communications, TCP/IP, and Client Access. Gerald has written presentations and taught international classes covering Client Access and Windows NT on the IPCS.

Tom Vernaillen is an I/T specialist in IBM Belgium. He has seven years of experience with the IBM AS/400 system. His areas of expertise include AS/400 network connectivity and communications, including TCP/IP.

Linda Morrison is an Advisory Instructor and Developer for the AS/400 Communications curriculum of IBM Education and Training. She has 21 years of experience with IBM S/390, S/36, and AS/400 communications and support. She has a degree in Electronic Engineering. Linda has five years of experience with IBM Support Line communication in Rochester, MN. Her areas of expertise include SNA, TCP/IP, and firewall.

Thanks to the following people for their invaluable contributions to this project:

Marcela Adan
Suehiro Sakai
International Technical Support Organization, Rochester Center

Rob Simonson
IBM Rochester

Ed Boden
Gary Diehl
Don Gillespie
Franklin Gruber
Susan Hall
Thomas Murphy Jr
Jeffrey Stevens
The rest of the TCP/IP Team
IBM Endicott

Comments welcome

Your comments are important to us!

We want our Redbooks to be as helpful as possible. Please send us your comments about this or other Redbooks in one of the following ways:

- Fax the evaluation form found in “IBM Redbooks evaluation” on page 709 to the fax number shown on the form.
- Use the online evaluation form found at http://www.redbooks.ibm.com/
- Send your comments in an Internet note to redbook@us.ibm.com

Chapter 1. Introduction to TCP/IP on the AS/400 system

Transmission Control Protocol/Internet Protocol (TCP/IP) is the protocol suite developed for the Internet. The protocol suite may be used to connect multiple systems from many different vendors together and allow them to communicate meaningfully. The TCP/IP protocol suite is made up of many components, including applications, transmission protocols, routing protocols, and so on. There is a set of components that is part of the core of TCP/IP. Other components are considered optional. It is up to each vendor to decide which optional components are implemented in their TCP/IP product.

TCP/IP communications is done across an *internet*. An internet is made up of one or more networks joined together using routers, switches, or other network hardware. The most famous internet is the Internet (big I), which is also referred to as the World Wide Web. An internet can also be a private internet known as an Intranet. While TCP/IP is the communications mechanism used in the Internet, you do not have to be connected to the Internet to use TCP/IP. We recommend that you read the book *TCP/IP Tutorial and Technical Overview,* GG24-3376, to learn more about the components of TCP/IP and the functions provided by each standard TCP/IP application.

You can provide a wide range of services to your users with TCP/IP as the basis for your network. You can:

- Put your AS/400 server on the Internet and take advantage or emerging e-commerce opportunities.
- Use your AS/400 server as a file and print server, so users can have a central repository for the files and print services.
- Run workgroup and collaboration software such as Lotus Notes Domino to harness the knowledge of your organization.
- Allow users who are in the field to have a connection and access to the corporate network.
- Enable users on PCs to communicate with one another or with a server.

The AS/400 implements many TCP/IP functions in V4. Depending upon the version you are running, some functions may not be available. Refer to Table 1 to determine which release is required for the function you want to perform.

Table 1. TCP/IP function availability reference

Function	Short description	V4 R1	V4 R2	V4 R3	V4 R4
TELNET	Allows logins from one system to another. AS/400 TCP/IP supports both the Telnet client and server. Use Operations Navigator to manage Telnet sessions, sign-ons, and configuration.	X	X	X	X
	Telnet 5250 extensions for printing, device name selection, and automatic sign-on.		X	X	X

Function	Short description	V4 R1	V4 R2	V4 R3	V4 R4
FTP	Transfer files between servers and clients across a TCP/IP network. AS/400 TCP/IP supports both the FTP client and server. Use Operations Navigator to administer FTP formats, mappings, and startups.	X	X	X	X
	Anonymous FTP. Implemented using exit programs.	X	X	X	X
	Passive FTP. Firewall friendly FTP.		X	X	X
PING	Packet InterNet Groper (PING) is a tool that can be used to check connectivity to a system. Type `PING` or `VFYTCPCNN` on the command line.	X	X	X	X
DNS	Translate domain names to IP address. AS/400 TCP/IP supports DNS. Operations Navigator contains an easy-to-use DNS server configuration display that allows you to work with primary and secondary DNS names for your network.		X	X	X
NSLOOKUP	Tool to query DNS servers for information. Call a program in V4R2. Use the command NSLOOKUP for V4R3 and beyond.		X	X	X
VPN	Make private links on the Internet. Create secure end-to-end paths between any combination of hosts and gateways with Operations Navigator. VPNs use authentication methods, encryption algorithms, and other precautions to ensure that data sent between the two endpoints of its connection remains secure.				X
	Native VPN support including remote access, branch office, and extranet scenarios.				X
IP Packet Security	Protect the AS/400 on the Internet with IP packet security. It contains two components, IP Packet Filtering and Network Address Translation, that act as a “firewall” of protection for your system. Operations Navigator provides extensive access to IP Packet Security functions.			X	X
	IP Packet auditing and journaling.		X	X	X
	IP Network Address Translation.			X	X
	IPCS firewall with IP security and VPN support.			X	X
	Filtering enhancements.				X

Function	Short description	V4 R1	V4 R2	V4 R3	V4 R4
LPR and LPD	LPR sends files to print to any AS/400 system over TCP/IP. LPD places print files on a print queue on any AS/400 system in your network. AS/400 supports these network printing protocols for ASCII and LAN-attached printers. Use Operations Navigator to display and manage LPR and LPD properties.	X	X	X	X
SMTP	Sends and receives e-mail on the AS/400 system. Use the AS/400 SMTP capabilities to setup an extensive e-mail system with your AS/400 system as the e-mail server. Operations Navigator allows easy and intuitive SMTP configuration and management.	X	X	X	X
POP server	Stores and forwards e-mail for e-mail client program connected to the AS/400. AS/400 supports POP by creating mailboxes for all enrolled e-mail users. AS/400 POP server allows users the ability to access their mailboxes from third-party e-mail client programs.	X	X	X	X
PPP	Allows system-to-system connection and data exchange through a modem and over a leased or telephone line. AS/400 supports point-to-point connections as a part of its wide area network (WAN) connectivity.	X	X	X	X
	PPP for WAN links. Operations Navigator includes a complete point-to-point interface for management and configuration.		X	X	X
	PPP dial-on-demand for WAN links, including frame relay.		X	X	X
SLIP	Allows system-to-system connection and data exchange over serial lines. Use Operations Navigator to configure and administer SLIP connections. We recommend using PPP rather than SLIP. Both are supported.	X	X	X	X
LDAP	Runs a directory service over TCP/IP. AS/400 Directory Services provides an LDAP server on the AS/400 system. LDAP runs over TCP/IP and is a popular choice as a directory service for both Internet and non-Internet applications. Use Operations Navigator to perform most setup and administration tasks of the LDAP directory server on the AS/400 system.			X	X

Function	Short description	V4 R1	V4 R2	V4 R3	V4 R4
AS/400 NetServer	Enables Windows clients using Network Neighborhood to access AS/400 shared directory paths and output queues, such as printers and file systems. AS/400 supports PC clients using a network file and print sharing functions that is included in their operating systems. This means that you do not need to install any additional software to benefit from AS/400 NetServer. You can administer and manage AS/400 NetServer with Operations Navigator.		X	X	X
HTTP server (formerly known as ICS)	Allows AS/400 systems attached to the Internet or intranets to provide objects (such as Web pages) at the request of any Web browser. The AS/400 HTTP server is the IBM cross-platform Web server.	X	X	X	X
	HTTP Proxy support allows the AS/400 HTTP server to act as a proxy for clients.			X	X
WSG	Transforms AS/400 5250 applications to HTML, allowing users to run AS/400 applications from an PC that has a Web browser. AS/400 supports this simplified way to incorporate a point-and-click interface for your end-users. Use Operations Navigator to control WSG's properties.	X	X	X	X
RouteD	Change routing paths on the TCP/IP servers. RouteD is the AS/400 RIP server, which is the only application that uses multicasting. Use Operations Navigator to monitor and manage the RouteD server.	X	X	X	X
RIP	Assists TCP/IP with routing IP data packets. the AS/400 system supports this dynamic routing protocol that reconfigures routing tables when there is a change in the network.	X	X	X	X
	RIP version 2 with CIDR and VLSM.		X	X	X
TFTP	Transfers files. After the client asks for and receives an IP address from the BOOTP server, the client initiates a TFTP request to the TFTP server for the file. AS/400 supports TFTP file transfers by leveraging the BOOTP server. Use Operations Navigator to start, stop, configure, and manage TFTP.	X	X	X	X

Function	Short description	V4 R1	V4 R2	V4 R3	V4 R4
BOOTP	Provides a means to notify a host of its assigned IP address, the IP address of a boot server host, and the name of a file to be loaded into memory and executed. Works in conjunction with TFTP. AS/400 supports both BOOTP clients and server. Use Operations Navigator to configure the local subnet mask, the local time offset, the addresses of default routers, and the addresses of various Internet servers.	X	X	X	X
DHCP **DHCP Relay**	Automatically assigns IP addresses. The AS/400 system supports this work-saving protocol that dynamically provides client and server configuration information. Operations Navigator includes a complete interface for DHCP configuration and administration.		X	X	X
REXEC	Issues AS/400 commands from other systems across TCP/IP networks. The AS/400 system supports the REXEC client and server (called the REXEC daemon). Use Operations Navigator to start and stop the REXEC server.	X	X	X	X
SNMP	Provides a means of managing an Internet environment over UDP. The AS/400 system supports SNMP. Use Operations Navigator to manage SNMP.	X	X	X	X
NETSTAT	Allows a system administrator to monitor and control the network status of an AS/400 system running TCP/IP or APPC over TCP/IP applications.	X	X	X	X
API	Combine networking application interfaces to communicate with each other. The AS/400 system supports API by allowing programmable multicast applications.	X	X	X	X
	IP multicast		X	X	X
ICMP	Allows for the generation of error messages, test packets, and information messages related to IP.	X	X	X	X
SOCKS	Acts as a gatekeeper for firewall systems. AS/400 TCP/IP includes SOCKS client support to provide the AS/400 system communications through a firewall running SOCKS server. Use Operations Navigator to configure SOCKS.		X	X	X

Function	Short description	V4 R1	V4 R2	V4 R3	V4 R4
SSL	Provides secure transmission of information over TCP/IP. It is an integral part of securing an internal network. The AS/400 system supports SSL on its HTTP server and LDAP server. Use the AS/400 Task page of the *Admin instance of the HTTP server to set up and manage SSL.	X	X	X	X
	SSL enabled TELNET proxy.		X	X	
	SSL for Client Access and TELNET				X
DDM and DRDA	Access data distributed across multiple machines. DRDA is part of the DDM architecture. The AS/400 system supports it by including DDM as part of OS/400. SQL, an IBM database programming language, also supports DRDA implementations.				
	DRDA Level 1 client and server.		X	X	X
	DDM server.		X	X	X
	DDM client.				X

Other applications, such as Domino, Net.Commerce, Payment Server, and user-written applications, can use TCP/IP.

Chapter 2. TCP/IP basic installation and configuration

This chapter covers the installation and configuration of TCP/IP on the AS/400 system. It includes the installation of the code from the shipped media.

2.1 Installing TCP/IP core applications on the AS/400 system

Installing Transmission Control Protocol/Internet Protocol (TCP/IP) on your AS/400 allows you to connect an AS/400 to a network.

To install TCP/IP on your AS/400 system, follow these steps:

1. Insert your installation media for TCP/IP into your AS/400 system. If your installation media is on CD-ROM, insert it into your optical device. If your installation media is on tape, insert it into your tape drive.
2. Type `Go LICPGM` on a command line. Press Enter to access the Work with Licensed Programs display (Figure 1). Type option `11` (Install licensed programs). Press Enter on the Work with Licensed Programs menu to see a list of licensed programs and optional parts of licensed programs (Figure 2 on page 8).

```
LICPGM                  Work with Licensed Programs
                                                        System:   AS21
Select one of the following:

Manual Install
      1. Install all

   Preparation
      5. Prepare for install

   Licensed Programs
     10. Display installed licensed programs
     11. Install licensed programs
     12. Delete licensed programs
     13. Save licensed programs


                                                                  More...
Selection or command
===> 11

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=Information Assistant
F16=AS/400 Main menu
```

Figure 1. LICPGM menu

3. Type `1` in the option column (Figure 2) next to the Licensed Programs you want to install from the Install licensed programs Menu. Press Enter.

```
                        Install Licensed Programs
                                                              System:   AS21
Type options, press Enter.
  1=Install

       Licensed  Installed
Option Program   Status       Description
       5716SVM                SystemView Base for AS/400
       5716SVM                SystemView Base for AS/400 - Launch window
       5769SV3                ADSTAR Distributed Storage Manager for AS/400
  1    5769TC1                TCP/IP Connectivity Utilities for AS/400
       5716VG1                VisualGen Host Services for AS/400
       5769VI1                ImagePlus VisualInfo for AS/400
       5769VI1                ImagePlus VisualInfo Library & Object Server
       5769VI1                ImagePlus VisualInfo Object Server
       5769WP1                OfficeVision for AS/400
       5769WP1                OfficeVision - Text Search
       5769WP1                OfficeVision - Calendar
       5769WP1                OfficeVision - Mail
       5769WP1                OfficeVision - Editor
                                                                      More...
F3=Exit   F11=Display release   F12=Cancel   F19=Display trademarks
```

Figure 2. Installed License Programs

Select TCP/IP Connectivity Utilities for AS/400 (5769-TC1). The AS/400 TCP/IP functions are divided between the operating system and the TCP/IP Connectivity Utilities for AS/400 Licensed Programs. However, the TCP/IP Utilities Licensed Program is shipped with OS/400 at no additional charge and it must be installed separately.

Other Licensed programs that you may want to install include:

- **5769-XD1** (V3R1M3 (or later) Client Access for Windows 95/NT: Provides Operations Navigator support that is used to configure some of the TCP/IP components.
- **5769-SS1 option 3** (OS/400 - Extended Base Support): Provides directory support for functions such as DHCP.
- **5769-SS1 option 31** (OS/400 - Domain Name Server): Maps host names to IP addresses.
- **5769-SS1 option 32** (OS/400 - Directory Services): Provides LDAP support.
- **5769-SS1 option 34** (OS/400 - Digital Certificate Manager): Use if you plan to use Secure Socket Layer (SSL) support.
- **5769-AC1, 5769-AC2, or 5769-AC3** (Cryptographic Access Provider): Provides SSL encryption support to many functions on the system. One of these products should be selected based on your country. These licensed program products (LPP) replace *both* 5769-NC1 and 5769-NCE.
- **5769-DG1** (IBM HTTP Server for AS/400): Provides an AS/400 Web server.

The Confirm Install of Licensed Programs display (Figure 3) shows the licensed program you selected to install. Press Enter to confirm.

```
                  Confirm Install of Licensed Programs
                                                              System:   AS21
Press Enter to confirm your choices for 1=Install.
Press F12 to return to change your choices.

       Licensed  Installed
Option Program   Status      Description
   1   5769TC1               TCP/IP Connectivity Utilities for AS/400









                                                                      Bottom
F11=Display release   F12=Cancel
```

Figure 3. Confirm installation selection

4. Complete the following choices on the Install Options display (Figure 4 on page 10):
   - Installation Device

     Type `OPT01` if installing from a CD-ROM device.

     Type `TAP01` if installing from a tape device.
   - Objects to install

     This option allows you to install both programs and languages objects, only programs, or only language objects. Type `1` for Programs and language objects.
   - Automatic IPL

     This option determines if the system automatically starts when the installation process has completed successfully. Type `N` for “no”. Press Enter.

```
                          Install Options
                                                          System:   AS21
Type choices, press Enter.

  Installation device  . . .   OPT01        Name

  Objects to install . . . .   1            1=Programs and language objects
                                            2=Programs
                                            3=Language objects

  Automatic IPL  . . . . . .   N            Y=Yes
                                            N=No









F3=Exit   F12=Cancel
```

Figure 4. Specify installation options

When TCP/IP successfully installs, either the Work with Licensed Programs menu or the Sign on display appears.

An Installed Status of *COMPATIBLE on the Installed Licensed Programs menu (Figure 5) shows the product is installed.

Note

After the installation of software, the most current PTF package should be installed.

```
                        Install Licensed Programs
                                                              System:   AS21
Type options, press Enter.
  1=Install

       Licensed  Installed
Option Program   Status       Description
       5716SVM                SystemView Base for AS/400
       5716SVM                SystemView Base for AS/400 - Launch window
       5769SV3                ADSTAR Distributed Storage Manager for AS/400
       5769TC1   *COMPATIBLE  TCP/IP Connectivity Utilities for AS/400
       5716VG1                VisualGen Host Services for AS/400
       5769VI1                ImagePlus VisualInfo for AS/400
       5769VI1                ImagePlus VisualInfo Library & Object Server
       5769VI1                ImagePlus VisualInfo Object Server
       5769WP1                OfficeVision for AS/400
       5769WP1                OfficeVision - Text Search
       5769WP1                OfficeVision - Calendar
       5769WP1                OfficeVision - Mail
       5769WP1                OfficeVision - Editor
                                                                      More...
F3=Exit   F11=Display release   F12=Cancel   F19=Display trademarks
```

Figure 5. Install Licensed Programs: TCP/IP connectivity utilities

5. Enter option `50` (Display log for messages) on the Work with Licensed Programs menu (Figure 1 on page 7) to verify that you have installed the licensed program successfully. If an error occurs, you see the message `Work with licensed program function not complete` on the bottom of the Work with Licensed Programs menu.

After the latest PTFs are installed, you are ready to configure TCP/IP on your AS/400 system.

2.2 Before you configure TCP/IP

Before you configure TCP/IP on a system, you must determine the values of the network information needed for TCP/IP. There are six values that may be used during the configuration of TCP/IP. These values should be given to you by the network administrator. If you are the network administrator, and you are setting up TCP/IP in your environment for the first time, you need to determine these values before you continue. Use Table 2 to record the values.

Table 2. TCP/IP configuration values

Parameter	Value
IP Address	
Subnet Mask	
Host Name	
Domain Name	
Domain Name System Server (DNS Server)	
Next Hop Gateway (Router)	

If you have multiple interfaces, you will have multiple IP addresses and a subnet mask. Each interface that is used by TCP/IP must have a unique address.

The required parameters to configure TCP/IP on the AS/400 system are IP address and subnet mask. Some functions will require additional parameters. One example is SMTP, which requires a host name and domain name.

2.2.1 IP address

If your system is going to be accessed by the public, it may require a publicly registered IP address. These addresses are obtained from your Internet Service Provider (ISP). If you are setting up a private internet (intranet), you or the network administrator assigns the address. There is a pool of addresses set aside for this purpose. These are referred to as non-routeable addresses. There are three ranges of these addresses set aside by the Internet Assigned Numbers Authority (IANA). There is one range for each class of addresses. The address ranges are shown in Table 3.

Table 3. Addresses reserved for private internet (intranet) use

Class of Network	Start of Address Block	End of Address Block
A	10.0.0.0	10.255.255.255
B	172.16.0.0	172.31.255.255
C	192.168.0.0	192.168.255.255

Refer to *Address Allocation for Private Internets* (RFC1918) for more details about Internet recommendations for private addresses.

2.2.2 Subnet mask

The subnet mask is used to define the range of addresses that directly connected to the network segment where this interface is attached. The subnet mask and the IP address of the interface are *ANDed* together to determine the network address of the network segment. For example, an interface is defined with an address of 10.1.1.1 and a subnet mask of 255.255.255.0. This means that the network address of the segment is 10.1.1.1, and that systems with an address between 10.1.1.1 and 10.1.1.254 will be found on this segment. Generally speaking, all interfaces on a segment of the network will have the same subnet mask. There are times such as when transparent subnetting is used where this may not be the case.

2.2.3 Host name and domain name

For TCP/IP, the default complete name of the system is defined by the host name and domain name parameters. The host name is the portion that identifies this system by name. Some applications allow you to specify other names in addition to the default host name. The domain name is the name of the domain of the network in which this system exists. If your company does not have a registered domain name, you should get a registered domain name even if you are not planning to have an Internet presence. The domain name is used for addressing e-mail as well as Web serving. To register your domain name, contact an Internet Service Provider (ISP). Many ISPs offer domain name hosting. This is a service that you can use to reserve your desired domain name so that, in the future, you will not have to change your configurations.

2.2.4 Domain Name System (DNS) server

A DNS server is used for name-to-address translation and address-to-name translation. A DNS server can also contain other such information as the mail server name for a domain. If you do not have a DNS server in your domain, we recommend that you configure one on the AS/400 system. The minimum configuration of a DNS will contain the name and address of the AS/400 server. The configuration should contain the names and addresses of all the other servers in the network. The configuration may not need to contain the names of other systems in the network that are clients. Some applications expect the DNS to know all clients and servers in the network. One example of this is a Line Printer Daemon (LPD) server that uses the IP address to create a banner page.

2.2.5 Next hop gateway

The next hop gateway or router is how this system gets to other systems in other parts of the network. If your network consist of a single segment, you will not have any entries for next hop gateway. If you have multiple routers on this segment, you may have multiple next hop gateway entries. The route selection is based on the most specific match. First, it checks for is direct route, or rather an interface that is connected to the destination segment. Next, it selects specific routes. The final selection is default routes. If no route is found, the data will not flow and the connect will fail.

If you have multiple interfaces on your AS/400 system and you have a router in your network, you should have one default route defined for each interface. Each route entry should have `Preferred binding interface` specified as a different interface. Define these routes with the same Duplicate route priority of six or higher. Then, the AS/400 system can balance the connections across all the interfaces. If your add route entry command does not contain these parameters, the function is not supported in your release of TCP/IP.

To learn more about these parameters and how they are used in TCP/IP, refer to the book *TCP/IP Tutorial and Technical Overview,* GG24-3376.

2.3 Configuring TCP/IP using Operations Navigator

You can configure and work with TCP/IP via Operations Navigator or the Command Line interface. For some functions, the entire configuration must be done using Operations Navigator, while other functions can only be configured using the command line interface. You may need to create a TCP/IP interface and start TCP/IP using the AS/400 command line interface before you can access Operations Navigator the first time.

This section covers how to perform the following tasks for TCP/IP using Operations Navigator (the graphical user interface):

- Accessing Basic TCP/IP configuration
- Configuring a TCP/IP interface
- Configuring the Domain and Host Name for TCP/IP
- Configuring Host Table Entries for TCP/IP
- Configuring a TCP/IP Route
- Starting and Stopping TCP/IP
- Verifying TCP/IP connection (PING)

2.3.1 Accessing the TCP/IP configuration

AS/400 Operations Navigator is a powerful graphical interface for Windows 9x/NT clients. To use Operations Navigator, you must have Client Access installed on your Windows 9x/NT PC and have a connection to the AS/400 system you want to configure.

TCP/IP allows you to connect an AS/400 to a network. To get to the point where you can configure TCP/IP for your AS/400 using Operations Navigator, perform the following steps:

1. Start Operations Navigator by clicking **Start->Programs->IBM AS400 Client Access->AS/400 Operations Navigator**. The AS/400 Operations Navigator window appears.
2. Double-click the AS/400 Systems icon (A). It gives you a list of all the AS/400 systems that you can access.
3. Double-click the AS/400 system (AS1) (B) that you want to configure.
4. Double-click **Networks** (C).
5. Double-click **Protocols** (D).
6. Right-click **TCP/IP** (E) to see the context menu (Figure 6).

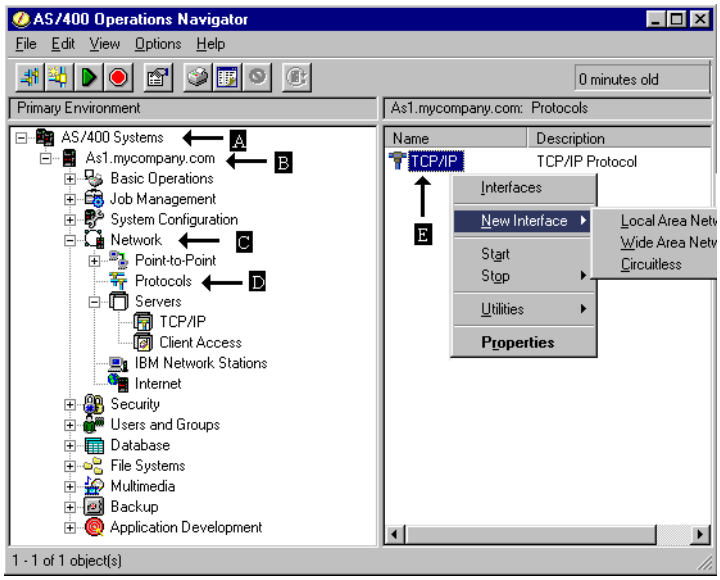


Figure 6. Operations Navigator protocols

You are now ready to start your configuration process.

2.3.2 Configuring a line for TCP/IP

The communication objects for AS/400 TCP/IP are the line descriptions, controller descriptions, and device descriptions. Operations Navigator lets you configure a line for an Ethernet or a Token-Ring network adapter. When TCP/IP starts the line, the controller and device description are automatically varied on. If the controller and device descriptions don't exist, TCP/IP automatically creates them.

The procedures for creating a line and adding TCP/IP support to it and adding TCP/IP support to an existing line are very similar. In this section, we show a combination of both procedures with notes to point out where the differences occur.

The configuration wizard takes you through the steps that are needed to configure a line for TCP/IP for the AS/400 system. To use the configuration wizard, perform the following steps:

1. Access the TCP/IP context menu using the steps in 2.3.1, “Accessing the TCP/IP configuration” on page 14.
2. Select **New Interface** from the context menu (Figure 6). Depending on your version and release, you may see the next selection menu Local Area Network, Wide Area Network, Circuitless. Click **Local Area Network**. You should now see the first window of the TCP/IP wizard interface. Click **Next**. The New TCP/IP Interface Type window appears (Figure 7).

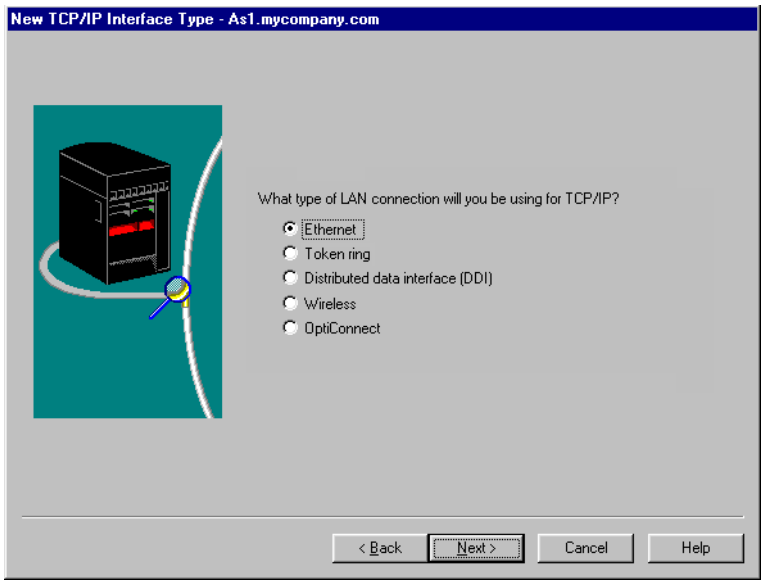


Figure 7. Select interface type

3. Select the type of connection you will define for TCP/IP (**Ethernet** or **Token-ring**). In our example, we selected Ethernet. If you select Token-Ring, you will see different parameters to define. Click **Next**. The New TCP/IP Interface Resource window appears (Figure 8 on page 16).

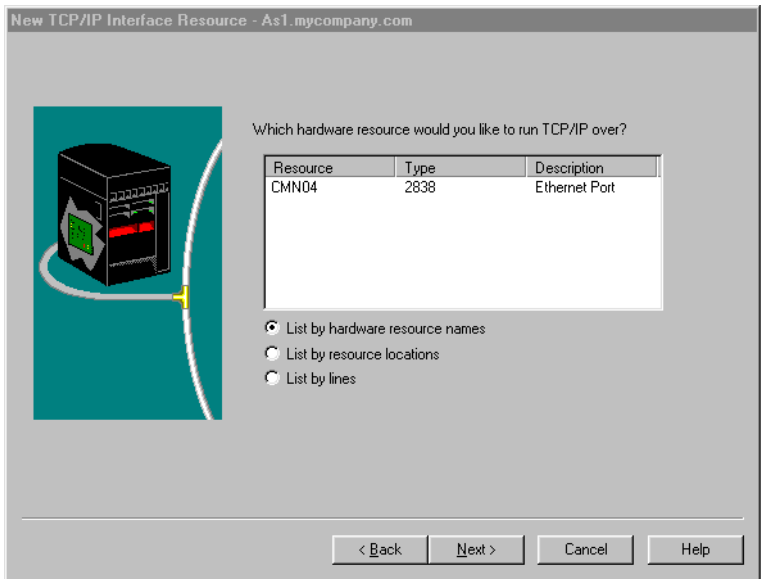


Figure 8. Select hardware resource

4. The New TCP/IP Interface Resource window shows all the hardware on your system that matches your type selection. In our example, we have one Ethernet adapter (CMN04). You should use the buttons on the window to determine the location of the adapter. You can also use the buttons to list communication lines that are currently defined. Right-click on the hardware resource you want to configure. Click **Next**.

   The Choosing a Line window appears (Figure 9) if a line is already defined for the hardware resource you selected. Go to step 5. The Creating a New Line Description window appears (Figure 10) if there are not any lines defined using the selected resource. Go to step 6.

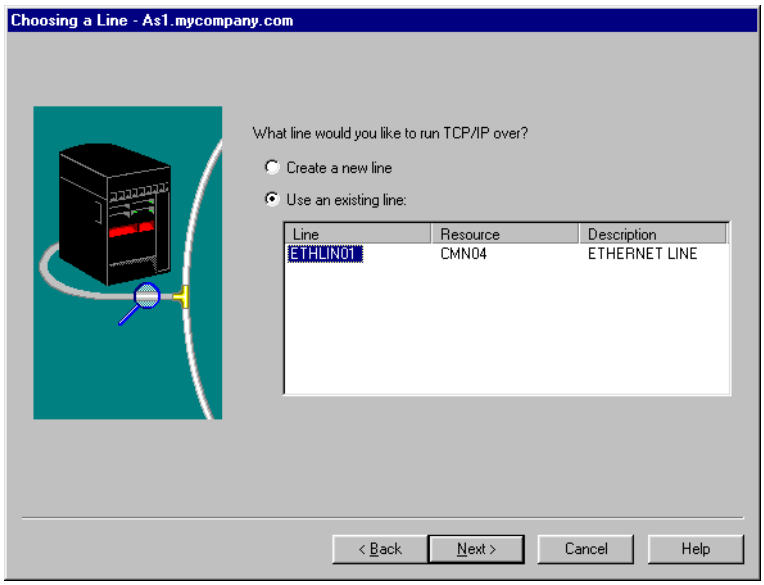


Figure 9. Choosing a Line

5. To configure a TCP/IP interface on an existing line, click **Use an existing line**, and select the line to use from the list provided. Click **Next**. Then, skip to step 8. To create a new line, click **Create a new line**, and click **Next**. Then, continue with step 6.

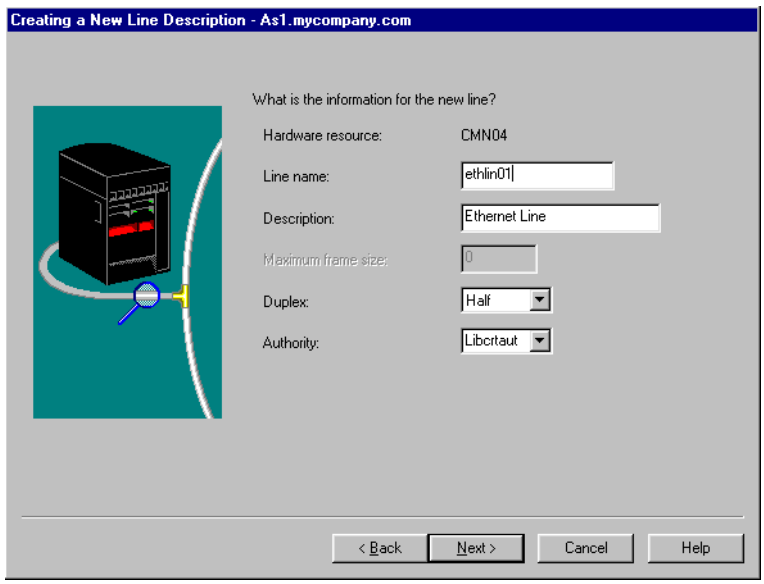


Figure 10. Creating a New Line Description

6. Enter a **Name** and a **Description** for the new line. Select the appropriate values for **Duplex** and **Authority** based on your environment. The Help button provides additional information to assist you in determining your correct values. The Duplex value is based on the type of network hardware you are using to construct your physical LAN. Click **Next**. The Ethernet Line Characteristics window displays (Figure 11).

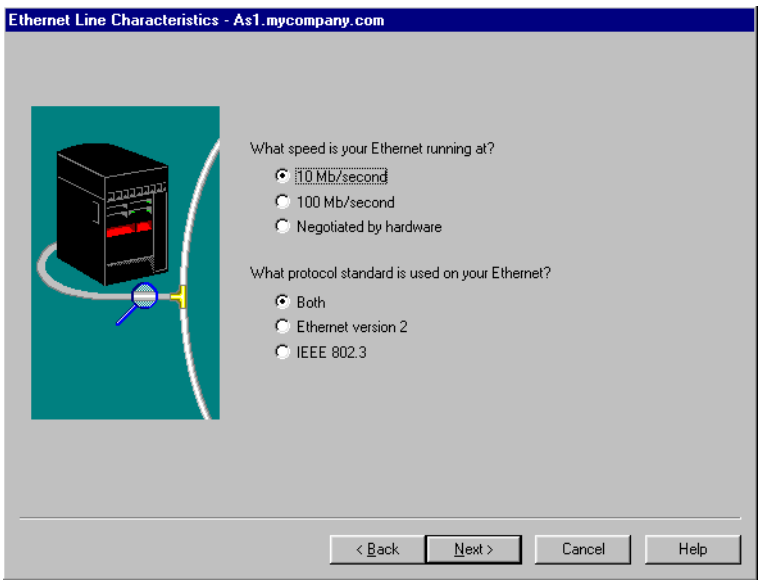


Figure 11. Ethernet Line Characteristics

7. Select the **speed** at which your LAN is running. Select the protocol standards that you want to support on this adapter. Click **Next**. The TCP/IP Interface Settings window displays (Figure 12).

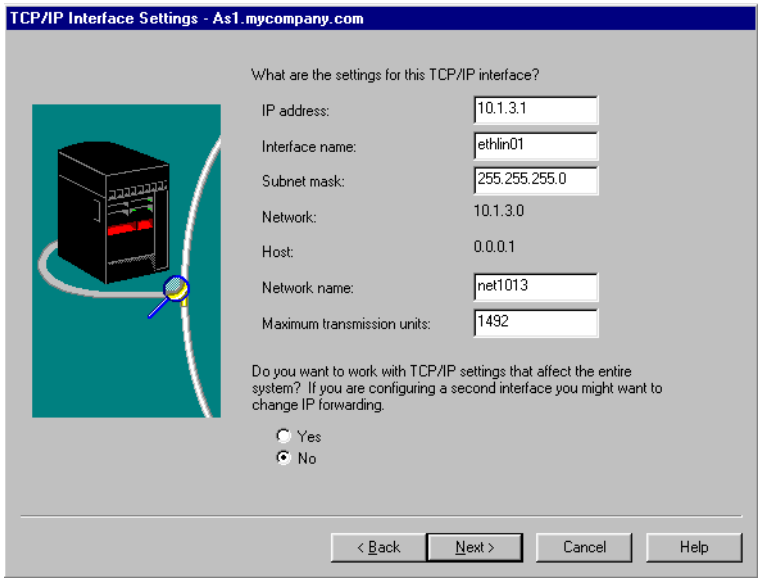


Figure 12. TCP/IP Interface Settings

8. The TCP/IP Interface Settings window (Figure 12) allows you to assign an IP address to your network adapter. Type the IP address, the interface name (we used line name), and the subnet mask for this IP address.

   For the IP address and Subnet mask parameter, specify the value provided by the LAN administrator or Internet Service Provider (ISP). The system takes the IP Address and Subnet Mask and performs a "logical AND" to determine the Network and Host values displayed in the window. The subnet mask and the IP address enable IP protocol to determine where to send the data it receives.

   Network name specifies the name of the network for which you are defining interfaces and routes for the given network address.

   The Maximum Transmission Unit (MTU), specifies the maximum size (in bytes) of the IP datagram that you can send on this interface. The maximum size specified for a particular route should not be larger than the smallest MTU that is supported by any router or gateway in that route. If the MTU size is larger than the smallest MTU in the route, the router with the small MTU will fragment the packet. This can increase the traffic on the segment and lead to performance degradation. The Help Button provides additional information about MTU.

   After you have specified all the values, click **Next**. The TCP/IP Routing window appears (Figure 13).

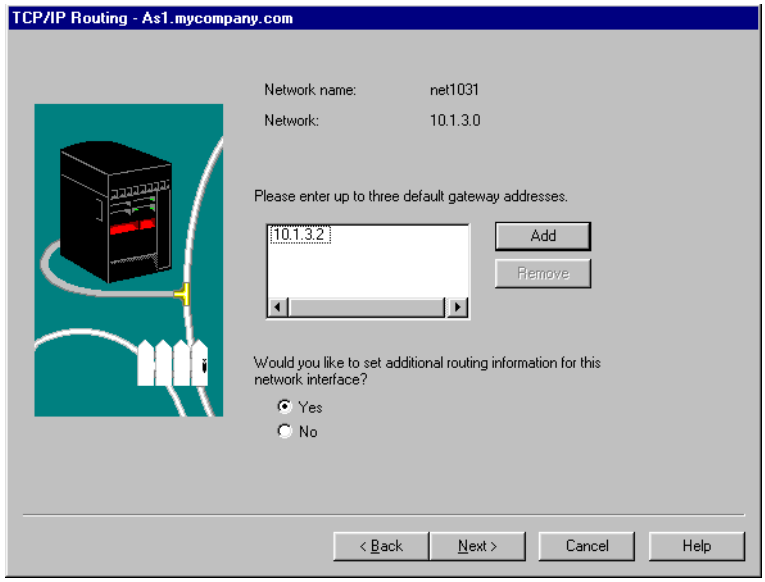


Figure 13. TCP/IP Routing

9. This window is where you list the gateways to which this route directly connects. A gateway is a piece of hardware that connects two are more network segments. It is often called a *router*. You can define up to three gateway addresses. If your AS/400 system is only attached to a single network, you do not need to specify any gateway addresses. This is also where you specify additional routing information for this interface. This may be used for load balancing or to define multiple routes for backup purposes. Click the **Yes** button to configure additional route information. Click **Next**. The TCP/IP Routing window (Figure 14) appears.

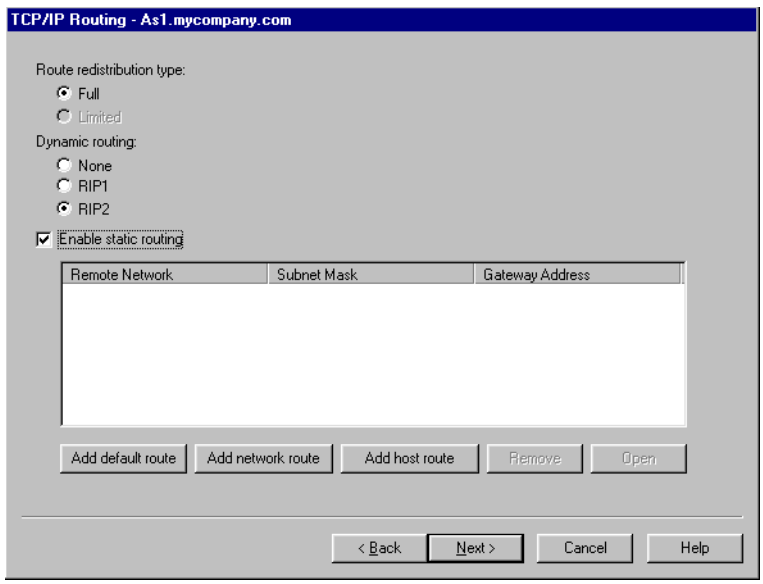


Figure 14. TCP/IP Routing additional information

10. The TCP/IP routing Additional information window allows you to specify whether these routes should be published to the network using RIP1 or RIP2. You can also define default routes, network routes, and routes to a specific

host. Click the appropriate button to add the required routes. In this example, we clicked **Add default route**. The Add Default Route window appears (Figure 15).

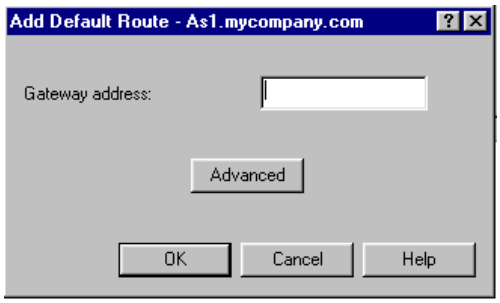


Figure 15. Add Default Route

11. Each of the Add route windows has an **Advanced** button. Specify the gateway address, and click **Advanced**. The Advanced Routing Settings window appears (Figure 16).

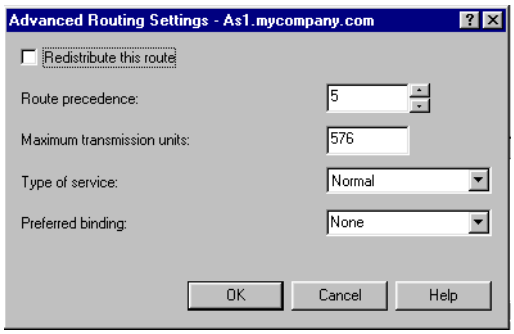


Figure 16. Advanced Routing Settings

12. The Advanced Routing Settings window allows you to specify information about the route. If you leave Route precedence set to 5, the route selection works as usual. If you set Route precedence to a value less than 5, this route will not be a preferred route to the destination network. If Route precedence is set to a value greater than 5, the route will be considered a preferred route to the destination network.

    You may have multiple interfaces defined to the same network, multiple routes defined using the interfaces, and the route precedence of these routes set to the same value greater than 5. In this case, the TCP/IP traffic will be balanced across all the interfaces with routes defined. Refer to Chapter 12, “LDAP on the AS/400 system” on page 449, for more details.

    Set the values that you need, and click **OK**. If you do not need to set any advanced values, click **Cancel**.

    When you have added all the route information you need, click **OK** until you reach the TCP/IP Routing window (Figure 14). Click **Next**. The Servers to be Started window appears (Figure 17).

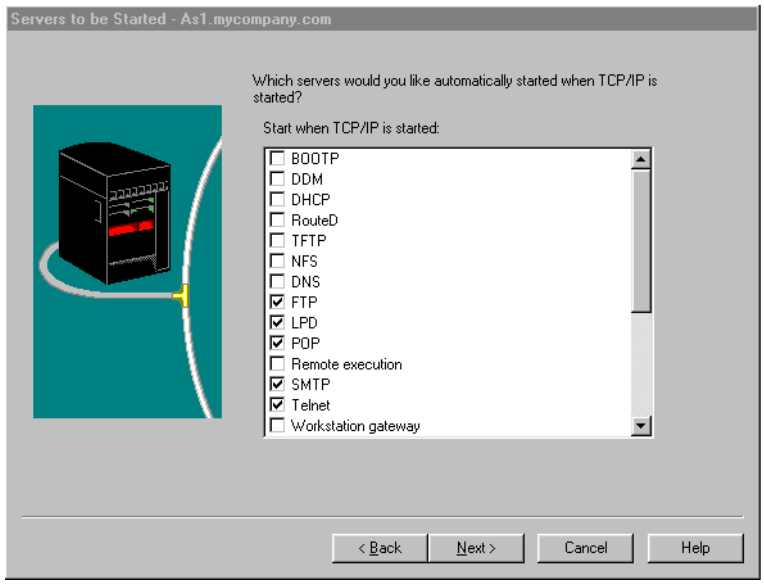


Figure 17. Define servers to start when TCP/IP is started

13. On the Servers to be Started window (Figure 17), select all the currently installed servers that you want to start automatically when TCP/IP starts. If you want to have a particular server automatically started when TCP/IP starts, check the corresponding check box. If you have BOOTP, DHCP, and BOOTP/DHCP servers, only one of them can be checked. After you select all the servers to start, click **Next**. The Start TCP/IP Interface window appears (Figure 18).

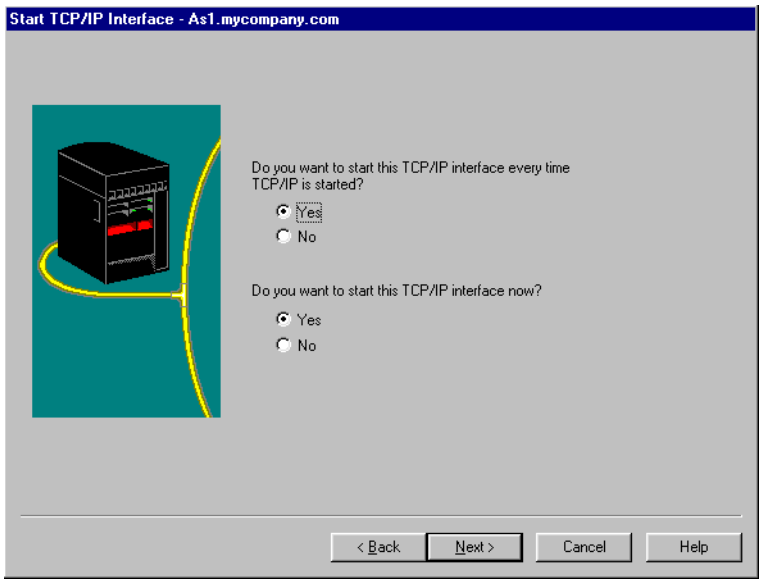


Figure 18. Interface start options

14. On the Start TCP/IP Interface window, identify whether you want this TCP/IP interface started whenever you start TCP/IP and whether you want this TCP/IP interface to start now. If you choose to start the TCP/IP interface here, the interface is tested when you click **Next**. After a successful test, the New TCP/IP Interface Summary window (Figure 19 on page 22) appears.

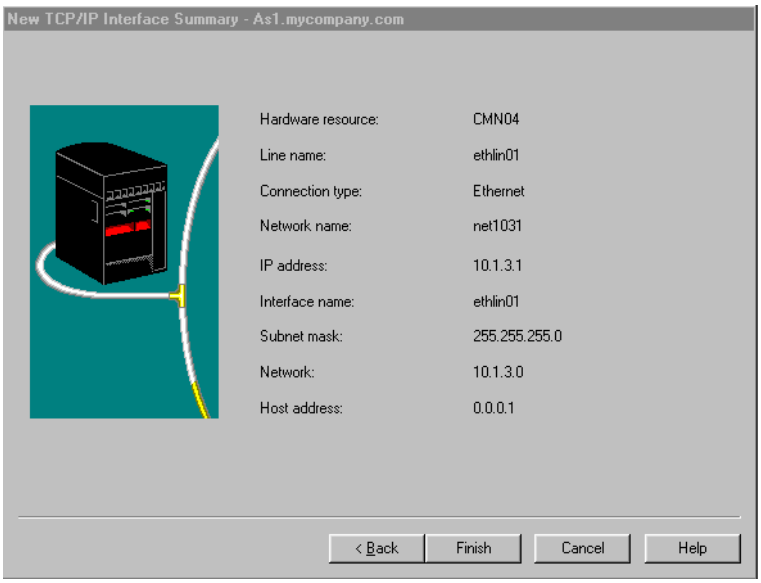


Figure 19. New TCP/IP Interface Summary

15. Verify that all the information displayed is correct. If you need to make changes, click **Back** to go back to the correct window and make your changes. If all the values are correct, click **Finish**.

You have now defined a TCP/IP interface using Operations Navigator.

2.3.3 Changing TCP/IP properties

The TCP/IP attributes of the AS/400 system are accessible from Operations Navigator using the properties selection of the context menu. To use the Operations Navigator, perform the following steps:

1. Access the TCP/IP context menu using the steps in 2.3.1, “Accessing the TCP/IP configuration” on page 14.
2. Select **Properties** (E) from the context menu to make detailed changes to the configuration of your TCP/IP interface. Figure 20 shows the TCP/IP Properties window. Click **Host Domain Information** to specify host domain information for your AS/400 TCP/IP communication. Specify the host name, the domain name, and up to three domain name servers. You can also specify the search order and set advanced TCP/IP settings.
   - **Host Name**: Specifies the name for the AS/400 system. You may not always remember a host by its IP address, but you may find it easier to remember hosts by a name. The host name can be combined with the domain name to make a fully qualified name.
   - **Domain Name**: The domain name is a descriptive label for your organization, such as your_workplace.com. The two parts of the local domain name are the local domain name and the local host
   - **Domain Name servers**: List up to three domain server IP address. The system uses the domain servers in the order that you list them. The domain name servers perform host name resolution by translating the host name into an IP address.

- **Search order**: Specifies whether you want the local host table searched before the domain name server. Figure 20 shows the Host Domain Information dialog.

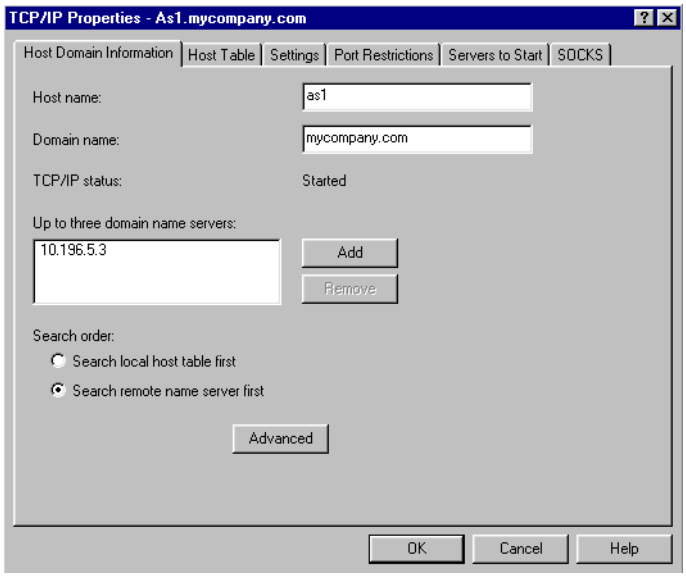


Figure 20. Host domain information

3. Click **Advanced** to set additional DNS values. The Advanced Host Domain Information window (Figure 21) appears. The default values shown work in most environments. If you have intermittent trouble resolving names to IP addresses, you may want to increase the Number of attempts and the interval between attempts. If these values are set too high, you may experience a long wait time before an `Unknown host` message is displayed.

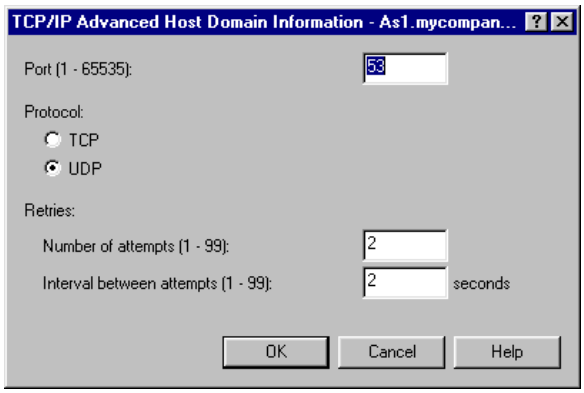


Figure 21. Advanced Host Domain Information settings

4. Click **Host Table** to add and remove host table entries. If you are using the Domain Name System (DNS), you do not necessarily need to add entries here. Figure 22 on page 24 shows the Host Table dialog.

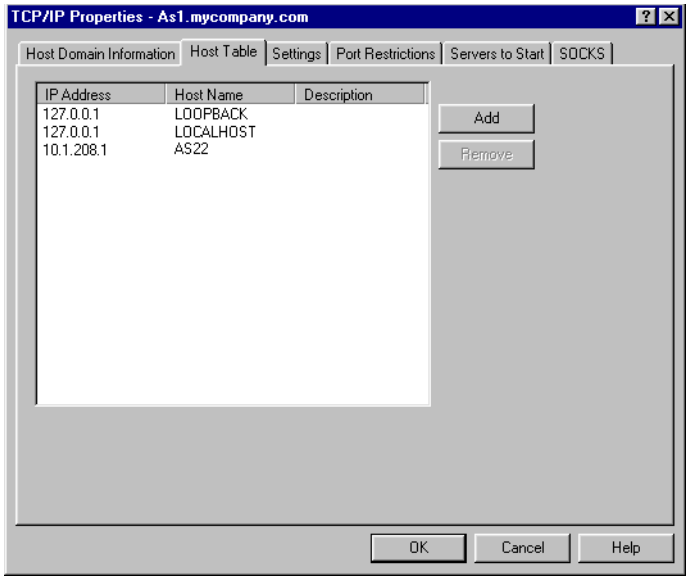


Figure 22. Host Table

5. Click **Settings** to specify IP datagram forwarding, select a TCP urgent pointer convention, and enter a TCP keep-alive time. You can also use the settings page to log protocol errors, enable IP source routing, and enter buffer size, time-out and other values.

   IP forwarding specifies whether you want the IP layer to forward IP datagrams between different networks. This specifies whether the IP layer acts as a gateway (router). It allows the AS/400 system to pass IP datagrams that come in one adapter and go out another adapter.

   The TCP keep-alive specifies the amount of time, in minutes, that TCP waits before sending a probe to the other side of a connection. TCP sends the probe when the connection is otherwise idle, even when there is no data to be sent. Figure 23 shows the Settings dialog.

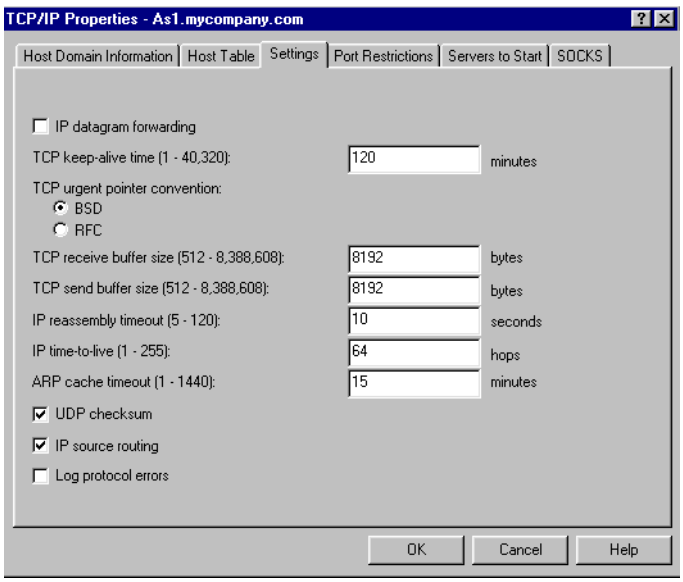


Figure 23. TCP/IP Settings

6. Click **Port Restriction** to limit port use to a user profile name. If you want to restrict a single port, you must specify the same starting and ending port number. Figure 24 shows the Port Restrictions dialog.

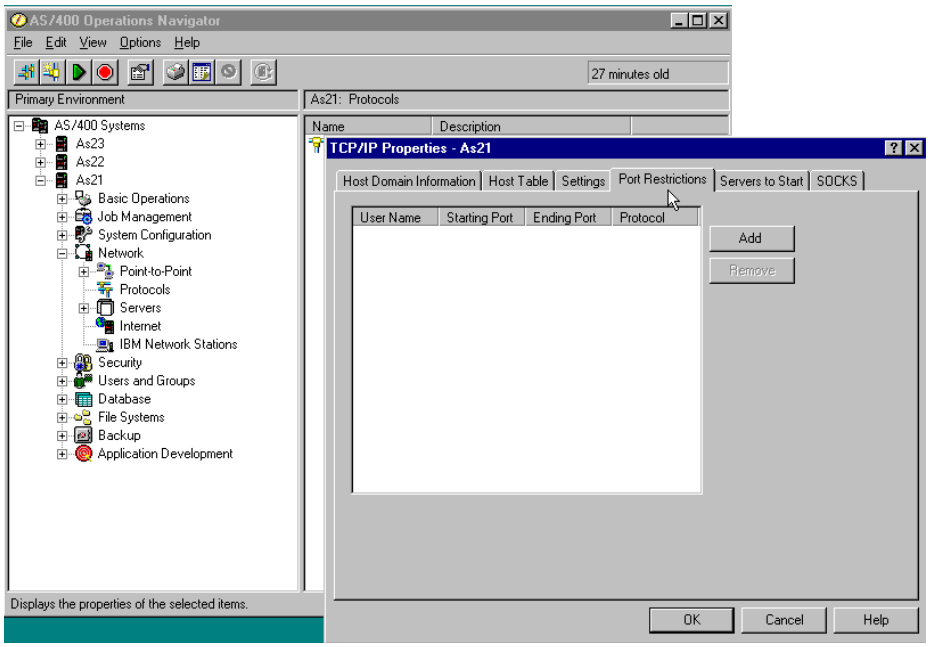


Figure 24. Port Restrictions

7. Click **Servers to Start** to select the currently installed servers that you want to start automatically when TCP/IP starts. Check the servers corresponding check box. If you have BOOTP, DHCP, and BOOTP/DHCP servers, only one of them can be checked. Figure 25 shows the Servers to Start dialog.

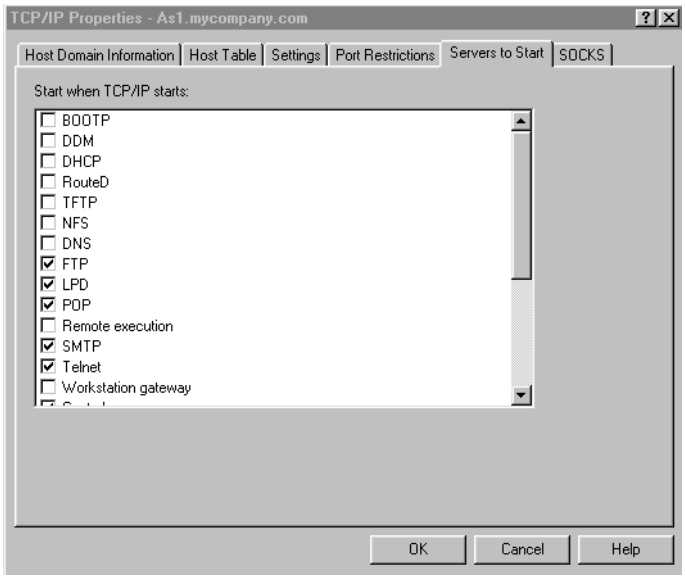


Figure 25. Servers to Start

8. Click **SOCKS** to define the TCP client connection to internal secure networks and to less secure networks. You can define a direct connection to servers in

the internal secure network. Users must have ISOSYS CFG special authority to change information on this dialog. Figure 26 shows the SOCKS dialog.

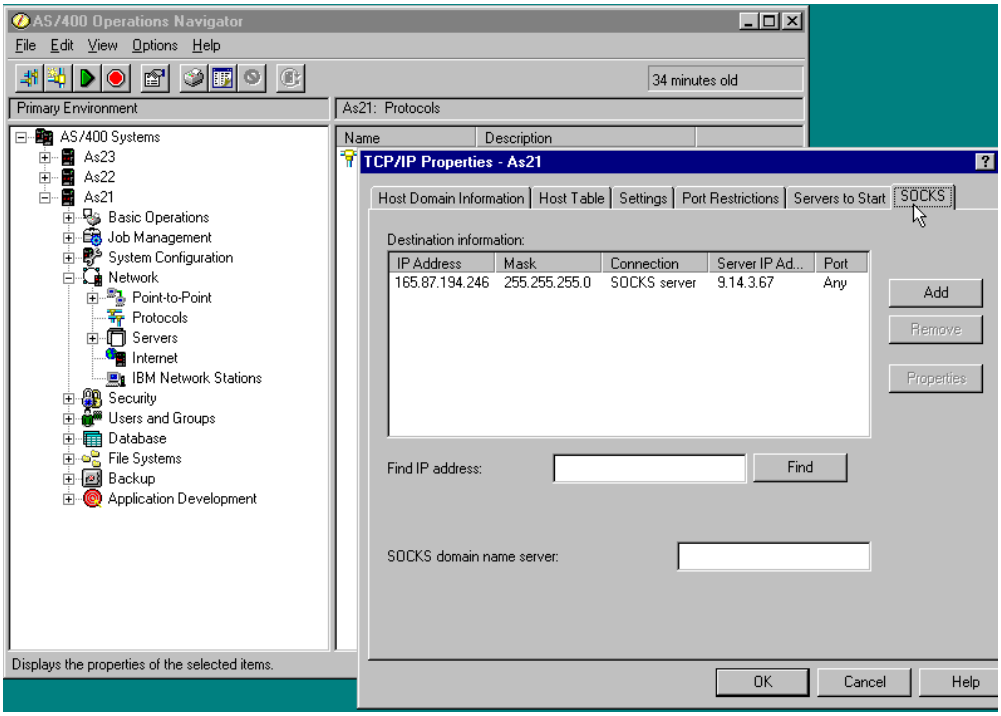


Figure 26. SOCKS dialog

9. After completing changes to the TCP/IP Properties Dialog, click **OK** to save the configuration file and close the window.

2.3.4 Configuring host table entries

You must configure host table entries for TCP/IP if you want the users of your AS/400 system to use easily remembered names rather than IP addresses. If you are using the Domain Name System (DNS), you do not need to configure host table entries.

The host table provides the advantage of not having to remember actual Internet addresses for systems in the network. A host table accomplishes this task by mapping Internet addresses to TCP/IP host names. The local host table on your AS/400 system contains a list of the Internet addresses and related host names for your network.

Before you begin configuring your host table entries for TCP/IP, you need to know the IP addresses of your hosts. You also need the to know the host names and descriptions of the hosts that you want to include in the host table.

To configure host table entries for TCP/IP using Operations Navigator, perform the following steps:

1. Select the appropriate TCP/IP window, as follows:
   a. Start Operations Navigator by clicking **Start->Programs->IBM Client Access->AS/400 Operations Navigator**. The AS/400 Operations Navigator window appears.

b. Double-click the AS/400 Systems icon (A) (Figure 27). A list of all the AS/400 systems should appear that can be configure.

c. Double-click the AS/400 system you want to configure (B).

d. Double-click **Networks** (C).

e. Double-click **Protocols** (D).

f. Right-click **TCP/IP** to open a context menu (E).

2. Select **Properties** from the context menu (F).

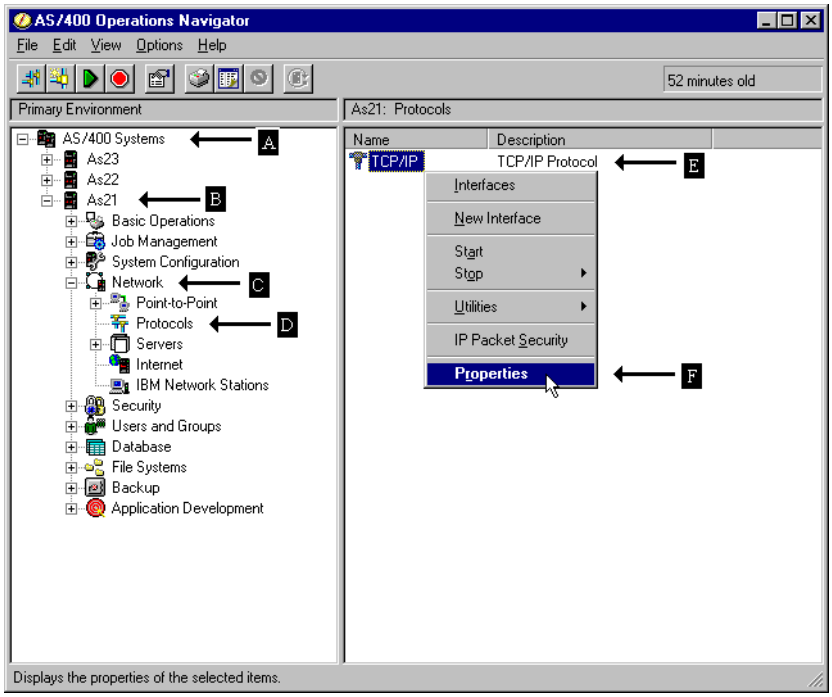


Figure 27. Context menu: Properties

3. Click **Host table** (A) as shown in Figure 28 on page 28.

4. Click **Add** (B) to specify the IP address, hostname, and description of the host that you want to include in the host table. Figure 28 on page 28 shows the TCP/IP Host Table Entry dialog.

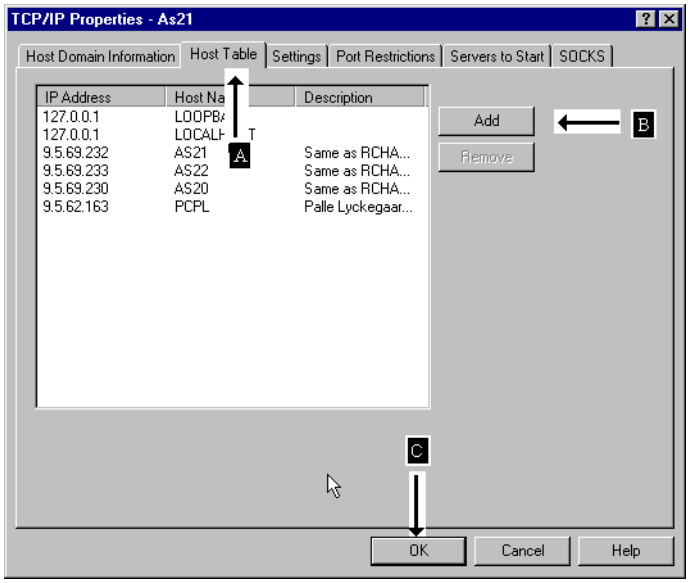


Figure 28. TCP/IP Host Table entry

5. After specifying the IP address, the host name, and the description of the hosts that you want to include in the host table, click **OK** (C) to save the configuration file and close the window.

2.3.5 Configuring the domain and host name

You must configure the local domain and host name if you use a remote name server that requires a full domain to resolved an IP address. The local domain name is information that is provided by:

- The network provider
- The local network administrator

If this is a “true” intranet, the name is created by the customer.

Within TCP/IP, the primary name associated with your system (your system can have more than one name) is called your *local domain* and *host name*. This is important if you later want to set up e-mail, LPR, and ANYNET. They require the local domain and host name. File transfer and Simple Network Management Protocol use these names, but don’t require them.

To configure a local domain and host name for TCP/IP, perform the following steps:

1. Select the appropriate TCP/IP window, as follows:
   a. Start Operations Navigator be clicking **Start->Programs->IBM Client Access->AS/400 operations Navigator**.
   b. Double-click your AS/400 Systems icon (A) (Figure 29). It should give you a list of all the AS/400 systems that you can configure.
   c. Double-click the AS/400 system for which you want to configure a domain and host name (B).
   d. Double-click **Networks** (C).
   e. Double-click **Protocols** (D).

f. Right-click **TCP/IP** to open a context menu (E).

g. Select **Properties** from the context menu (F).

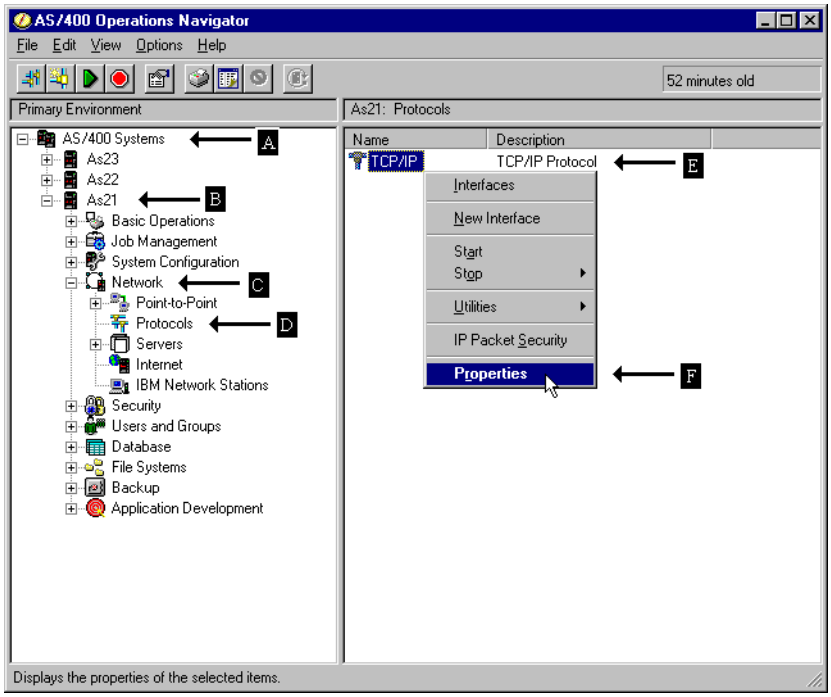


Figure 29. Context menu: Properties

h. Click **Host Domain Information** (A) (Figure 30).

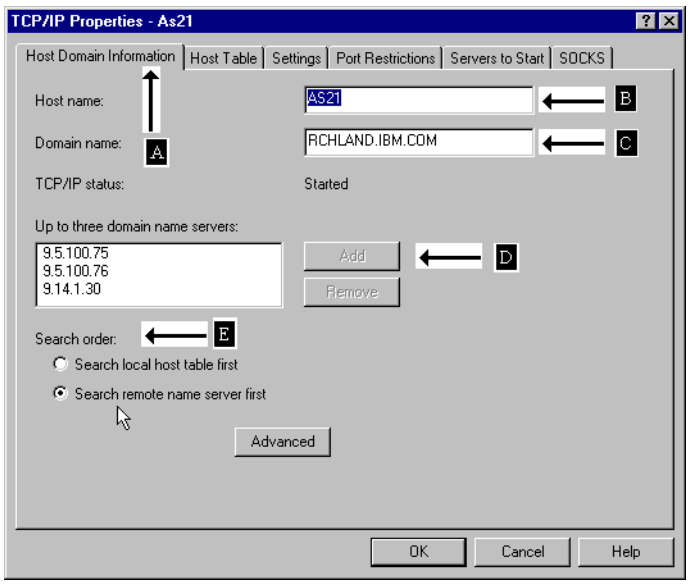


Figure 30. Host Domain Information

2. Specify your host name (B) and domain name (C). You can also select the search order (D), set advanced TCP/IP settings, and specify up to three domain name servers (Figure 30).

3. Click **OK** to save the configuration file.

2.3.6 Configuring a TCP/IP route

A network can consist of many interconnected networks. A route must be defined for your system to communicate with a system on another network. If you want to reach remote networks, you need to configure a TCP/IP route for your AS/400 system.

A TCP/IP interface must be defined before defining a route. A TCP/IP interface implicitly defines a direct route. This is because interfaces define a route to a network to which the AS/400 system is directly connected. Routes added using the AS/400 route commands are called *indirect routes* because they define a route to a network that the AS/400 system is not connected to directly.

The NextHop Internet address for a route definition must exist on a network to which one or more TCP/IP interfaces are connected. The NextHop Internet address usually defines a router or gateway.

Specify the IP address of the router as the default routing entry on the AS/400 system (next hop). This tells the AS/400 system to look for this router if it cannot find a TCP/IP address on its own local network. If you do not configure a TCP/IP route, your AS/400 system cannot reach systems that are on other networks. You may also want to configure a TCP/IP route to give TCP/IP clients access to your AS/400 system.

You do not need to manually configure the routes that tell TCP/IP how to reach the local networks. AS/400 TCP/IP generates these routes automatically from the configuration information for the interfaces every time that TCP/IP starts. Any changes that you make to the routing information take effect immediately.

To configure a TCP/IP route, perform the following steps:

1. Select the appropriate TCP/IP window as follows:
    a. Double-click your AS/400 System icon (A) (Figure 31). It should give you a list of all the AS/400 systems that you are configuring.
    b. Double-click the AS/400 system for which you want to configure a TCP/IP route (B).
    c. Double-click **Network** (C).
    d. Double-click **Protocols** (D).
    e. Right-click **TCP/IP** to open a context menu (E). Select **New Interface** (F).

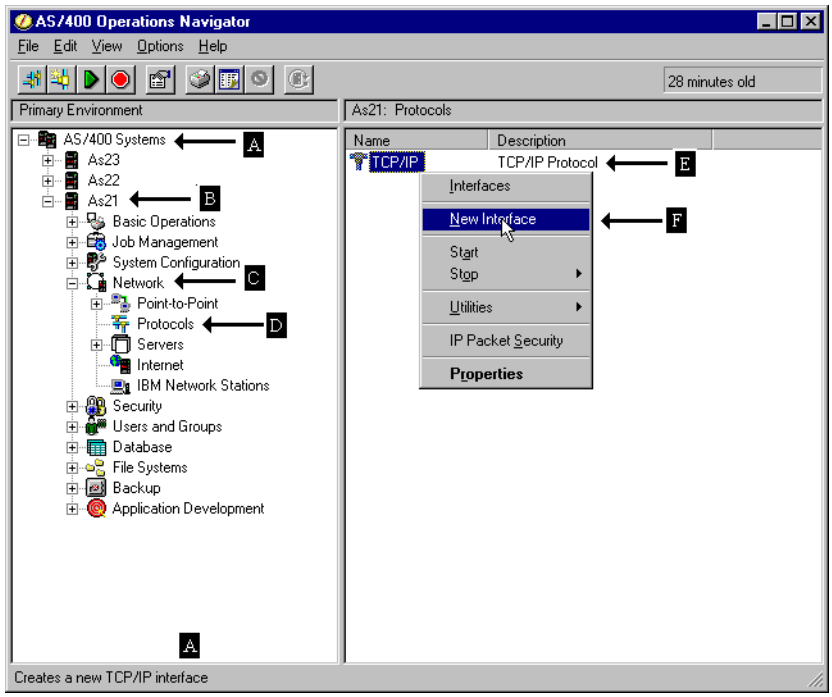


Figure 31. Context menu: New interface

2. Follow the wizard instructions to configure your TCP/IP route. Figure 32 shows the first window of the TCP/IP interface wizard.

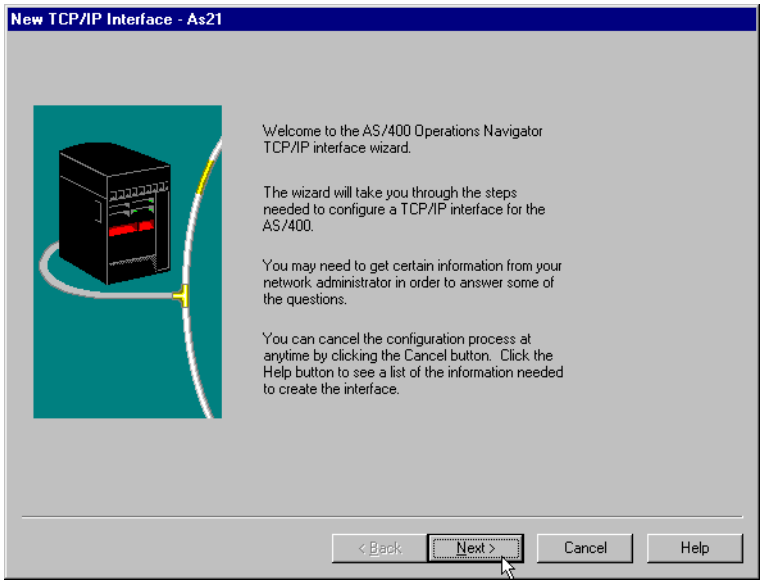


Figure 32. TCP/IP wizard interface

2.3.7 Starting or stopping TCP/IP

Starting TCP/IP initializes and activates the TCP/IP processing, starts the TCP/IP interfaces, and starts the TCP/IP server jobs. TCP/IP must be started before any TCP/IP processing can be performed on the AS/400 system. The starting of TCP/IP only starts the TCP/IP application jobs that have the AUTOSTART

configuration attribute value of *yes. After starting TCP/IP, the QTCPIP job in QSYSWRK subsystem is started. The QTCPIP job is used for activating and deactivating TCP/IP interfaces.

When TCP/IP or ANYNET is already active, use the Start TCP/IP Server (STRTCPSVR) command to start additional TCP/IP application servers.

2.3.7.1 Starting TCP/IP

To start TCP/IP, complete these steps:

1. Select the appropriate TCP/IP window as follows:
   a. Double-click on the AS/400 system icon (A) (Figure 33) in the Operations Navigator tree to give you a list of the AS/400 systems that you are configuring.
   b. Double-click the AS/400 system you want to start TCP/IP processing (B).
   c. Double-click **Network** (C).
   d. Double-click **Protocol** (D).
   e. Right-click **TCP/IP** to open the context menu (E).
2. Select **Start** (F) to initialize and activate TCP/IP processing, start TCP/IP interfaces, and start TCP/IP server jobs.

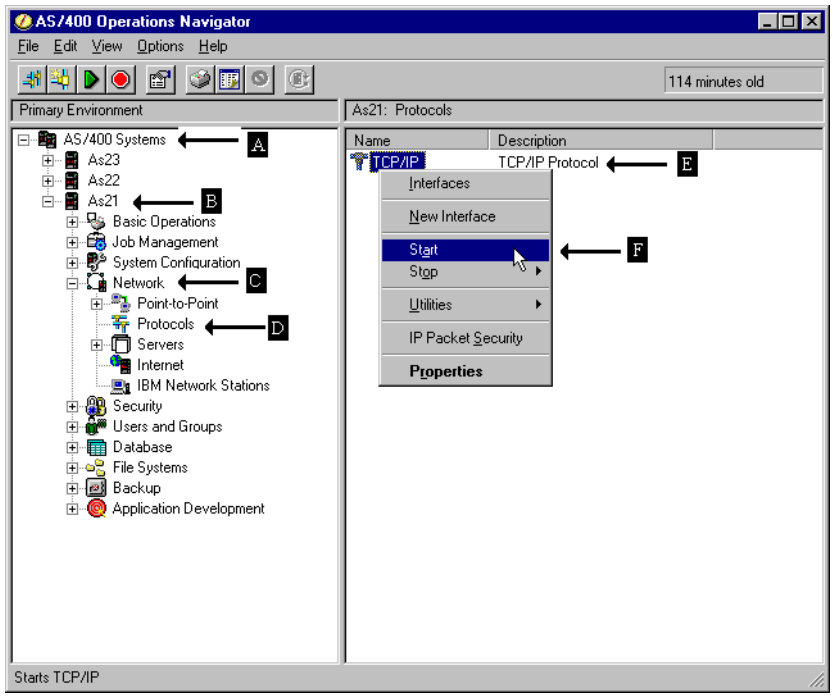


Figure 33. Start TCP/IP

2.3.7.2 Stopping TCP/IP

Stopping TCP/IP ends all TCP/IP processing, all active TCP/IP interfaces, and all TCP/IP connections on the AS/400 system on which you are working. Unless ENDSVR (*NO) is specified, all TCP/IP server jobs for agents that are currently active in QSYSWRK subsystem are ended. There is no confirmation display shown when stopping TCP/IP, so this should be done with caution. There are two

possible values when stopping TCP/IP: Controlled and Immediately. Follow these steps to stop TCP/IP using Operations Navigator:

1. Complete these steps:
   a. Double-click on the AS/400 system icon (A) (Figure 34) in the Operations Navigator tree to give you a list of the AS/400 systems that you are configuring.
   b. Double-click the AS/400 system for which you want to stop TCP/IP processing (B).
   c. Double-click **Network** (C).
   d. Double-click **Protocol** (D).
   e. Right-click **TCP/IP** to open a context menu (E).
2. Select **Stop** (F).
3. Select **Controlled** or **Immediately** (G).

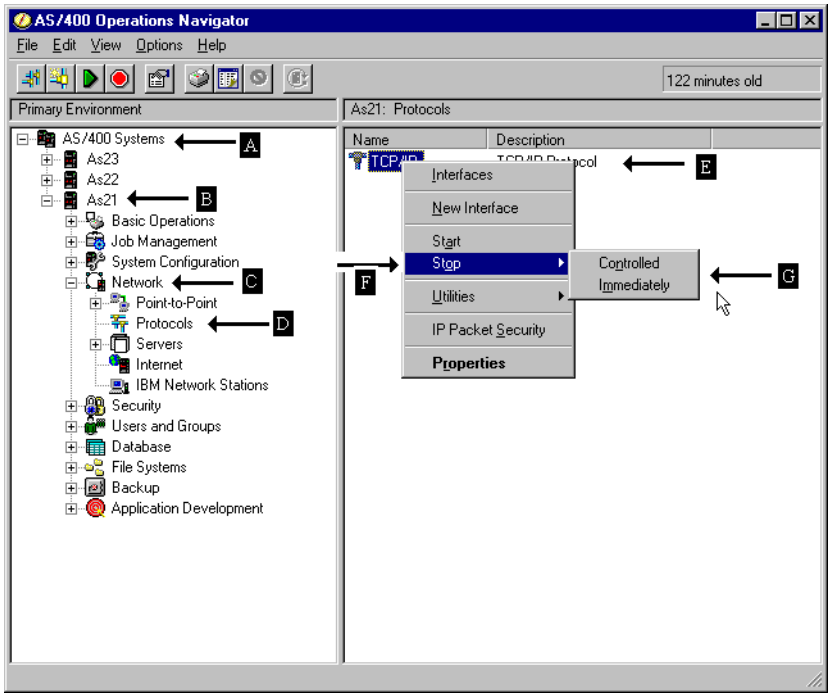


Figure 34. Stop TCP/IP

2.3.8 Verifying a TCP/IP connection (PING)

Verifying a network connection (PING) function is one of the best problem determination tools available for quick diagnosis of a problem in your TCP/IP network. PING tests the TCP/IP connection between a system and the remote system specified on the remote system parameter. It tells you if you can see the host to which you are trying to connect.

When you PING a machine, you send an Internet Control Message Protocol (ICMP) echo request to that machine. A successful reply means that the network's primary transport and communication systems are functioning properly.

To PING a machine using Operations Navigator, complete the following steps:

1. Double-click the AS/400 system icon (A) (Figure 35). It should give you a list of all the AS/400 systems that you can configure.
2. Double-click **Network** (B).
3. Double-click **Protocol** (C).
4. Select **TCP/IP** (D).
5. Right-click **TCP/IP** to open a context menu (E).
6. Select **Utilities** (F).
7. Select **Ping** (G).

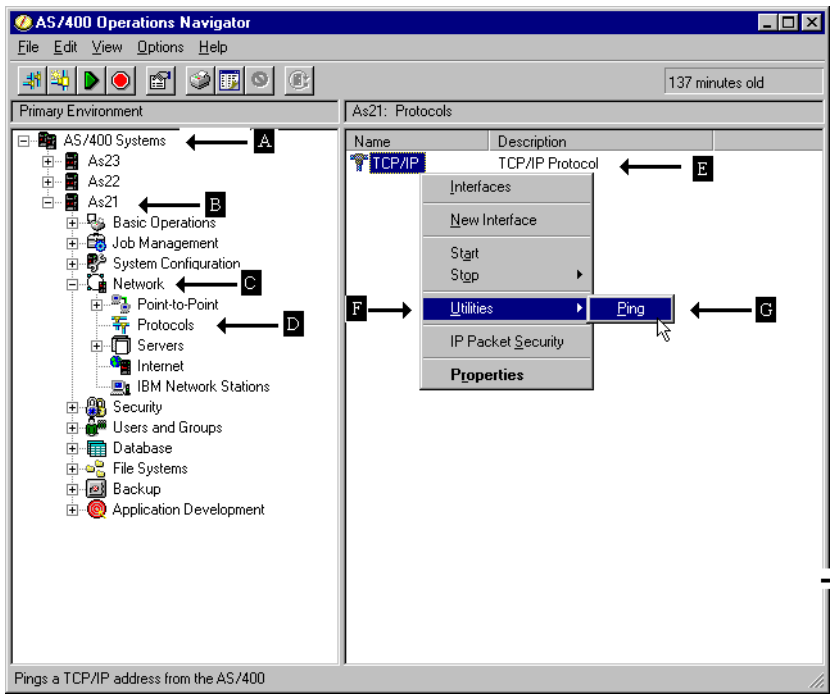


Figure 35. PING dialog

8. Type the IP address or host name of the interface of the host to which you want to test connectivity, and click **Ping Now**. Results of the PING are displayed. Figure 36 shows the PING from dialog window.

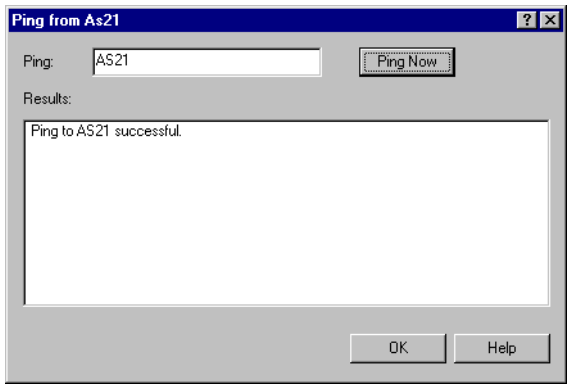


Figure 36. PING from dialog

2.4 Starting and stopping TCP/IP and TCP/IP servers

This section reviews a few of the command line interface commands that are used with TCP/IP. For additional details, refer to *TCP/IP Configuration and Reference,* SC41-5420.

2.4.1 Starting TCP/IP

Before any TCP/IP services are available, TCP/IP processing must be initialized and activated. Starting TCP/IP starts the TCP/IP interfaces and the TCP/IP server jobs.

Only the TCP/IP interfaces and TCP/IP applications with AUTOSTART *YES are started. To start TCP/IP server jobs for applications with AUTOSTART *NO, use the STRTCPSVR command. However, if you enter STRTCPSVR and press the Enter key, all TCP/IP applications are started. Therefore, individually select the servers you want started by entering STRTCPSVR and pressing F4. There are two ways to start TCP/IP on the AS/400 system using CL Interface:

- Enter the STRTCP command, and press F4.
- Type GO TCPADM, and enter option 3 (Start TCP/IP).

To start TCP/IP from the TCP/IP Administration menu, perform the following steps:

1. Type GO TCPADM from the main AS/400 menu. The TCP/IP Administration menu is displayed as shown in Figure 37.

```
TCPADM                      TCP/IP Administration
                                                          System:   AS21
Select one of the following:

     1. Configure TCP/IP
     2. Configure TCP/IP applications
     3. Start TCP/IP
     4. End TCP/IP
     5. Start TCP/IP servers
     6. End TCP/IP servers
     7. Work with TCP/IP network status
     8. Verify TCP/IP connection
     9. Start TCP/IP FTP session
    10. Start TCP/IP TELNET session
    11. Send TCP/IP spooled file

    20. Work with TCP/IP jobs in QSYSWRK subsystem


Selection or command
===> 3

F3=Exit    F4=Prompt    F9=Retrieve    F12=Cancel
```

Figure 37. TCP/IP Administration menu

2. Enter option 3 (Start TCP/IP). Press Enter. The Start TCP/IP display is shown (Figure 38 on page 36).

```
                          Start TCP/IP (STRTCP)

Type choices, press Enter.


                        Additional Parameters

Start application servers  . . .   *YES          *YES, *NO
Start TCP/IP interfaces  . . . .   *YES          *YES, *NO









                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 38. Start TCP/IP servers and interfaces

3. After TCP/IP is started, verify that the QTCPIP is there by entering the following CL command:

   ```
   WRKACTJOB SBS(QSYSWRK) JOB(QT)
   ```

   After this job is started, you can proceed with the TCP/IP connection verification.

2.4.2 Stopping TCP/IP

Stopping TCP/IP ends all TCP/IP processing, all active TCP/IP interfaces, and all TCP/IP connections on the AS/400 system with which you are working. Unless you specified ENDSVR (*NO), all TCP/IP server jobs for agents that are currently active in QSYSWRK subsystem are ended. There are two possible values when stopping TCP/IP: Controlled and Immediately. There are two ways to stop TCP/IP using CL interface:

- Enter the `ENDTCP` command, and press F4.
- Type `GO TCPADM`, and enter option `3` (End TCP/IP).

To stop TCP/IP from the TCP/IP Administration menu, perform the following steps:

1. Type `GO TCPADM` from the AS/400 Main menu. The TCP/IP Administration menu appears (Figure 39).

```
TCPADM                      TCP/IP Administration
                                                              System:   AS21
Select one of the following:

     1. Configure TCP/IP
     2. Configure TCP/IP applications
     3. Start TCP/IP
     4. End TCP/IP
     5. Start TCP/IP servers
     6. End TCP/IP servers
     7. Work with TCP/IP network status
     8. Verify TCP/IP connection
     9. Start TCP/IP FTP session
    10. Start TCP/IP TELNET session
    11. Send TCP/IP spooled file

    20. Work with TCP/IP jobs in QSYSWRK subsystem


Selection or command
===> 4

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
(C) COPYRIGHT IBM CORP. 1980, 1998.
```

Figure 39. TCP/IP Administration

2. Enter option 4 (End TCP/IP), and press Enter. The End TCP/IP display is shown (Figure 40).

```
                              End TCP/IP (ENDTCP)

Type choices, press Enter.

How to end . . . . . .   *IMMED        *IMMED, *CNTRLD

                           Additional Parameters

End application servers  . . . .   *YES          *YES, *NO

                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 40. ENDTCP menu

Note: A confirmation display is not shown when you enter the ENDTCP command. The ENDTCP command must be used with caution. When it is used, it ends all TCP/IP processing on the AS/400 system on which you are working.

3. To individually stop a TCP/IP server, enter the ENDTCPSVR command. Press F4. The End TCP/IP Server display is shown (Figure 41).

```
                        End TCP/IP Server (ENDTCPSVR)

Type choices, press Enter.

Server application . . . . . .   *ALL          *ALL, *SNMP, *ROUTED...
               + for more values

                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 41. ENDTCPSVR

2.4.3 Verifying a TCP/IP connection

The Verify TCP/IP connection (VFYTCPCNN) command, also known as PING, tests the TCP/IP connection between a system and the remote system specified on the remote system parameter. It can be used to test the TCP/IP code and adapter. Verifying a network connection (PING) function is one of the best problem determination tools available for quick diagnosis of a problem in your TCP/IP network.

To test the TCP/IP code, Token-Ring adapter, and Token-Ring connection, specify the Internet address of the local adapter or host name as defined in the host table:

1. Enter: `PING (*INTNETADR) INTNETADR('10.1.3.1')`
2. Enter: `PING RMTSYS(AS21)`
3. Type `GO TCPADM`. Enter option `8` (Verify TCP/IP connection). Enter the IP address of the adapter. The TCP/IP Administration menu is displayed as shown in Figure 42.

```
TCPADM                      TCP/IP Administration
                                                          System:   AS21
Select one of the following:

     1. Configure TCP/IP
     2. Configure TCP/IP applications
     3. Start TCP/IP
     4. End TCP/IP
     5. Start TCP/IP servers
     6. End TCP/IP servers
     7. Work with TCP/IP network status
     8. Verify TCP/IP connection
     9. Start TCP/IP FTP session
    10. Start TCP/IP TELNET session
    11. Send TCP/IP spooled file

    20. Work with TCP/IP jobs in QSYSWRK subsystem


Selection or command
===> 8

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
(C) COPYRIGHT IBM CORP. 1980, 1998.
```

Figure 42. TCP/IP Administration menu

To test TCP/IP code without sending anything out of the token-ring adapter, specify the special host name LOOPBACK, enter:

```
PING LOOPBACK
```

To test the connection to a another system, enter:

```
PING RMTSYS(AS23)
```

If the PING operation is successful, you will see messages similar to those in Figure 43 on page 40.

```
                          Display All Messages
                                                              System:   AS21
Job . . :   QPADEV0005    User . . :   ITSCID62      Number . . . :   003805

3>> Ping '10.1.1.1'
    Verifying connection to host system 10.1.1.1.
    PING reply 1 from 10.1.1.1 took 11 ms. 256 bytes. TTL 64.
    PING reply 2 from 10.1.1.1 took 9 ms. 256 bytes. TTL 64.
    PING reply 3 from 10.1.1.1 took 9 ms. 256 bytes. TTL 64.
    PING reply 4 from 10.1.1.1 took 9 ms. 256 bytes. TTL 64.
    PING reply 5 from 10.1.1.1 took 9 ms. 256 bytes. TTL 64.
    Round-trip (in milliseconds) min/avg/max = 9/9/11
    Connection verification statistics: 5 of 5 successful (100 %).









                                                                      Bottom
Press Enter to continue.

F3=Exit   F5=Refresh   F12=Cancel   F17=Top   F18=Bottom
```

Figure 43. Successful PING message

If the PING operation is unsuccessful, you should see messages similar to those shown in Figure 44.

```
                          Display All Messages
                                                              System:   AS21
Job . . :   QPADEV0005    User . . :   ITSCID62      Number . . . :   003805

3>> Ping '10.1.4.1'
    Verifying connection to host system 10.1.4.1.
    No response from host within 1 seconds for connection verification 1.
    No response from host within 1 seconds for connection verification 2.
    No response from host within 1 seconds for connection verification 3.
    No response from host within 1 seconds for connection verification 4.
    No response from host within 1 seconds for connection verification 5.
    Connection verification statistics: 0 of 5 successful (0 %).









                                                                      Bottom
Press Enter to continue.

F3=Exit   F5=Refresh   F12=Cancel   F17=Top   F18=Bottom
```

Figure 44. Unsuccessful PING messages

If you received unsuccessful PING messages, perform the following steps:

1. Check your configuration steps on the local system.
2. Check the configuration at the remote system.
3. Make sure the remote system is not powered down or has ended TCP/IP.

Chapter 3. SSL security on the AS/400 system

This chapter explains how you can provide security for your Internet application and describes:

- An overview of the elements of transaction security available on the Internet
- A high-level explanation of the Secure Sockets Layer (SSL) protocol
- How to use Digital Certificate Manager on AS/400 to create an intranet certificate authority (CA) and server certificates
- How to configure the IBM HTTP Server for AS/400 to use SSL
- Running a servlet under SSL

3.1 Internet security elements

There is no one single answer to Internet security. Some people think that by installing a firewall between their networks and the Internet, the company's network will be safe.

Is it simply a firewall that shields your company from any inappropriate Internet access? No. Security is a concept, rather than a single device or procedure. It is a set of different security measures that are selected based on the needs of a specific installation. Therefore, it is essential to first discuss the type of Internet security you need to achieve.

First, the policy established by high-level management indicates how your company should deal with the Internet and the level of security that is to be achieved. Various Internet security features, such as cryptography or host system security functions, help you implement what is designed.

Users must be educated to follow and maintain the implemented security procedures and to observe specific rules when acting as Internet clients. These concepts are highlighted in Figure 45 on page 42.

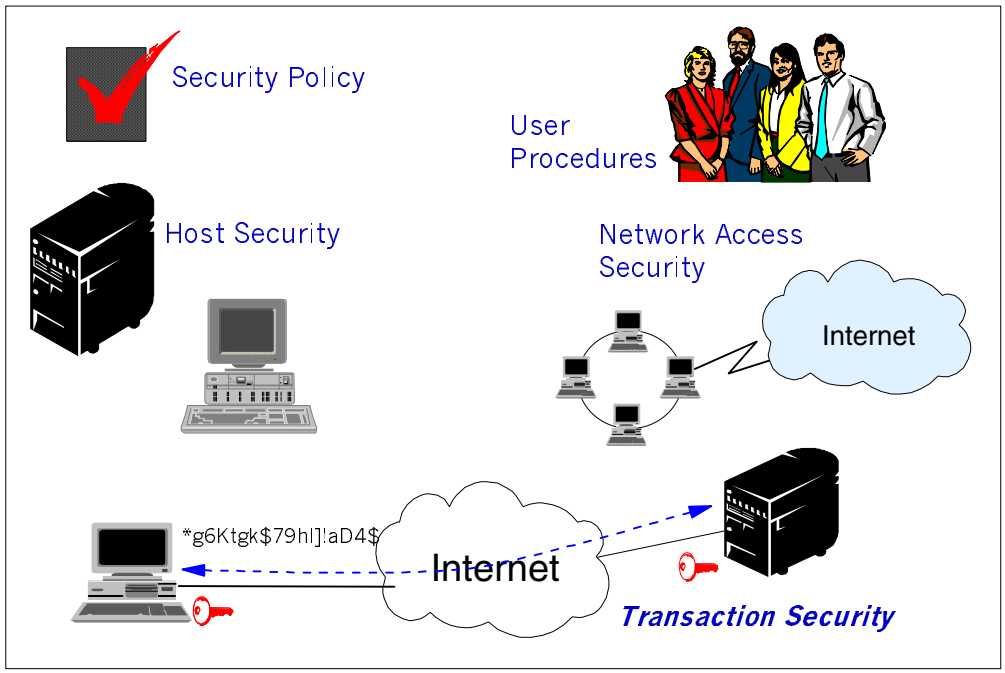


Figure 45. Internet security elements

3.1.1 Transaction security and Secure Sockets Layer

Transaction security includes several basic elements, such as:

- Confidentiality and privacy
- Integrity
- Authentication
- Accountability

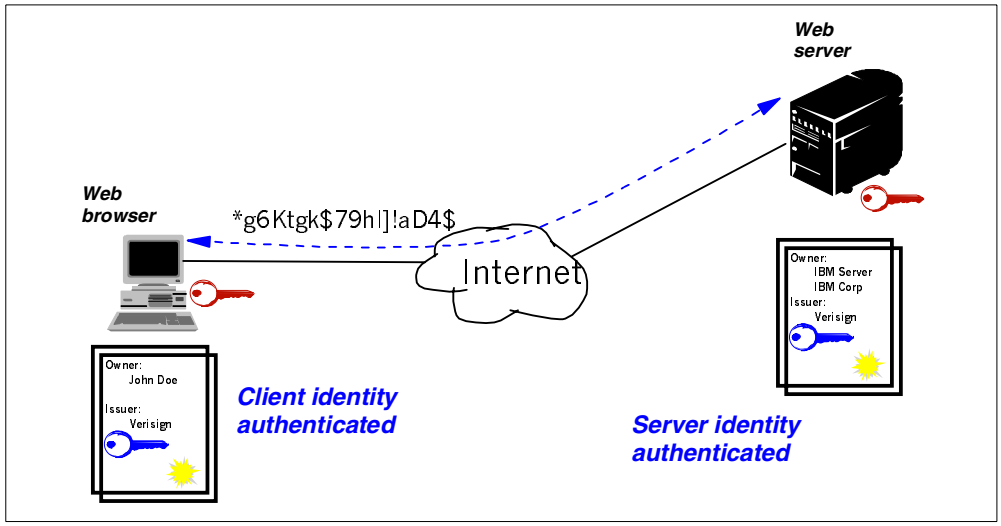


Figure 46. Transaction security

Secure Sockets Layer (SSL) is the protocol is defined by Netscape Communications Corporation. It provides a private channel between a client and

server that ensures privacy of data, authentication of session partners, and message integrity.

Digital certificates are used for session partner authentication. Server authentication is common. Client authentication is not yet common, but it is growing in popularity. Keys are the base for end-to-end information encryption. Figure 46 provides a high-level view of SSL and transaction security.

TCP/IP applications must be rewritten to use SSL. Primarily, SSL is used by HTTP (HTTPS) for Web browsing. In OS/400, the V4R3 Directory Services Server (LDAP) is SSL enabled. Other TCP/IP applications will follow.

3.1.1.1 Confidentiality

Consider this problem: Intruders can eavesdrop on private information as messages travel across the network. The solution lies in encryption. The sender scrambles the message, and the receiver unscrambles it using a secret key.

Confidentiality means that the contents of the messages remain private as they pass through the Internet. Without confidentiality, your computer broadcasts the message to the network, which is similar to shouting the information across a crowded room. *Encryption* ensures confidentiality.

3.1.1.2 Integrity

Consider, for example, that you want to know if the data received is the same as the data that was sent. You can determine this through two possible solutions: *digital signature (hashing)* and *encryption*.

The sending system calculates a value based on the data that is sent. The value is appended to the transmission. The receiving system uses the same calculation to generate a value. The receiving system compares the calculated value with the received value. If the values are different, it assumes that the data changed. Message hashing should be used with encryption for better protection.

Integrity means that the messages are not altered while being transmitted. Any router along the way can insert or delete text or garble the message as it passes by. Without integrity, you have no guarantee that the message you sent matches the message received. Encryption and digital signature ensure integrity.

3.1.1.3 Authenticity

Consider the scenario where you want to know who is at the other end of a Web site to test its authenticity. One way to find out is through the use of digital certificates and digital signatures (Figure 47 on page 44).

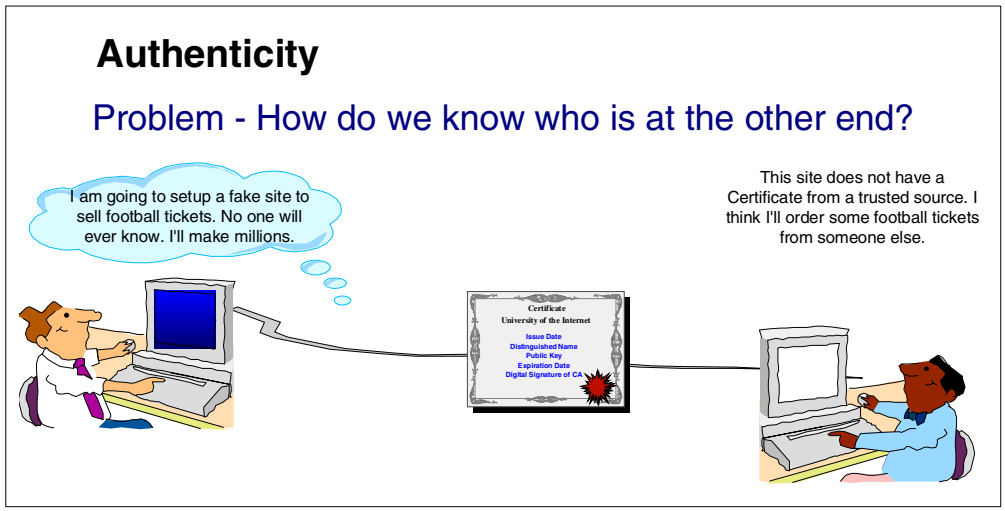


Figure 47. Verifying identity: Digital certificates and digital signatures

Authenticity means that you know who you are talking to and that you trust that person. Without authenticity, you have no way to be sure that anyone is who they say they are. *Authentication* through digital certificates and digital signatures ensures authenticity.

There are two ways in which the server uses authentication:

- Digital signature
- Digital certificates

A digital signature ensures accountability. But how do you know if the person sending you a message is who he says he is?

You look at the sender's *digital certificate*. A public key certificate is issued by a trusted third party known as the *certifying authority* (CA). The browser and server exchange information, including their public key certificate. SSL uses the information to identify and authenticate the sender of the certificate.

A digital certificate is like a credit card with your picture on it and a picture of the bank president with his arm around you. A merchant trusts you more because you look like the picture on the credit card, and they know the bank president trusts you, too.

You base your trust for the authenticity of the sender on whether you trust the third party (a person or agency) that certified the sender. The third party or certification authority (CA) issues digital certificates.

How can you ensure that the person sending the message is really trustworthy? To illustrate this, consider the following example.

If you wake up one day feeling ill, you may decide to visit a doctor. You can select a doctor from your phone book and go to their office for a visit. Once you arrive at the office, how can you be sure that the person about to examine you is really a doctor? After all, you have never met this person before. They may look like a doctor and act like a doctor, but how do you know that this person has successfully completed all the training necessary to become a doctor?

You need certification by a trusted third party to reassure you that this person really is a doctor. The doctor probably has a diploma on the wall stating that they have successfully completed their training. If the diploma is from a well-known school, you may be reassured that you are about to be examined by a real doctor. What if the diploma is from the medical school of a correspondence school whose name you don't recognize? You may not be reassured.

Authentication works the same way. Trusted third parties verify that the server really is who it claims to be. This verification is provided with a digital certificate (the digital equivalent of your doctor's diploma hanging on the wall). You base your trust for the authenticity of the server on whether you trust the third party that certified the server (the school that issued the diploma). That third party is called a *certifying authority* (CA).

The term *trusted root* is given to a trusted certifying authority (CA) on your server. A *trusted root key* is the key belonging to the CA.

Authentication can be used server to client (server authentication) or client to server (client authentication). Server authentication was described earlier. The clients authenticate the servers. With client authentication, the client is authenticated by the server. For example, if a server contains hospital patient information, you may use client authentication to verify that the client attempting to access the data is really who they said they are before allowing them access to patient records.

3.1.1.4 Accountability

Consider the situation where you want to prove that a transaction took place. We combine all the techniques we have seen. First, the data is hashed using cryptography to assure its integrity. The data is encrypted by using the keys derived from the public key exchange, which assures the identity of the session partners. This is used in combination with a time stamp in the data to provide a log of the transactions.

Accountability means that both the sender and receiver agree that the exchange took place. Without accountability, the addressee can easily say that the message never arrived. Digital signatures ensure accountability. Accountability is *not* part of the SSL protocol.

3.1.2 HTTP server over SSL (HTTPS)

SSL ensures that data transferred between a client and a server remains private. It allows the client to authenticate the identity of the server. In addition, SSL V3 allows a server to authenticate a client.

Figure 48 on page 46 shows the high-level view of the flow that takes place when a client (browser) sends an HTTPS request to an HTTP server.

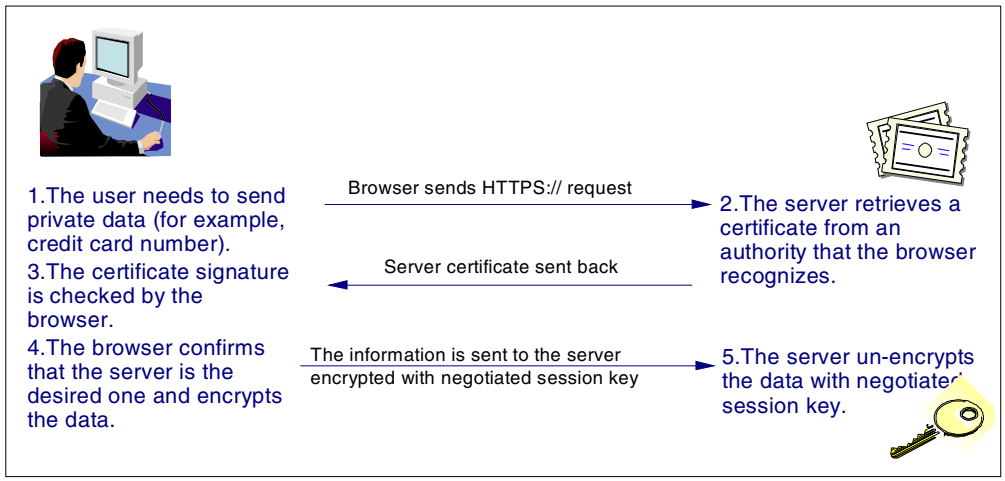


Figure 48. HTTP server using SSL

If SSL client authentication is configured, the server requests the client's certificate for any HTTPS request. The server establishes a secure session depending on whether the client has a valid certificate. This depends on the server configuration, which may specify no client authentication, optional client authentication, and mandatory client authentication.

Once your server has a digital certificate, SSL enabled browsers can communicate securely with your server using SSL. With SSL, you can easily establish a security-enabled Web site on the Internet or on your corporate network.

SSL uses a security handshake to initiate the secure TCP/IP connection between the client and the server. During the handshake, the client and server agree on the security keys that they will use for the session and the algorithms they will use for encryption and to compute message digest or hashes. The client authenticates the server. In addition, if the client requests a document protected by SSL client authentication, the server requests the client's certificate. After the handshake, SSL is used to encrypt and decrypt all information on both the HTTPS requests and the server response, including:

- The URL that the client is requesting
- The contents of any form being submitted
- Access authorization information, such as user names and passwords
- All data sent between the client and the server

The benefits of HTTP using SSL include:

- The target server is verified for authenticity.
- Information is encrypted for privacy.
- Data is checked for transmission integrity.

HTTPS is a unique protocol that combines SSL and HTTP. You need to specify https:// as an anchor in HTML documents that link to SSL protected documents. A client user can open a URL by specifying https:// to request an SSL protected document.

Because HTTPS (HTTP + SSL) and HTTP are different protocols, and usually use different ports (443 and 80, respectively), you can run both secure and

non-secure servers at the same time. As a result, you can choose to provide information to all users using no security, and specify information only to browsers that make secure requests. This is how a retail company on the Internet can allow users to look through merchandise without security, complete order forms, and send their credit card numbers using SSL security. A browser that does not have support for HTTP over SSL naturally cannot request URLs using HTTPS. The non-SSL browsers do not allow users to send forms that need to be submitted securely.

Figure 49 shows how clients can access the same server instance in normal mode (port 80) or encrypted using SSL (port 443).

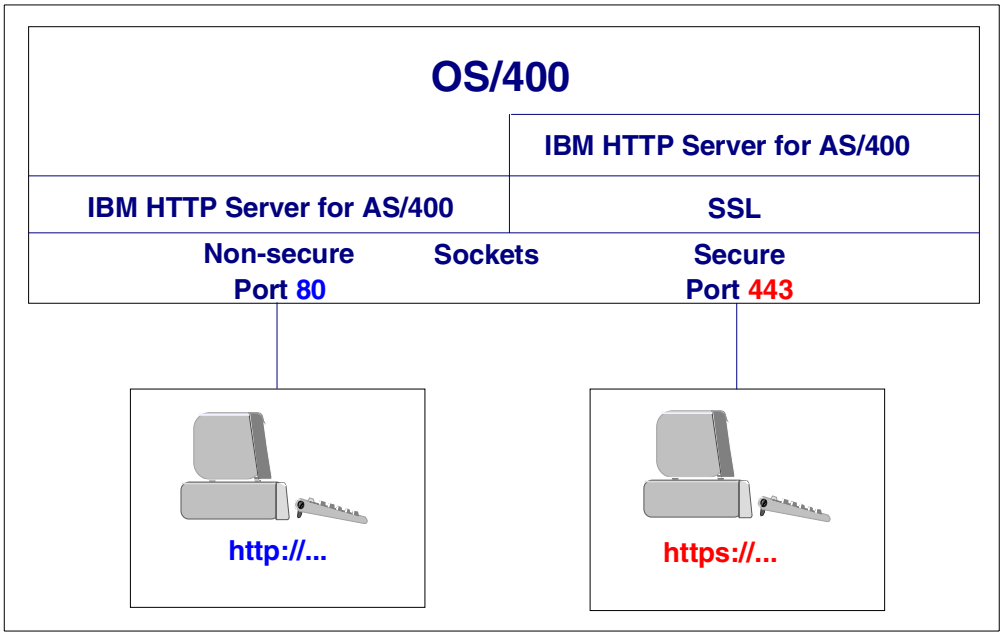


Figure 49. Accessing a secure HTTP session

3.2 Digital certificates and certificate authority

A digital certificate identifies a user or a system and is required before SSL can be used. Once a server has a digital certificate, SSL-enabled browsers, such as the Netscape Navigator, can communicate securely with the server using SSL. A digital certificate consists of:

- Owner's distinguished name
- Owner's public key
- Digital signature of certificate authority (CA)
- Name of the CA
- Issue date of certificate
- Certificate expiration date
- Serial number

Plus, digital certificates have the following characteristics:

- Digital certificates are digital documents that validate the identity of a certificate's owner.
- There are three types of digital certificates: CA, server, and client certificates.

- Digital certificates contain a public key that binds it to an identity.
- Digital certificates are created by trusted third parties called certificate authorities (CA).
- Digital certificates can be distributed freely.
- A digital signature in the digital certificate prevents tampering.

A digital certificate is issued by a certificate authority (CA). CAs are entities that are trusted to properly issue certificates and have controls in place to prevent fraudulent use. They are the equivalent to the Department of Motor Vehicles for a driver's license. An individual may have many certificates from different CAs just as we have many forms of personal identification (Social Security card, insurance card, fitness center membership card). If you can trust a CA, you can be reasonably assured that any certificate they issue properly represents the individual that is holding it.

The certificate authority charges a fee for issuing a certificate:

- Certificate authorities broadcast their public key and distinguished name.
- People add them as a trusted root key to Web servers and browsers.
- This means your server will trust anyone who has a certificate from that CA.
- There are several common CAs in the marketplace.
- Servers and browsers are shipped with several default trusted root keys, and more can be added as needed.

Some examples of universally recognized Internet certificate authorities (CA) include:

- Thawte
- VeriSign
- US Postal Service
- AT&T
- MCI

For testing purposes, or for applications that will be used exclusively in an intranet environment, you may issue digital certificates using an intranet certificate authority. The AS/400 system with Digital Certificate Manager (DCM) can act as an intranet certificate authority.

For secure communications, the receiver must trust the CA that issued the certificate, whether the receiver is a browser or a server. Any time a sender signs a message, the receiver must have the corresponding CA certificate and public key designated as trusted root key.

3.3 AS/400 implementation of Digital Certificate Manager

You can configure your AS/400 system as an intranet Certificate Authority. Digital Certificate Manager (DCM) is a Web-browser based administration facility that allows you to create, manage, and use certificates within an enterprise and with partners of an enterprise. You can use DCM to request digital certificates from such Internet certificate authorities as VeriSign and Thawte.

DCM allows you to create your own intranet certificate authority (CA). You can then use the CA to dynamically issue digital certificates to servers and users (client certificates) on your intranet. When you create a server certificate, DCM

automatically generates the private key and public key for the certificate. You can also use DCM to register and use digital certificates from Verisign or other commercial organizations on your intranet or the Internet.

Digital Certificate Manager is option 34 of OS/400 (5769-SS1 option 34). You must install this option to use DCM. DCM is a link in the AS/400 Tasks page, which runs in the *ADMIN HTTP server instance. Therefore, you must have installed IBM HTTP Server for AS/400 (5769-DG1) and use it to access DCM. In addition, you must install IBM Cryptographic Access Provider licensed program (5769-AC1,or AC2, or AC3) to create certificate keys. These cryptographic products determine the maximum key length permitted for cryptographic algorithms on your AS/400 system. Government export and import regulations determine which version is available in your country. To use all the options available in DCM, you must have *SECOFR and *SECADM authority.

To access the Digital Certificate Manager, click on the hyperlink for **Digital Certificate Manager** from the AS/400 Tasks page. When using Digital Certificate Manager, you can click the Help button on any page at any time to access online help.

3.3.1 Configuring a digital certificate environment

You can use your AS/400 system to configure a digital certificate environment. You can also configure the HTTP server to use digital certificates and run over SSL.

Perform the following series of steps to configure an intranet digital certificate environment using the AS/400 system as a certificate authority:

1. Use DCM to create an intranet CA for one or more AS/400 systems.
2. Using DCM, the intranet CA issues server certificates that can be used in the local server (same AS/400 system where the CA is configured) or exported to a remote server.
3. For the clients to recognize and trust the server certificates issued by the intranet CA, the CA certificate must be installed in the browsers and designated as a trusted root.
4. If the server requests client certificates for client authentication, the users must request and install client certificates in their browsers.
5. The HTTP server must be configured to enable SSL (SSL On) and specify the key-ring file where the server certificate is stored (keyfile). To optionally authenticate client certificates (SSL_ClientAuth client), add PROTECTION/PROTECT directives to protect resources.

3.4 Creating a self-signed certificate

This section describes how to create a self-signed certificate using your AS/400 system as an intranet certificate authority. The steps used in V4R3 and V4R4 of DCM are similar. Watch for different procedures for the different releases. To test your secure Web site before you deploy it, you have to create a self-signed certificate.

Because self-signed certificates are not recognized by a visitor's browser as coming from a trusted third party, they should not be used in customer transaction

situations over the Internet. Use them only on your test and development systems, and for demonstration purposes. You can also use a self-signed certificate for intranet applications.

To obtain a self-signed certificate, perform the following tasks:

1. Create an intranet certificate authority.
2. Create a server certificate with your intranet CA.
3. Configure your HTTP server to use the server certificate.

3.4.1 Creating an intranet certificate authority

Digital Certificate Manager (DCM) allows you to create your own intranet CA in your AS/400 system and use it to issue server and client certificates for testing purposes or applications within your organization. This section outlines the steps you must perform to create a CA on your AS/400 system. You only need to perform this task if the system administrator has not previously created an intranet certificate authority and if you want to use your AS/400 system to issue intranet server certificates. We recommend that you always create a CA on your AS/400 in case you need one for testing.

To create an intranet CA in your AS/400 system, follow these steps:

1. Start the HTTP *ADMIN server on your AS/400 system. From the command line, type:

   ```
   STRTCPSVR SERVER(*HTTP) HTTPSVR(*ADMIN)
   ```

2. Access the AS/400 Tasks page from your browser by entering the URL:

   ```
   http://System_name:2001
   ```

3. You are prompted to enter a user name and password. Sign on with a user that has *SECOFR and *SECADM authority.

   The AS/400 Tasks page appears as shown in Figure 50.

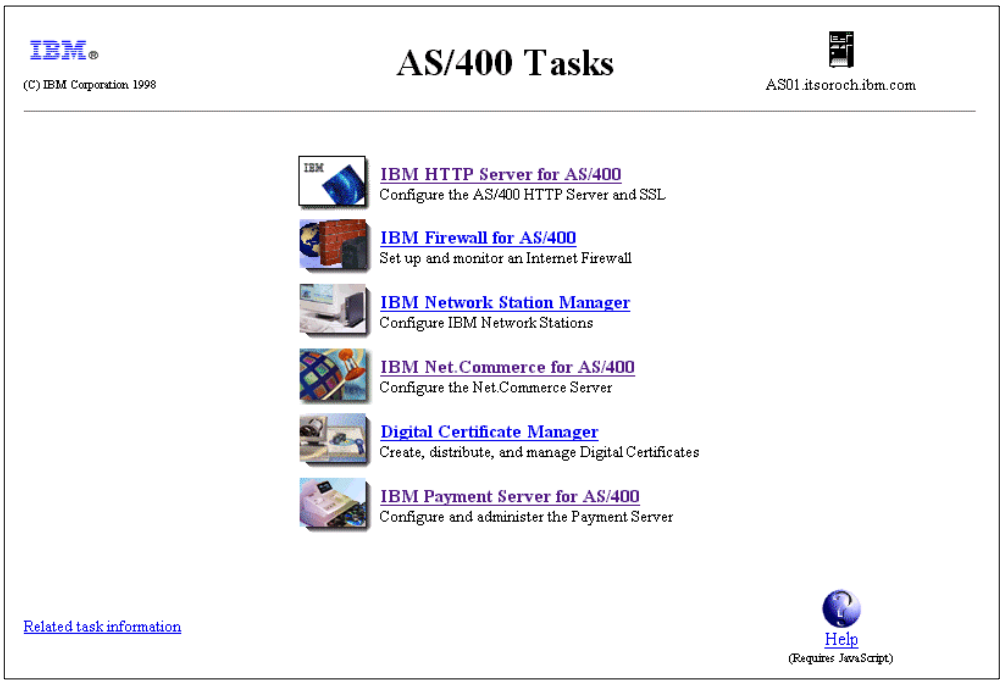


Figure 50. AS/400 Tasks page

4. Click **Digital Certificate Manager**.
5. Click **Certificate Authority (CA)**.
6. Click **Create a Certificate Authority**.

   Note: If a certificate authority (CA) was previously created on your system, the Create a Certificate Authority link does not appear.
7. Complete the Create a Certificate Authority form as shown in Figure 51 on page 52.

   Replace the field values appropriately with your organization's information.

Figure 51. Create a Certificate Authority

Click **OK**.

8. After DCM processes the form, it stores a copy of the CA certificate and other information in the IFS directory /QIBM/USERDATA/ICSS/CERT/CERTAUTH/.

   At this point, you can install the CA certificate in your browser so that it recognizes the certificates issued by the intranet CA. DCM displays a page similar to the page shown in Figure 52. The contents of the page vary based on release.

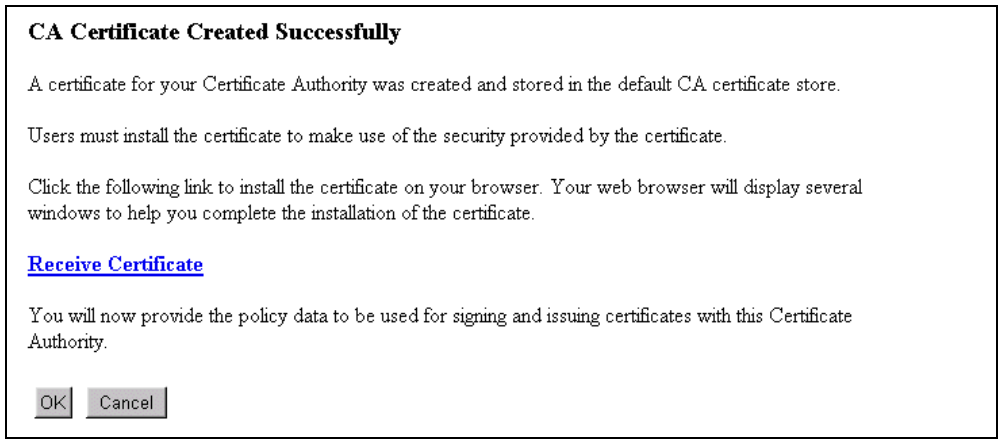


Figure 52. CA Certificate Created Successfully

Click **Receive Certificate** if you want to install the CA certificate in your browser now. Or, click **OK** to proceed to the next setup window, and install the CA certificate in your browser at a later time. Notice the default path and file name where the intranet CA key-ring file is stored.

9. Complete the CA Policy Data form to set the client certificate policy for your CA. See Figure 53.

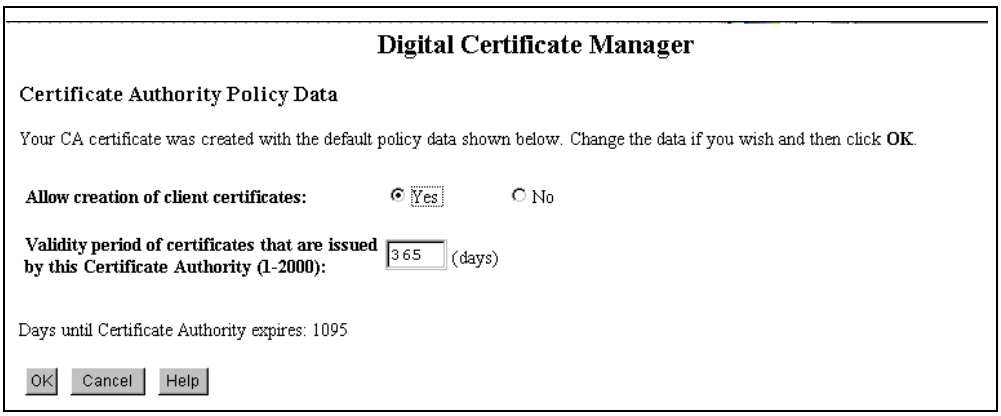


Figure 53. Certificate Authority Policy

This is where you define whether your CA can issue and sign client certificates. If the CA can issue client certificates, indicate the length of time for which the certificates will be valid.

10. The following message appears: `The policy data for the Certificate Authority was successfully changed`. At this point, you can continue to create a server certificate signed by your certificate authority. This allows server authentication by clients that use this system as a server.

    If you are using V4R3, skip to 3.4.2, “Creating a server certificate with your intranet CA (V4R3)” on page 54.

    If you are using V4R4 of DCM, you are presented with a window that allows you to trust this CA for applications. A sample of this is shown in Figure 54 on page 54. On this panel, you should select any applications that are going to use this CA for security. If you have installed and configured SSL in the HTTP instance, an entry is listed for the HTTP server instance name. The entry will be in the form `QIBM_HTTP_SERVER_instancename`, where `instancename` is the name of the HTTP server instance.

Figure 54. Trusting the CA for applications

11. After you select any applications, click **OK**. You receive a message indicating that the system will now create a system certificate.

If you are using V4R4, go to 3.4.3, “Creating a system certificate with your intranet CA (V4R4)” on page 57.

3.4.2 Creating a server certificate with your intranet CA (V4R3)

Immediately after creating the intranet CA, DCM leads you to create a server certificate. To use Secure Sockets Layer (SSL) for secure Web serving, your server must have a digital certificate. When you create a server certificate in DCM, the server certificate and keys are stored in the following default directory and file /QIBM/USERDATA/ICSS/CERT/SERVER/DEFAULT.KYR.

Note: When you create a server certificate, Digital Certificate Manager stores a copy of the CA certificate in the server key ring and designates it as a trusted root.

To create a server certificate with your intranet CA, complete the following steps:

1. Complete the Create a Server Certificate form as shown in Figure 55. Replace the field values with your organization's information.

   The options for the key size are determined by the IBM Cryptographic Access Provider (5769-ACx) licensed program product installed in your system. This is the key size that is used to generate your public and private keys.

Figure 55. Create a Server Certificate page

   By default, the system inserts the fully qualified name of the AS/400 system into the system name field. You can give the server any name. However, the fully qualified TCP/IP host name is usually used for the server name. Some CAs require that the state name be spelled out completely. We recommend that you always use the entire name rather than a short form.

2. Click **OK**.

   The Server Certificate Created Successfully page appears (Figure 56 on page 56).

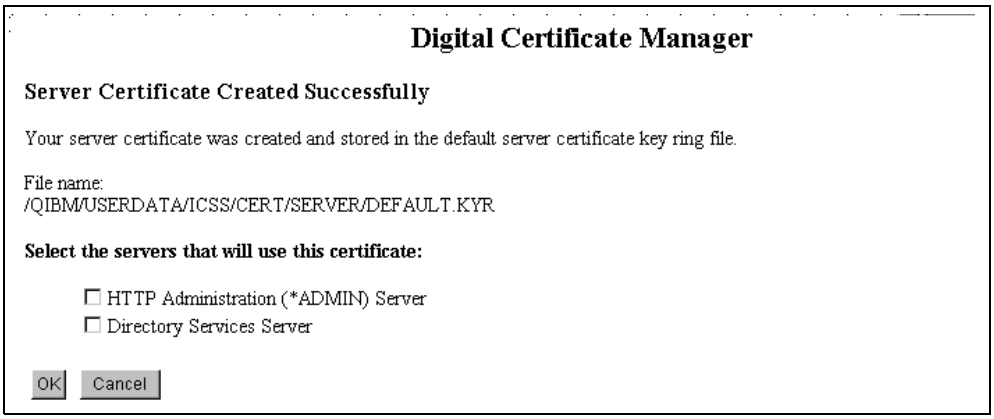


Figure 56. Server Certificate Created Successfully page

From this page, you can select whether the HTTP ADMIN server or the Directory Services server (LDAP) uses this server certificate for SSL connections. Do *not* select any of these options.

3. Copy the file and path name where the server certificate is stored to the clipboard. It is /QIBM/USERDATA/ICSS/CERT/SERVER/DEFAULT.KYR.

   Click **OK**. Click **Done**.

3.4.2.1 Creating a server certificate with an existing intranet CA

The steps to create a server certificate described in the previous section assume that you are creating the intranet CA for the first time. If your administrator has already created an intranet CA and server certificate, you can use the existing server certificate in your HTTP server configuration.

If you want to create a new server certificate using an existing intranet CA, start by clicking **Create a server certificate** under **Server Certificates** in DCM (Figure 57).

Figure 57. Create a server certificate with an existing intranet CA

Select **Local Certificate Authority**, and click **OK**.

The Create Server Certificate page appears (Figure 55 on page 55).

3.4.2.2 Authorizing QTMHHTTP to the key-ring file

You may need to give QTMHHTTP (or the user profile under which your HTTP server runs) authority to the key-ring and stash files. The key-ring and stash files are created with *PUBLIC authority *EXCLUDE. QTMHHTTP (or the user profile under which the HTTP server runs) must at least have read rights to those files.

Perform the following steps:

1. To authorize QTMHHTTP to the key-ring and stash file, type the command:

   ```
   WRKLNK '/QIBM/UserData/ICSS/Cert/Server'
   ```

2. Enter `5` (Next level) to display the files in the directory.
3. Enter `9` (Work with authority) by the key-ring file (DEFAULT.KYR).
4. Enter `1` (Add user). Set User to `QTMHHTTP` and Data Authority to `*R`.
5. Repeat steps one through three to authorize QTMHHTTP to the stash file (DEFAULT.sth).

3.4.3 Creating a system certificate with your intranet CA (V4R4)

Immediately after creating the intranet CA, DCM leads you to create a system (or server) certificate. To use Secure Sockets Layer (SSL) for secure Web serving, your server must have a digital certificate. When you create a system certificate in DCM, the system certificate and keys are stored in the default directory /QIBM/USERDATA/ICSS/CERT/SERVER/. This is also known as the certificate store *SYSTEM.

To create a system certificate with your intranet CA, complete the following steps:

1. Complete the Create a Server Certificate form as shown in Figure 58 on page 58. Replace the field values with your organization's information.

   The options for the key size are determined by the IBM Cryptographic Access Provider (5769-ACx) licensed program product installed in your system. This is the key size that is used to generate your public and private keys.

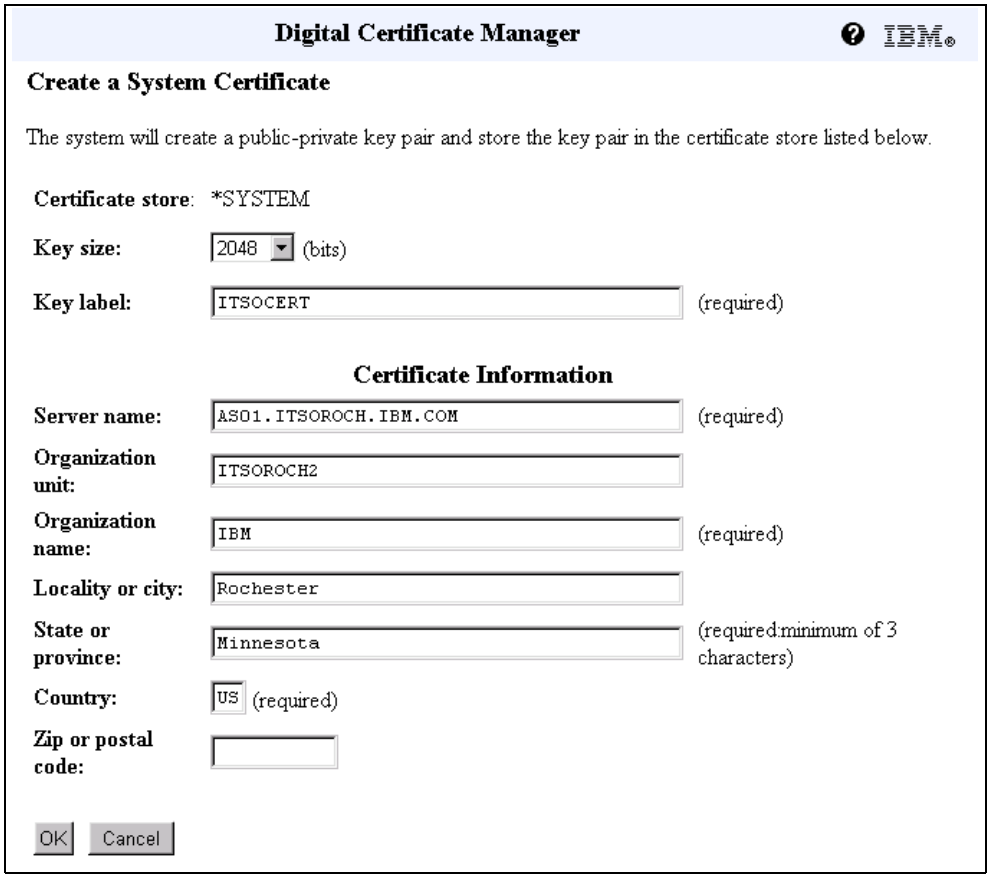


Figure 58. Create a System Certificate page

By default, the system inserts the fully qualified name of the AS/400 system into the system name field. You can give the server any name. However, the fully qualified TCP/IP host name is usually used for the server name. Some CAs require that the state name be spelled out completely. We recommend that you always use the entire name rather than a short form.

2. Click **OK**.

   The System Certificate Created Successfully page appears (Figure 59).

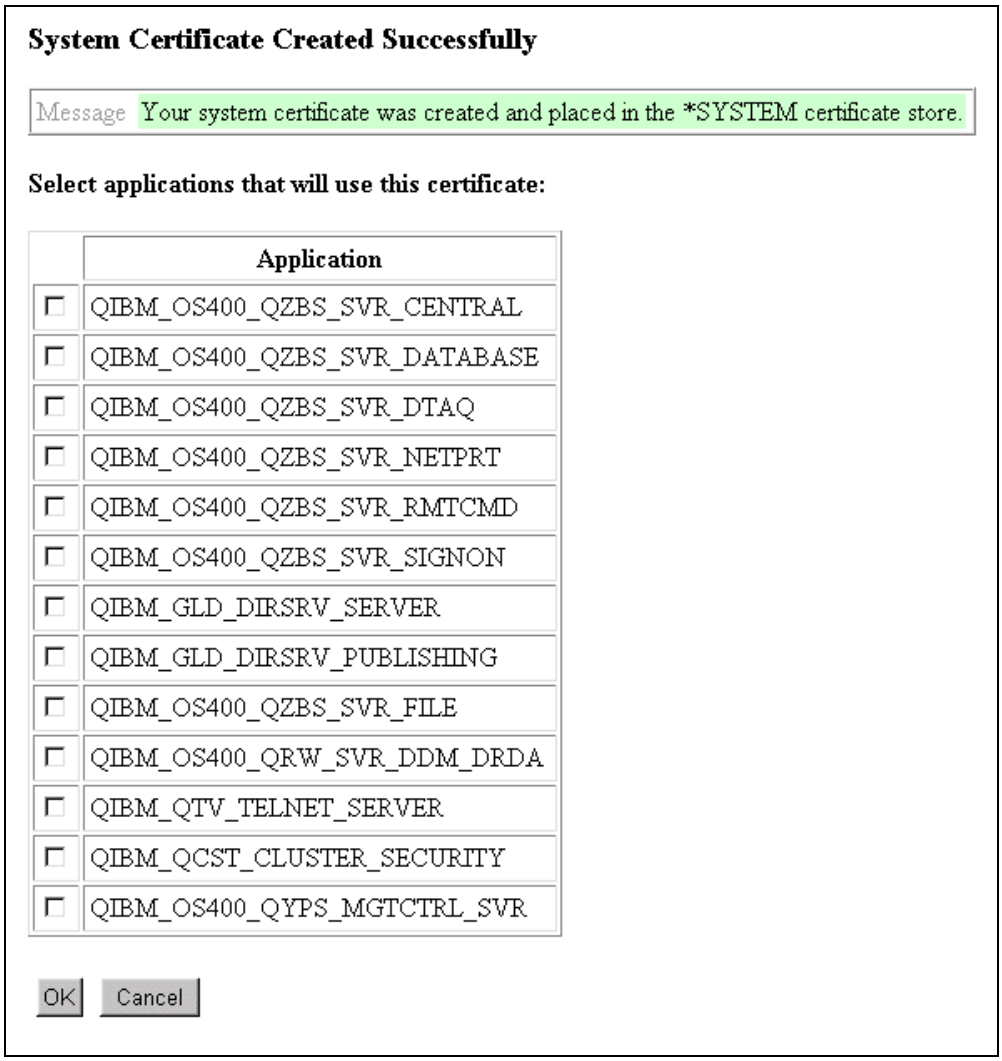


Figure 59. System Certificate Created Successfully page

3. From this page (Figure 59), you can select which applications use this system certificate for SSL connections. After you make your selection, click **OK**. A message appears that confirms that any applications you selected will use this system certificate. Click **Done**.

3.4.3.1 Creating a server certificate with an existing intranet CA

The steps to create a server certificate described in the previous section assume that you are creating the intranet CA for the first time. If your administrator has already created an intranet CA and server certificate, you can use the existing server certificate in your HTTP server configuration.

Complete the following steps to create a new system certificate using an existing intranet CA:

1. Click **System Certificates->Work with certificates** (A) in DCM (Figure 60 on page 60). A right-hand panel appears. Enter the certificate store password when prompted.

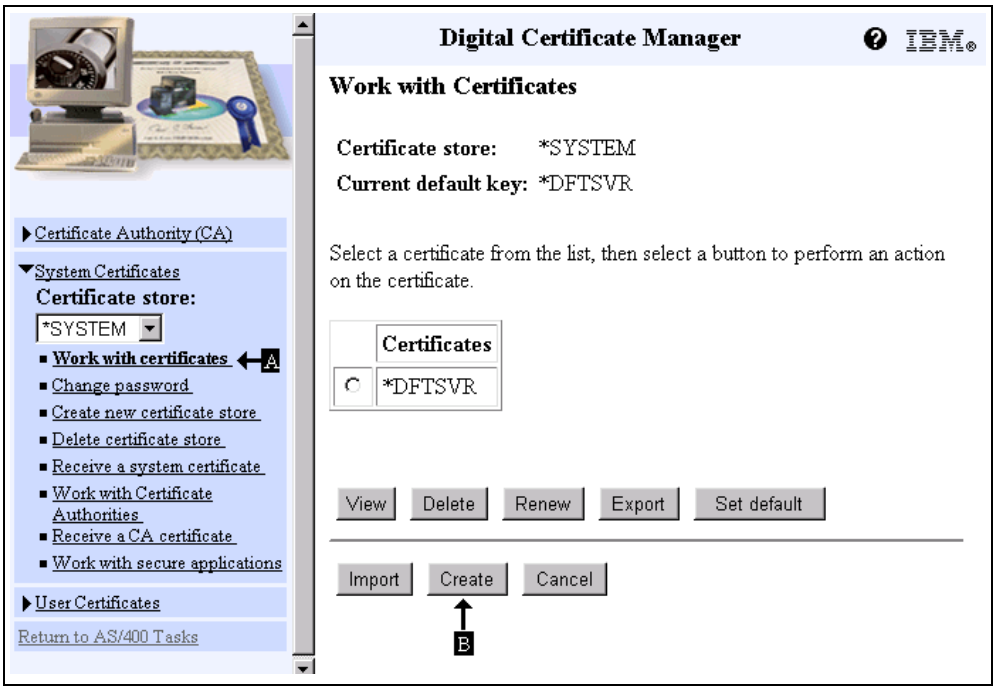


Figure 60. Create a server certificate with an existing intranet CA

2. Click **Create** (B) to create a new system certificate. The display shown in Figure 61 appears.

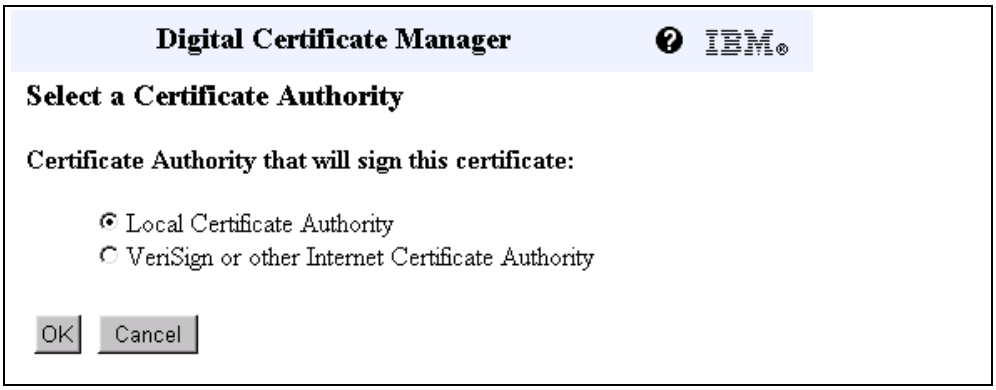


Figure 61. Create a server certificate with an existing intranet CA

3. Select **Local Certificate Authority**, and click **OK**.

You are now at the same place in the process that the system takes you to when you create a system certificate during the create CA process. Go step 1 of 3.4.3, “Creating a system certificate with your intranet CA (V4R4)” on page 57, and complete the procedure found there.

3.4.4 Configuring the Web server to use SSL (V4R3)

The Web server must be configured to run over SSL and use the server certificate you created in 3.4.2, “Creating a server certificate with your intranet CA (V4R3)” on page 54. To configure your HTTP server to run over SSL and use a server certificate, you must perform the following tasks:

1. From Digital Certificate Manager, click **Return to AS/400 Tasks**. The AS/400 Tasks page is displayed (see Figure 50 on page 51).
2. Click **IBM HTTP Server for AS/400**.
3. Click **Configuration and Administration**.
4. Click **Configurations** in the left frame.
5. Select your HTTP configuration file in the drop-down box immediately beneath the Configurations link as shown in Figure 62.

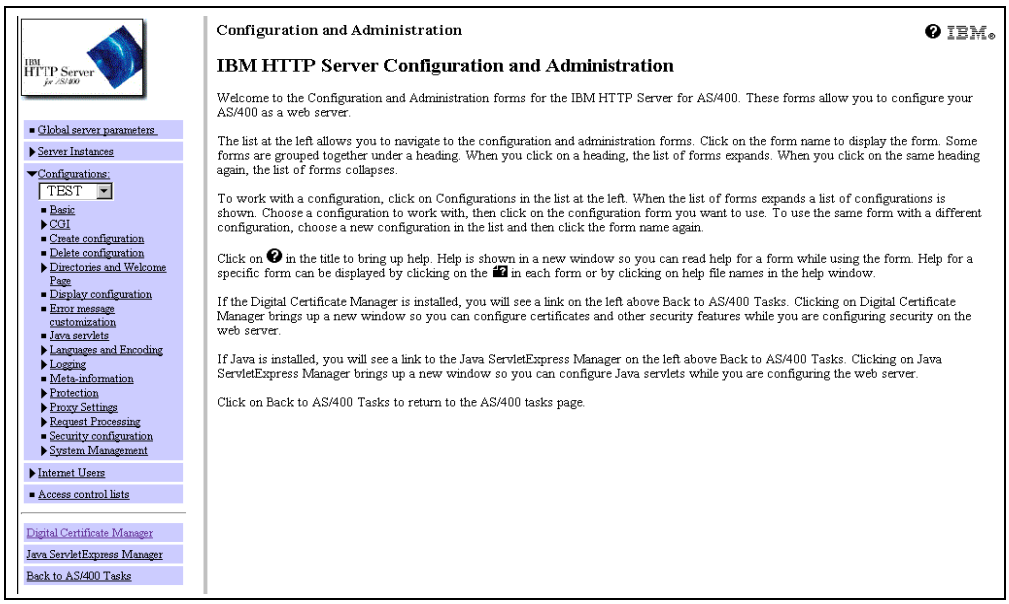


Figure 62. HTTP Server Configuration

6. Click **Security configuration**. Complete the Security configuration page (Figure 63 on page 62).
   a. Check **Allow SSL connections**.
   b. Accept the default SSL port (443) or specify the port you wish to use for SSL.
   c. Deselect **Enable SSL client authentication**.
   d. Add the key-ring path and file name:

      `/QIBM/USERDATA/ICSS/CERT/SERVER/DEFAULT.KYR`

      If you copied it to the clipboard, you can paste it now.

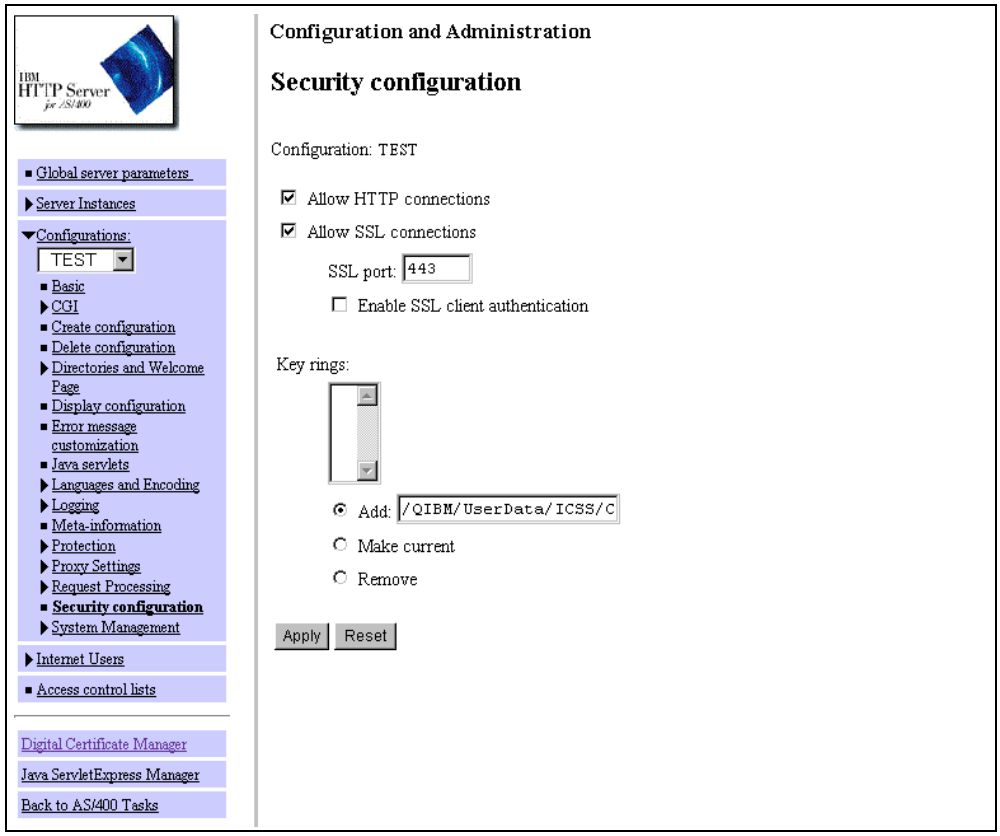


Figure 63. Security configuration page

e. Click **Apply**. You should see the following message at the top of the screen:

```
The configuration file was successfully updated. Server instances that
are using this configuration must be stopped and started for the changes
to take affect.
```

You should also see your key-ring file added in the Key rings box.

7. Stop the server instance, and start it again. In the left pane window, click **Server Instances**.
8. Click **Work with server instances**.
9. From the drop-down box, select your server instance (Figure 64).

Figure 64. Work with server instances

Click **Stop**. Wait until you see this message at the top of your window:

```
The server instance was successfully stopped.
```

10. From the drop-down box, select your server instance (Figure 64).

    Click **Start**.

    You should see this message:

    ```
    The server instance was successfully started.
    ```

You have now successfully configured your Web server to use SSL with server authentication.

3.4.5 Configuring the Web server to use SSL (V4R4)

The Web server must be configured to run over SSL and use the server certificate you created in 3.4.3, “Creating a system certificate with your intranet CA (V4R4)” on page 57. To configure your HTTP server to run over SSL and use a server certificate, you must perform the following tasks:

1. From Digital Certificate Manager, click **Return to AS/400 Tasks**. The AS/400 Tasks page is displayed (Figure 50 on page 51).
2. Click **IBM HTTP Server for AS/400**.
3. Click **Configuration and Administration**.
4. Click **Configurations** in the left frame.
5. Select your HTTP configuration file in the drop-down box immediately beneath the Configurations link as shown in Figure 65 on page 64.

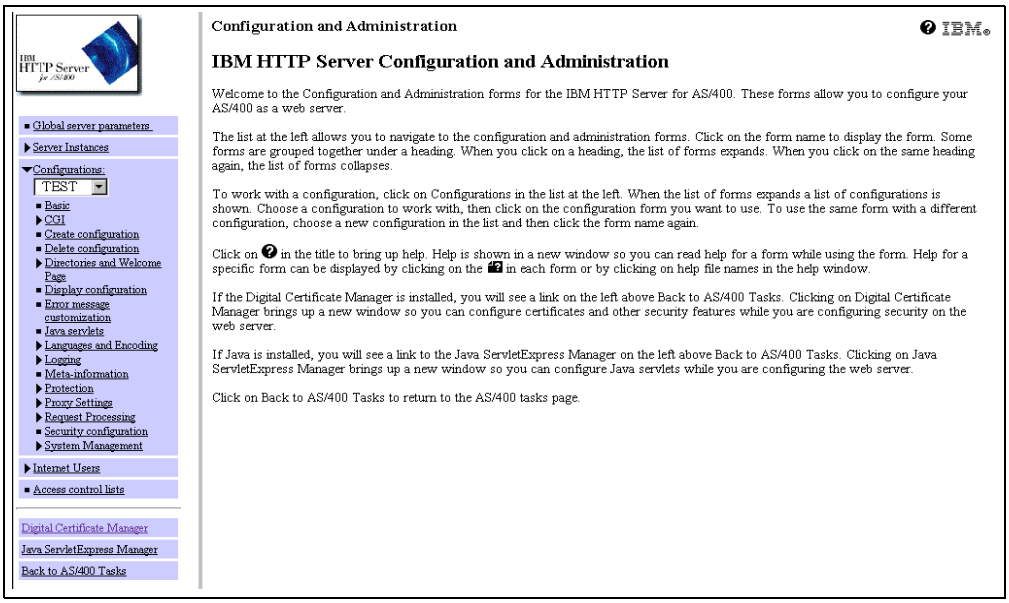


Figure 65. HTTP Server Configuration

6. Click **Security configuration**. The display shown in Figure 66 appears.
7. Complete the Security configuration page:
   a. Check **Allow SSL connections**.
   b. Accept the default SSL port (443), or specify the port you wish to use for SSL.

> **Note**
>
> The **Application ID** value on the page (Figure 66) may show one of several values. The value may be `No ID created yet` or `QIBM_HTTP_SERVER_xxxxxxxx`, where `xxxxxxxx` is the name of this configuration or the name of the configuration that this configuration was built from. Even if all the values are specified correctly, you should still click **Apply** in the next step.

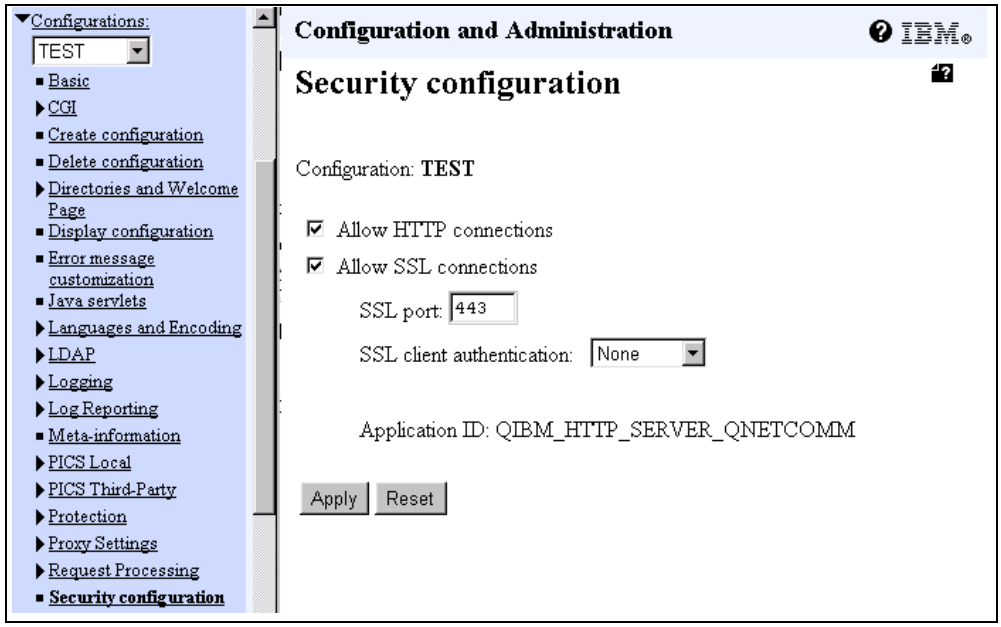


Figure 66. Security configuration page

c. Click **Apply**.

You should see this message at the top of the screen:

```
The configuration file was successfully updated. Server instances that are
using this configuration must be stopped and started for the changes to take
affect.
```

You should also see the application ID added or changed to the value `QIBM_HTTP_SERVER_xxxxxxxx`, where `xxxxxxxx` is the name of this configuration (TEST in this example). Record the application ID because you need to know it to complete the SSL configuration. Enable this HTTP server configuration as a secure application in Digital Certificate Manager (DCM).

8. In the left frame, click **Digital Certificate Manager**. The DCM welcome page is shown in a new browser window.
9. Click **System Certificates->Work with secure applications** in DCM. The panel shown in Figure 67 on page 66 appears. Enter the certificate store password when prompted.

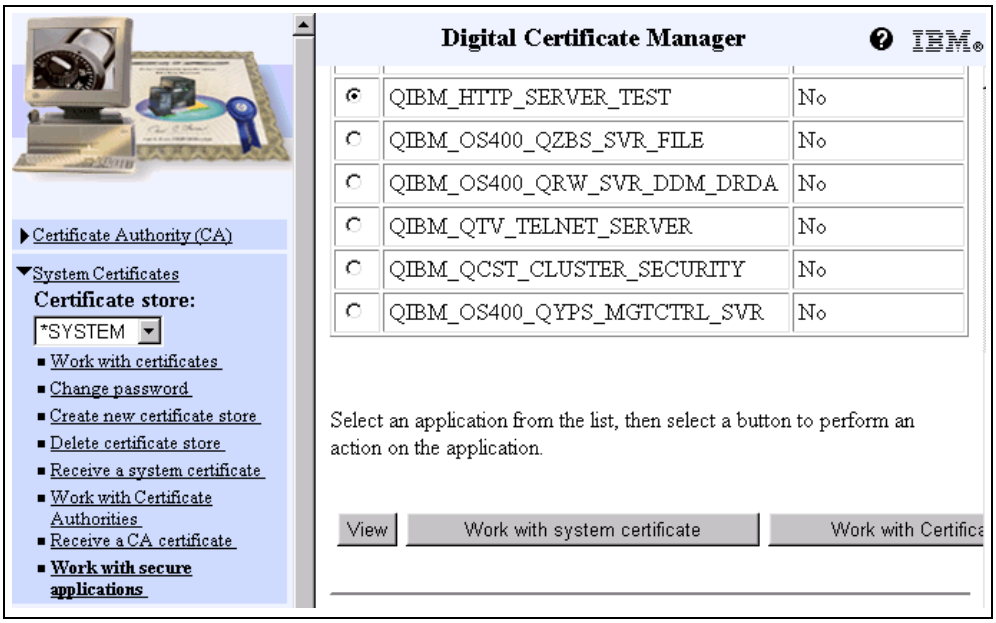


Figure 67. Work with secure applications in DCM

10. Scroll down the list of applications until you find the application ID you want to secure. Select the application, and click **Work with system certificate** in the right frame. This allows you to select which certificate will be used for SSL. The display shown in Figure 68 appears.

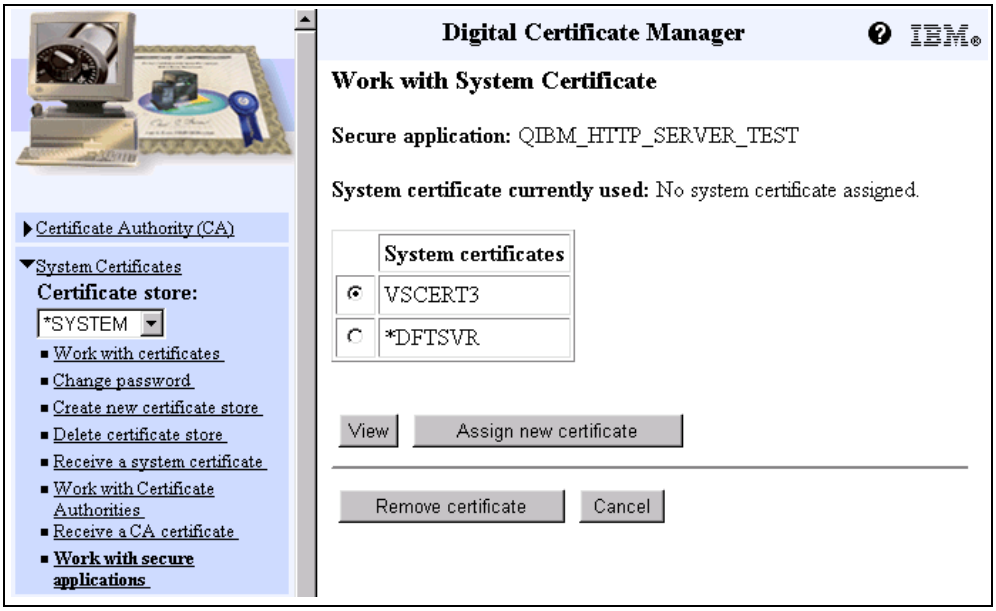


Figure 68. Work with System Certificate

11. An application may only use one certificate. However, one certificate may be used with multiple applications. You may click the **View** button to display the details about the certificates available. Select a system certificate from the list to be used with this application (VSCERT3 in this example), and click **Assign new certificate**. A message displays stating:

```
The system certificate was assigned to the application.
```

When the certificate is assigned, the CA that issued the certificate is set as a trusted root for the application. You can use the **Work with Certificate Authority** button shown in Figure 67 to check the CA assignment for an application.

12. Click **OK**. The display shown in Figure 67 appears. Stop the server instance, and start it again.
13. Return to the browser window that contains the IBM HTTP Server Configuration and Administration page (Figure 65 on page 64). In the left pane window, click **Server Instances**.
14. Click **Work with server instances**.
15. From the drop-down box, select your server instance (Figure 69).

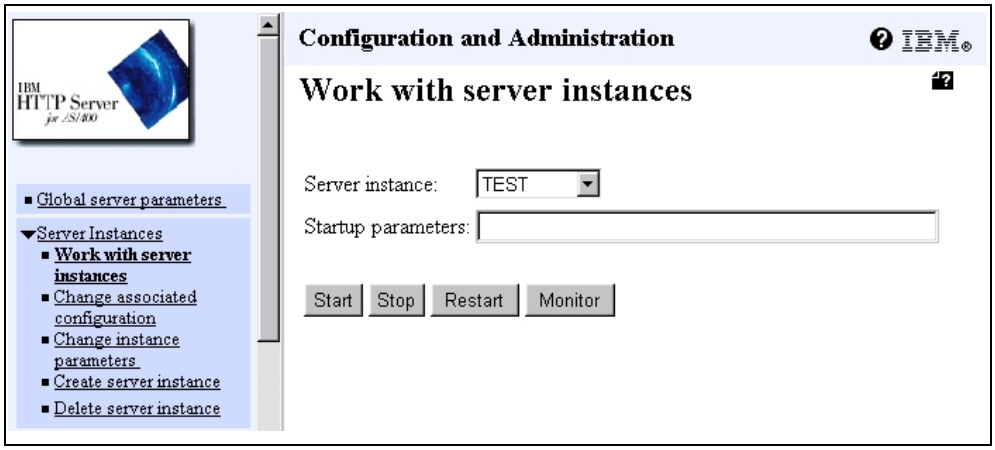


Figure 69. Work with server instances

Click **Stop**. Wait until you see this message at the top of your window:

```
The server instance was successfully stopped.
```

16. From the drop-down box, select your server instance (Figure 69). Click **Start**. You should see this message:

```
The server instance was successfully started.
```

You have now successfully configured your Web server to use SSL with server authentication.

3.5 Requesting a server certificate from an Internet CA

To conduct commercial business on the Internet, you should request your server certificate from an Internet certificate authority. For example, you may consider a CA, such as VeriSign or Thawte, which are widely known by clients, browsers, and servers.

For your private Web network within your own company, university, or group, or for testing purposes, act as your own CA using Digital Certificate Manager (DCM). Section 3.4, “Creating a self-signed certificate” on page 49, explains this procedure.

This section describes how to obtain a server certificate from an Internet certificate authority. To use a server certificate issued by an Internet CA, perform these steps:

1. Request the server certificate from an Internet CA.
2. Receive a server certificate for this server.
3. Configure the HTTP server to use SSL and server authentication.

3.5.1 Requesting a server certificate from an Internet CA (V4R3)

To use SSL for secure Web serving, your server must have a digital certificate. You can use an intranet CA to issue a server certificate (see 3.4, “Creating a self-signed certificate” on page 49), or you can use an Internet CA.

When you choose to use an Internet CA to issue a server certificate, you must first request the certificate. Follow these steps:

1. From the Digital Certificate Manager (DCM) page, click **Server Certificates** in the left-hand frame to display an extended list of server tasks.
2. Click **Create a server certificate** from the list to display the Select a Certificate Authority page.
3. Select **VeriSign or another Internet Certificate Authority** as shown in Figure 70.

Figure 70. Requesting a certificate from VeriSign or other Internet CA

Click **OK** to display the Create a Server Certificate form.

4. Complete the Create a Server Certificate form as shown in Figure 71. Replace the field values with your organization information.

   The options for the key size are determined by the IBM Cryptographic Access Provider (5769-ACx) licensed program product installed in your system. This is the key size that will be used to generate your public and private keys.

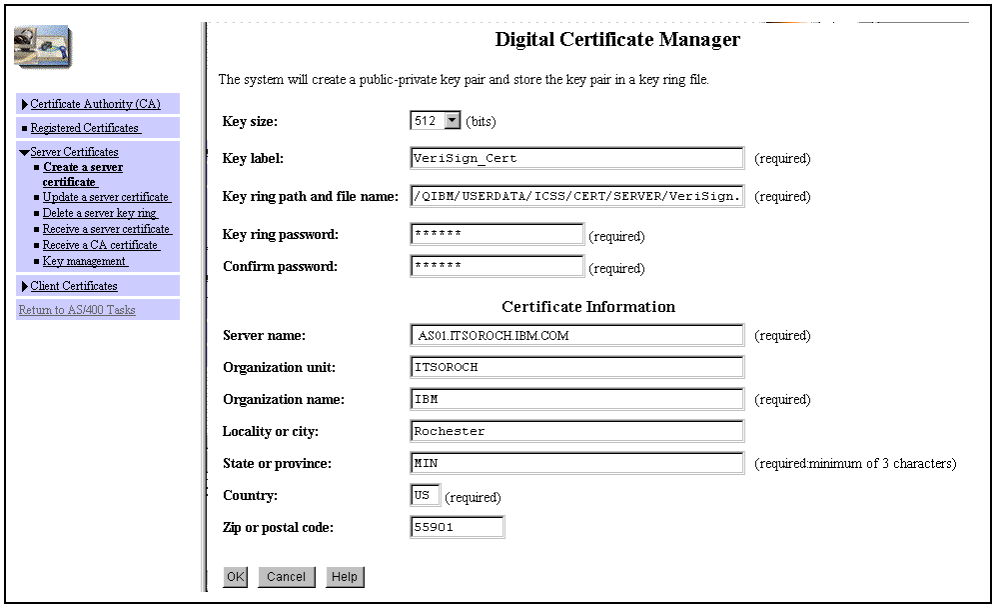


Figure 71. Request a server certificate from an Internet CA

By default, the system inserts the fully qualified name of the AS/400 system into the system name field. *Do not change this name.* This is the name used to describe your server. You can give the server any name, although the fully qualified TCP/IP host name is usually used for the server name.

5. Click **OK** to process the Create a Certificate Request form.

   You receive the Server Certificate Request Created page as shown in Figure 72.

Figure 72. Server certificate request generated by DCM

Note: Do not click Done or close the browser yet. You need to cut and paste the certificate request when you submit the Certificate Signing Request to the Internet CA.

6. Copy the Server Certificate Request to your clipboard. Start at `-----BEGIN NEW CERTIFICATE REQUEST-----` and end at `-----END NEW CERTIFICATE REQUEST-----`. Click **Done** to close the page.

7. Follow your Internet CA procedures to paste the certificate request. For example, to request a certificate from VeriSign, follow the instructions that are described at the Web site: `http://www.verisign.com`

   When VeriSign is satisfied that you meet all of its requirements, it e-mails the secure server certificate to you. You should receive it in three to five business days. Other certificates authorities have their own procedures.

3.5.2 Receiving a server certificate for your server (V4R3)

After you receive the certificate from the Internet CA, you need to copy the signed server certificate to a text file that DCM can access when you perform the Receive server certificate task. Perform the following steps:

1. Copy the signed server certificate presented to you by the Internet CA to your clipboard. Start at `-----BEGIN CERTIFICATE REQUEST-----`, and end at `-----END CERTIFICATE REQUEST-----`.
2. Paste the signed server certificate in your clipboard into a .txt file. Use a text editor of your choice, for example Notepad, to create a .txt file and paste the server certificate issued by the Internet CA.
3. Save the file in your AS/400 system IFS. Use a mapped network drive, and save the .txt file that contains the server certificate issued by the Internet CA in the path (enter a file name of your choice) /QIBM/USERDATA/ICSS/CERT/SERVER/rcvcert.txt.
4. In DCM, click **Receive a server certificate**, and complete the Receive a Server Certificate page (Figure 73).

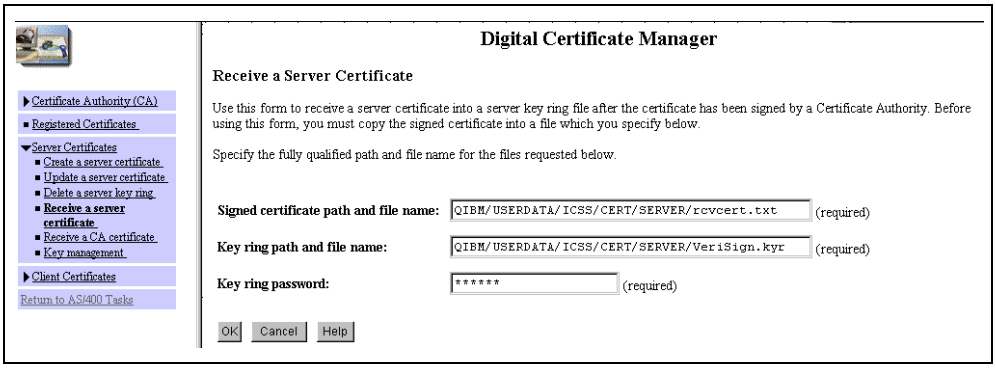


Figure 73. Receiving a server certificate issued by an Internet CA

5. The Certificate Received page is displayed. You have the option to use the received certificate with the ADMIN or LDAP server. *Do not select these options*. Click **OK**.
6. You should receive a Server Configuration Status message indicating the server certificate operations are complete. Click **Done**.
7. You must now set the key as the default key. In DCM, click **Key management**. Complete the Key Management page, and select **Work with keys** (Figure 74).

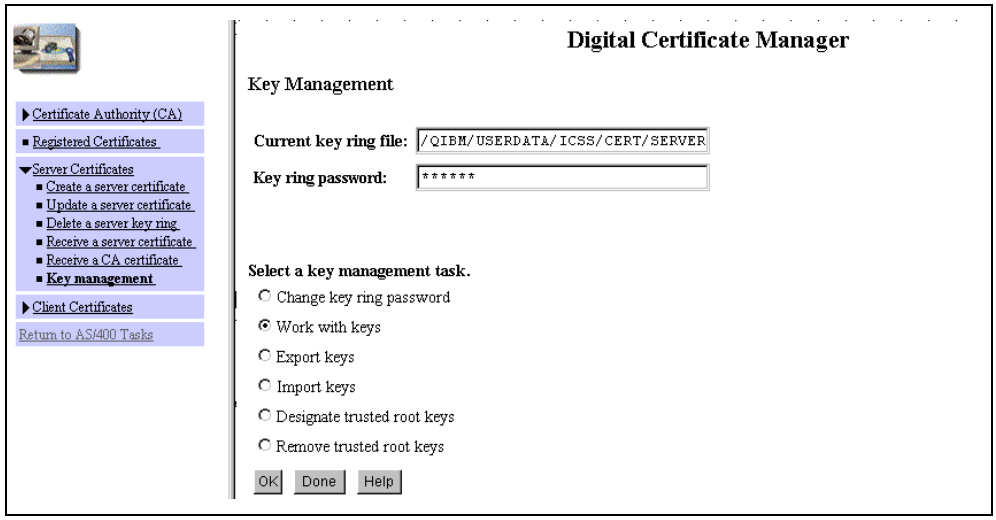


Figure 74. Key Management page

8. Select the key with the label corresponding to the certificate you received from the Internet CA (VeriSign_Cert in our example). Select **Set key** to be the default, and click **OK**.

3.5.3 Requesting a system certificate from an Internet CA (V4R4)

To use SSL for secure Web serving, your system must have a digital certificate. You can use an intranet CA to issue a system certificate (see 3.4, “Creating a self-signed certificate” on page 49), or you can use an Internet CA.

When you choose to use an Internet CA to issue a system certificate, you must first request the certificate. Follow these steps:

1. Click **System Certificates->Work with certificates** (A) in DCM (Figure 75 on page 72). A right-hand panel appears. Enter the certificate store password when prompted.

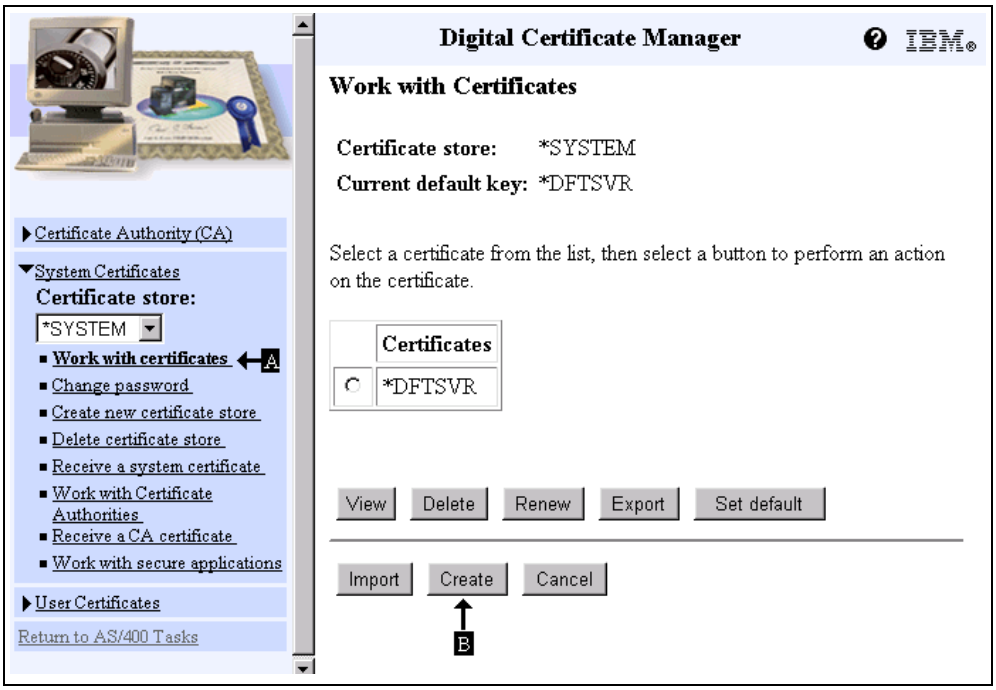


Figure 75. Requesting a certificate from VeriSign or other Internet CA

2. Click **Create** (B) to create a new system certificate. The display shown in Figure 76 appears.

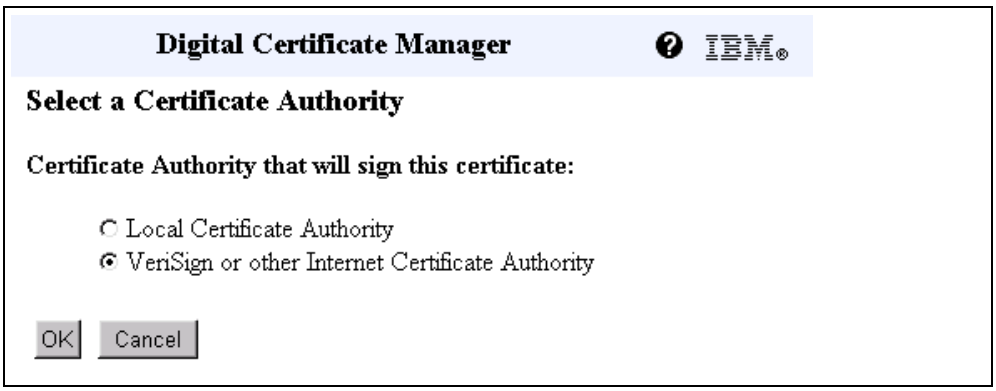


Figure 76. Create a server certificate with an Internet CA

3. Select **VeriSign or other Internet Certificate Authority.** Click **OK**.
4. Complete the Create a Server Certificate form as shown in Figure 78. Replace the field values with your organization's information.

   The options for the key size are determined by the IBM Cryptographic Access Provider (5769-ACx) licensed program product installed in your system. This is the key size that is used to generate your public and private keys.

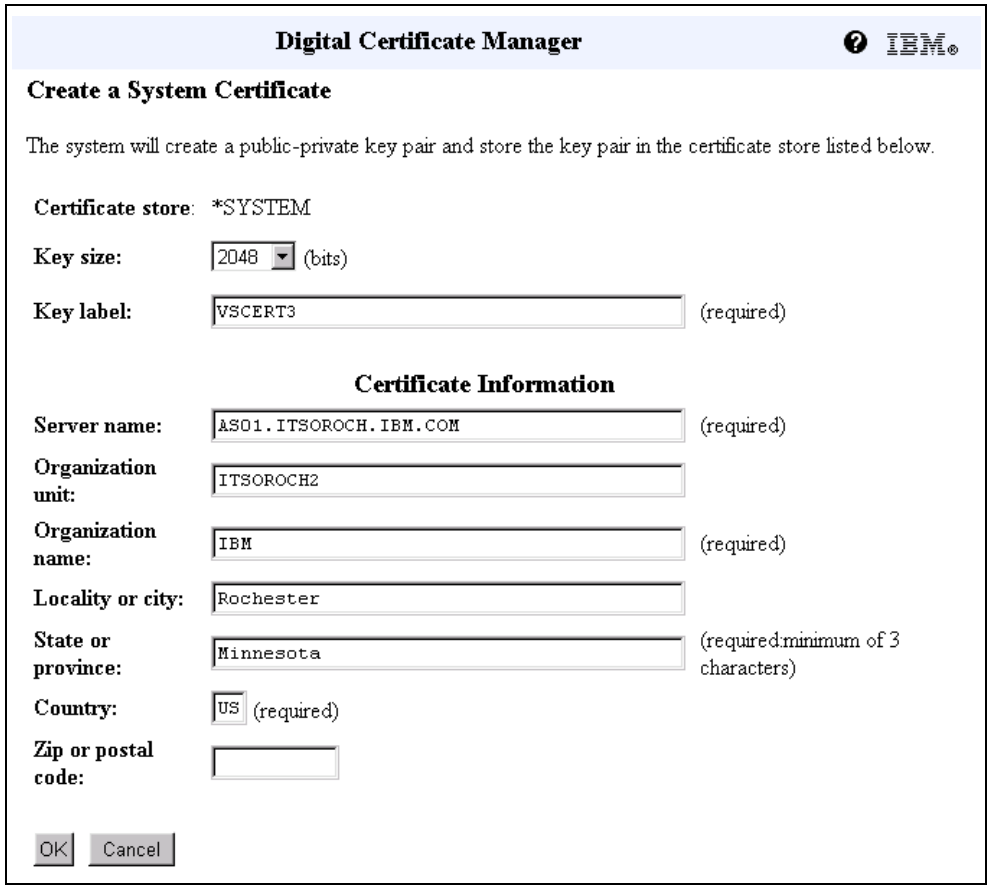


Figure 77. Create a System Certificate page

By default, the system inserts the fully qualified name of the AS/400 system into the system name field. *Do not change this name.* This is the name used to describe your server. You can give the server any name. However, the fully qualified TCP/IP host name is usually used for the server name.

5. Click **OK**. The System Certificate Request Created page appears (Figure 78 on page 74).

Figure 78. Server Certificate Request Created page

Note: Do not click Done or close the browser yet. You need to cut and paste the certificate request when you submit the Certificate Signing Request to the Internet CA.

6. Copy the Server Certificate Request to your clipboard. Start at `-----BEGIN NEW CERTIFICATE REQUEST-----` and end at `-----END NEW CERTIFICATE REQUEST-----`. Click **Done** to close the page.
7. Follow your Internet CA procedures to paste the certificate request. For example, to request a certificate from VeriSign, follow the instructions that are described at the Web site: `http://www.verisign.com`

   When VeriSign is satisfied that you meet all of its requirements, it e-mails the secure server certificate to you. You should receive it in three to five business days. Other certificates authorities have their own procedures.

3.5.4 Receiving a system certificate (V4R4)

After you receive the certificate from the Internet CA, you need to copy the signed server certificate to a text file that DCM can access when you perform the Receive server certificate task. Perform the following steps:

1. Copy the signed server certificate presented to you by the Internet CA to your clipboard. Start at `-----BEGIN CERTIFICATE REQUEST-----`, and end at `-----END CERTIFICATE REQUEST-----`.
2. Paste the signed system certificate from your clipboard into a .txt file. Use a text editor of your choice, for example Notepad, to create a .txt file, and paste the server certificate issued by the Internet CA.
3. Save the file in your AS/400 system IFS. Use a mapped network drive and save the .txt file that contains the server certificate issued by the Internet CA. In our example, we created a directory structure and file with the path /verisign/certificates/vscert3.txt.
4. In DCM, click **Receive a system certificate**. The display shown in Figure 79 appears.

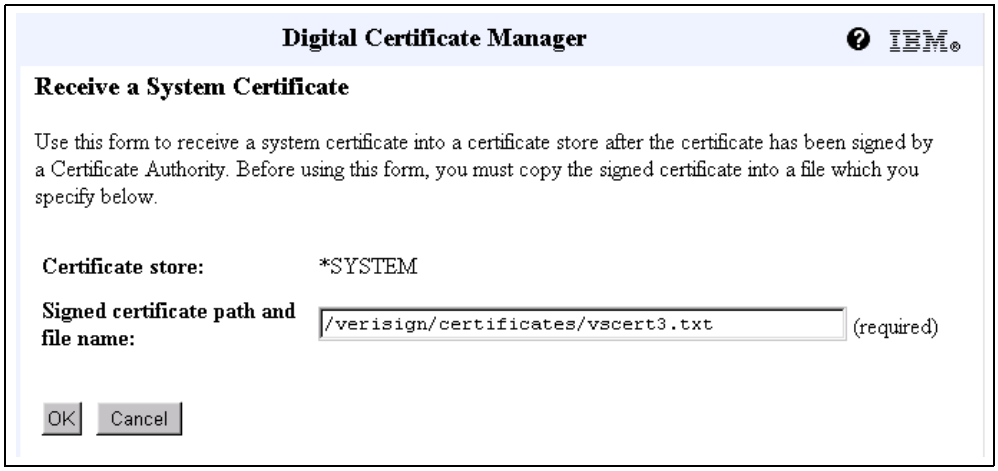


Figure 79. Receiving a System Certificate issued by an Internet CA

5. Complete the Receive a System Certificate page (Figure 79) by typing the directory path and file name where you stored the signed system certificate received for the Internet CA. Click **OK**. The Certificate Received page (Figure 80) appears.

Figure 80. Key management page

6. Click **OK**. You return to the Receive a System Certificate page (Figure 79). Click **Cancel**. Specify which applications will use this system certificate.

3.5.5 Configuring the HTTP server to use SSL

This task is described in 3.4.4, “Configuring the Web server to use SSL (V4R3)” on page 60, and 3.4.5, “Configuring the Web server to use SSL (V4R4)” on page 63.

3.6 Using the SSL configuration

In 3.3, “AS/400 implementation of Digital Certificate Manager” on page 48, we show how to configure the test AS/400 system. We now use this configuration to access a Web page, using Netscape Communicator. If you use another browser,

you use similar dialogs to help you control the security of the application. To access the page, we enter:

```
https://AS400ABC/securepg.html
```

Before the page appears, we are presented with security dialogs. Since we did not obtain a digital certificate from a universally recognized Internet Certificate Authority, the browser displays the warning dialog shown in Figure 81.

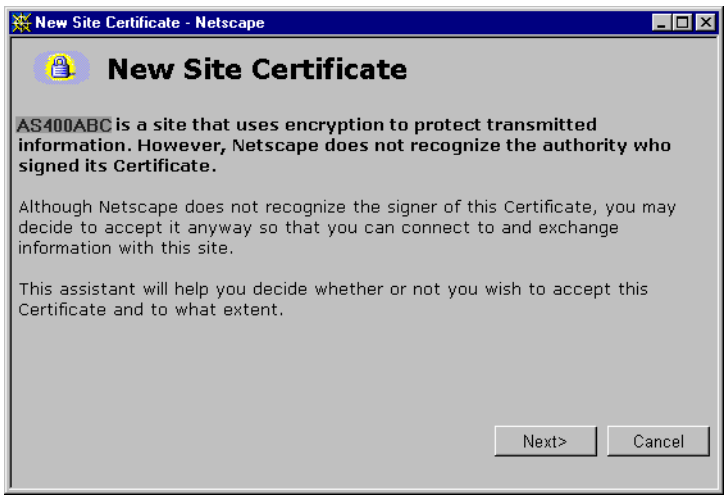


Figure 81. New Site Certificate

Clicking the **Next** button causes the dialog shown in Figure 82 to appear. This dialog allows us to display more information about the certificate that is being presented.

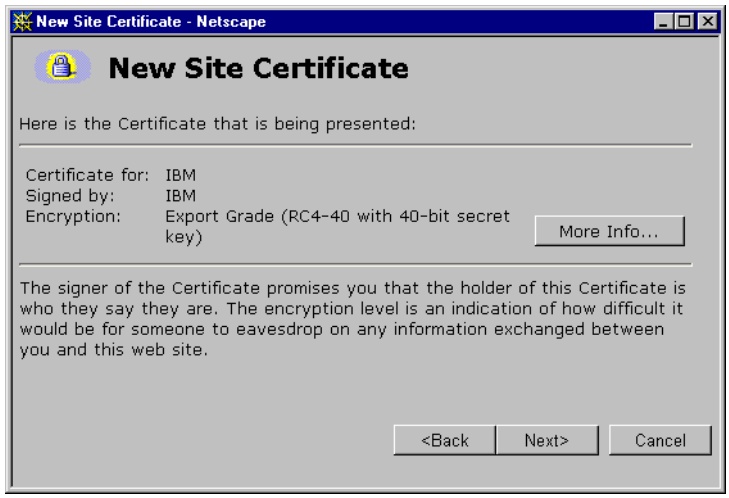


Figure 82. New Site Certificate information

Clicking the **More Info...** button shows a dialog, which contains information about the issuer of the certificate (Figure 83).

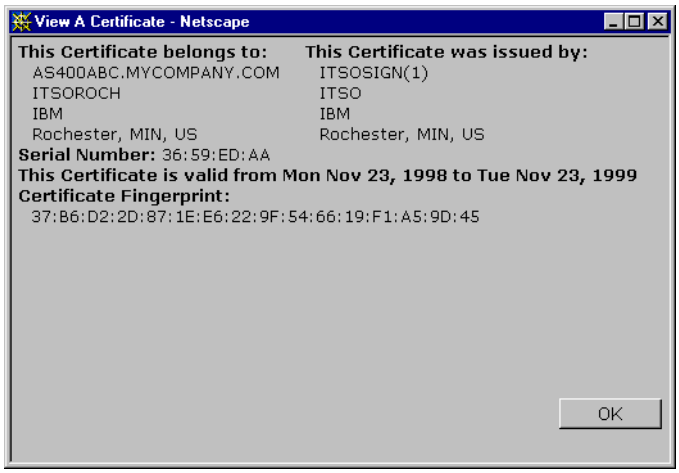


Figure 83. View a Certificate

Clicking the **Next** button in the dialog displayed in Figure 82 causes the dialog shown in Figure 84 to appear.

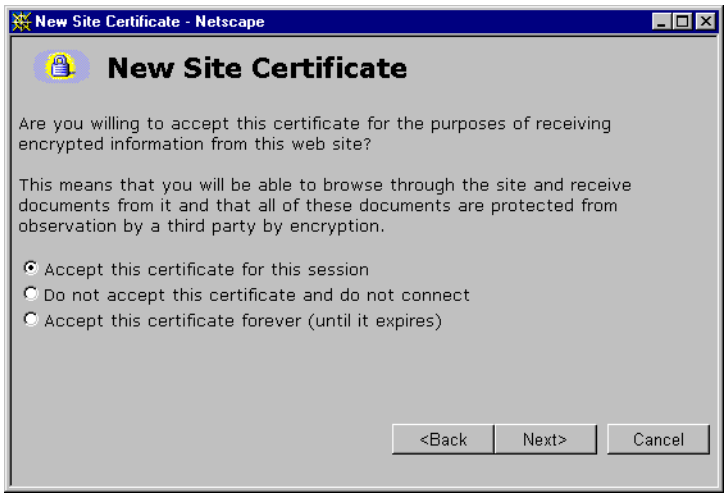


Figure 84. New Site Certificate acceptance dialog

The dialog shown in Figure 84 allows us to choose how we want to deal with the certificate received from the remote site. In this case, we recognize that it is a site that we can trust. We select the **Accept this certificate for this session** radio button and click the **Next** button. This causes the dialog shown in Figure 85 on page 78 to appear.

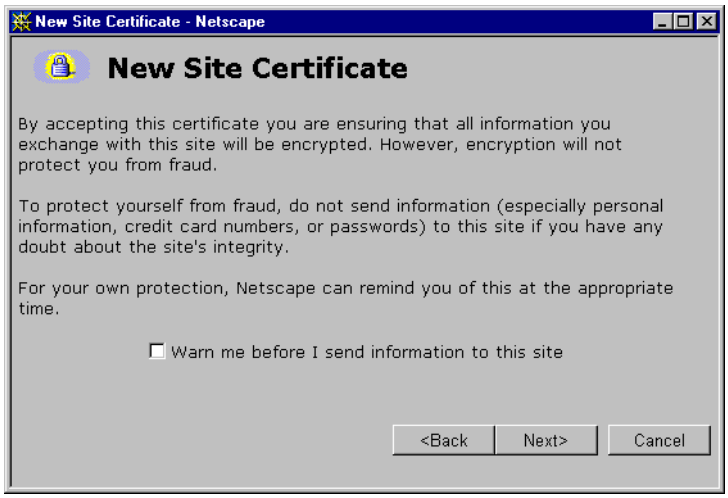


Figure 85. Netscape certificate warning dialog

After we accept the certificate, the browser displays a final warning that allows us to choose to be reminded with further warning messages. Clicking the **Next** button starts the SSL session with the remote system. If this is the first time that we have requested a secure document, we are presented with the dialog shown in Figure 86.

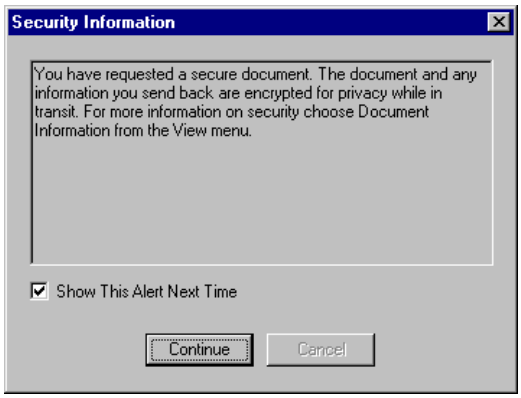


Figure 86. Netscape Security Information dialog

We can use the check box shown in Figure 86 to control whether we want to see this warning dialog in the future. If we click the **Continue** button, the Web page displays as shown in Figure 87. The lock icon shown in the lower left corner of the browser indicates that we are running under an SSL session.

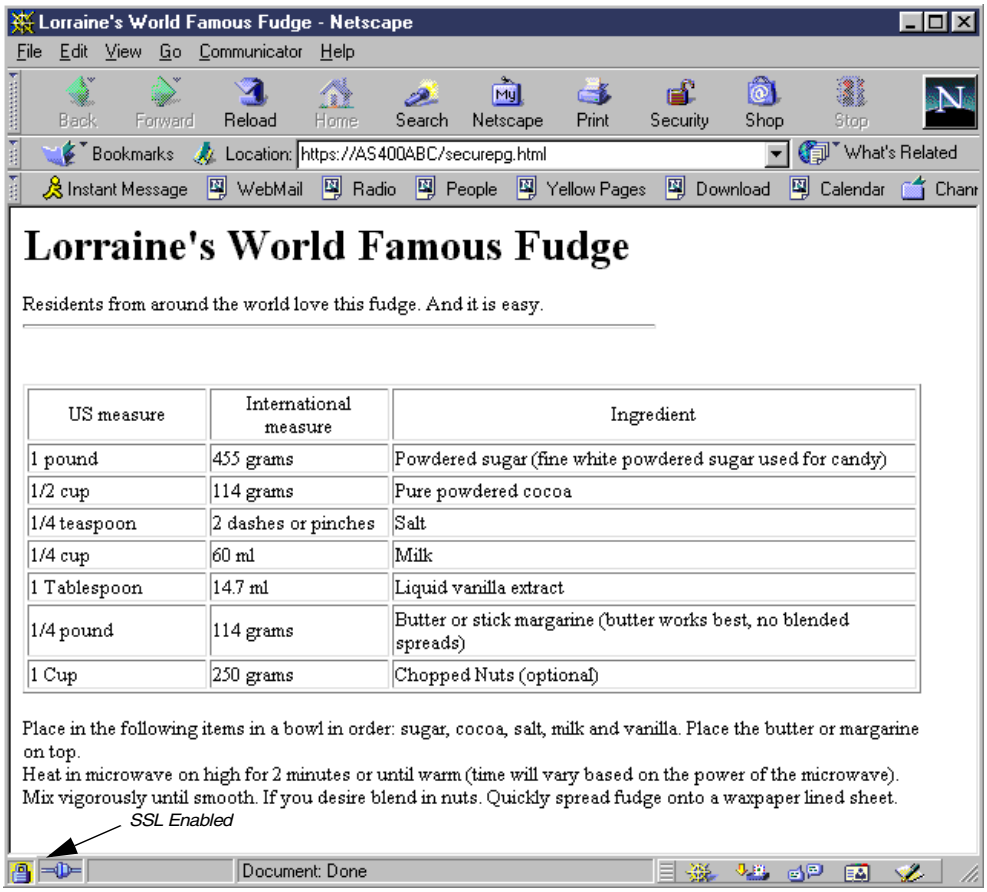


Figure 87. Welcome page under SSL

3.6.1 Additional resources

For additional information, consult the following resources:

- *HTTP Server for AS/400 Webmaster's Guide,* GC41-5434
- *Securing Your AS/400 from HARM on the Internet,* SG24-4929
- `http://publib.boulder.ibm.com/pubs/html/as400/ic2924/info/index.htm`

  Click **Internet->Digital certificate management**

- `http://www.software.ibm.com/webservers/`
- `http://www.ibm.com/security`
- `http://www.internet.ibm.com/commercepoint/registry/`
- `http://www.rsa.com`

Chapter 4. Configuring PPP and SLIP

This chapter describes the background, functionality, and configuration of the protocols used for connecting systems in a point-to-point manner, also known as wide area network (WAN). Scenarios of both Point to Point Protocol (PPP) and Serial Line Internet Protocol (SLIP) are shown. For more information on the PPP and SLIP protocols, refer to *TCP/IP Configuration and Reference,* SC41-5420.

Note

The windows shown for PPP configuration are from a V4R3 system. The information may be formatted differently in other releases. Use the examples here as a guide.

4.1 Introduction to WAN connectivity using PPP and SLIP

PPP and SLIP allow for WAN connectivity (point-to-point connectivity). Examples are AS/400 to AS/400, PC to AS/400, and AS/400 to ISP (Internet).

When two systems are physically connected, it is typically referred to as a *point-to-point connection* or *link*. Several different protocols, such as the TCP/IP PPP, SLIP, X.25, and Frame Relay, are viewed as point-to-point protocols. However, PPP and SLIP offer a lower cost, more efficient connection alternative to X.25 and Frame Relay. Support for PPP and SLIP is included on your AS/400 system as part of WAN connectivity.

Table 4 shows some of the different WAN alternatives available. Please note that the costs are not intended to reflect current pricing.

Table 4. WAN connection alternatives

Service	Line speed	Required equipment	DTE/DTC interface	Relative cost/month
Analog (leased and switched)	33.4 Kbps or less	Modem	RS232 Asynchronous	$20-$150
Digital Data service (DDS)	56 Kbps or less	CSU/DSU	X.21/V.35/RS-449 Synchronous	$50-$500
Switched -56	56 Kbps	CSU/DSU with V.25bis dial	V.35/RS-449 Synchronous	$50-$250
ISDN switched	56, 64, 112 or 128 Kbps	ISDN terminal adapter	RS232 Asynchronous	$50-$250
Fractional T1	56 Kbps to 1544 Kbps	CSU/DSU or T1 mux	X.21/V.35/RS-449 synchronous	$100-$2000
T1/E1	56 Kbps to 1544/2048 Kbps	CSU/DSU or T1 mux	X.21/V.35/RS-449 synchronous	$350-$2000

The connection methods include:

- **Analog phone lines**: Use standard V.42bis modems, or the latest technology: X2 or 56Flex (V.90), which extends the speed to 56Kbs in one direction.
- **DDS or digital service lines**: The most basic form of digital services. Allows for speeds up to 56Kbps. Requires a special box called Channel Service Unit/Data Service Unit (CSU/DSU), which replaces the modems used in an analog scenario.
- **Switched-56**: Another digital service. Connects via CSU/DSUs and includes a dialing pad from which you enter the phone number of the remote host. Is only available in North America.
- **ISDN**: Switched end-to-end digital connectivity. ISDN can carry both voice and data over the same connection. There are different types of ISDN services, with Basic Rate Interface (BRI) being the most common. BRI consists of two 64 Kbps B channels to carry customer data and a D channel to carry signaling data. The two B channels can be linked together to give a combined rate of 128 Kbps. The ISDN “modems” are called Terminal Adapters.
- **T1/E1**: A T1 connection bundles together twenty-four 64 Kbps channels over a 4 wire copper circuit, giving a total bandwidth of 1544 Kbps. An E1 circuit in Europe bundles thirty-two 64 Kbps lines for a total of 2048 Kbps. They typically connect using a V.35 interface to a CSU/DSU and a synchronous protocol.
- **Fractional T1**: A customer can lease any 64 Kbps sub-multiple of a T1 line.

4.2 Point-to-Point Protocol (PPP)

One goal of PPP is to allow interoperability among the remote access software of different manufacturers. Another goal is to allow the same physical communication line to be used by multiple network communication protocols. This section describes different PPP concepts and gives a basic understanding of the PPP protocol and its use.

The PPP protocol is described in multiple RFC standards:

- RFC1661 Point-to-Point Protocol
- RFC1662 PPP on HDLC-like framing
- RFC1994 PPP challenge Handshaking Protocol (CHAP)

More information about the RFCs can be found at: `http://www.rfc-editor.org`

4.2.1 What PPP is

PPP is a method of connecting two hosts to each other over, for example, a phone line or a leased line. A common example is a PPP connection that is established between a remote office and the home office to transfer data using the TCP/IP protocol. This, for example, could be from a PC to the office system as shown in Figure 88.

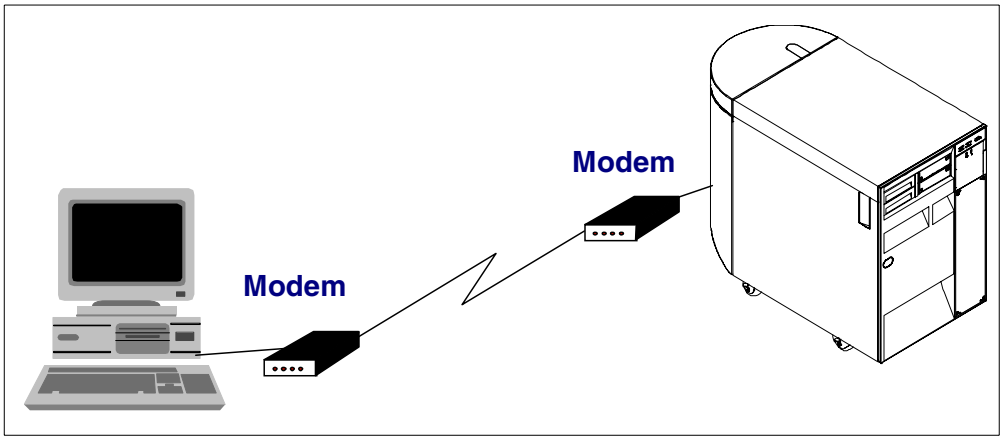


Figure 88. PPP connection from a remote location to the home office

The remote system could then access AS/400 applications, such as Lotus Domino, or work as a terminal using the PC5250 application. Another example (Figure 89) shows an AS/400 system that wants to connect to an Internet Service Provider (ISP), such as the IBM Global Network (IGN). This would allow clients on the Internet to connect to the AS/400 system through the Internet provider. Or the AS/400 could act as a gateway to the Internet for the local workstations.

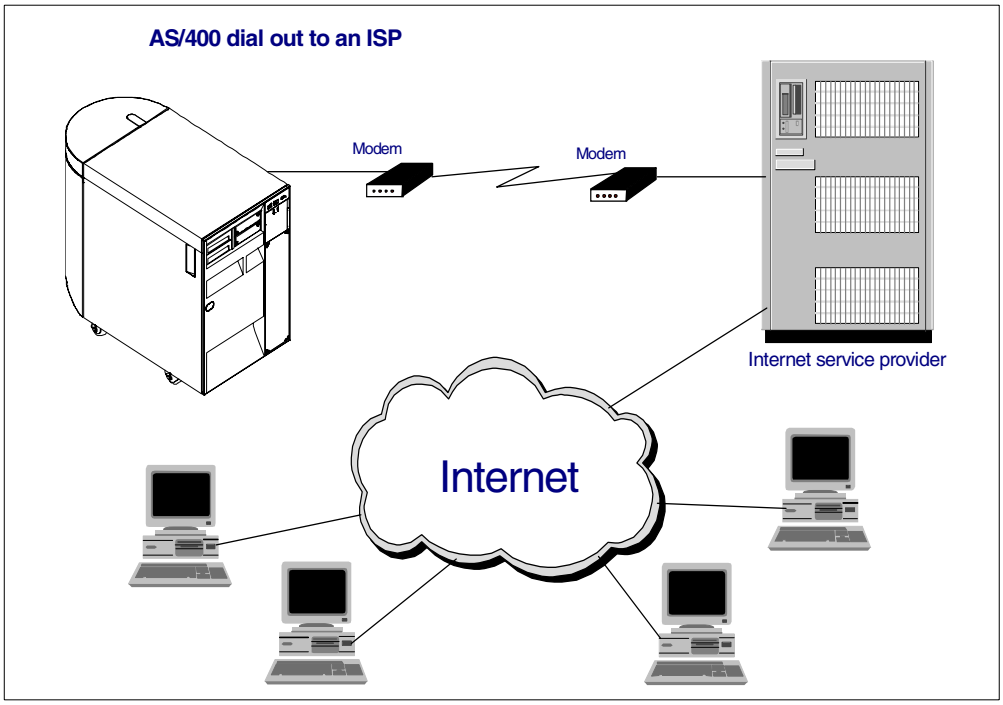


Figure 89. Connecting to an ISP using PPP protocol

The difference between a WAN and a LAN is that, on a WAN, the physical communication is normally limited to two systems. This is called a *point-to-point connection*. Several different protocols, such as PPP, SLIP, X.25 or Frame Relay, can be viewed as *point-to-point protocols*. The connection between the two systems can be either a direct connection, such as a leased telephone line, or they can be connected using a dial-up line, where the connection is established

when needed. In this case, the system can be setup either as a dial-out system (calling system), or as a dial-in system (answering system).

Figure 90 show a complex PPP network. The central AS/400 system has several PPP connections:

- To an Internet service provider (leased line)
- To a corporate network (leased line)
- To a remote office AS/400 (switched or leased line)
- To a remote server (could be an RS/6000 running AIX, or a UNIX system)
- To mobile clients, for example, used to dial in from an employee's home (switched line)

In this scenario, the clients on the Office and Remote LAN could have access to both the corporate network and the Internet.

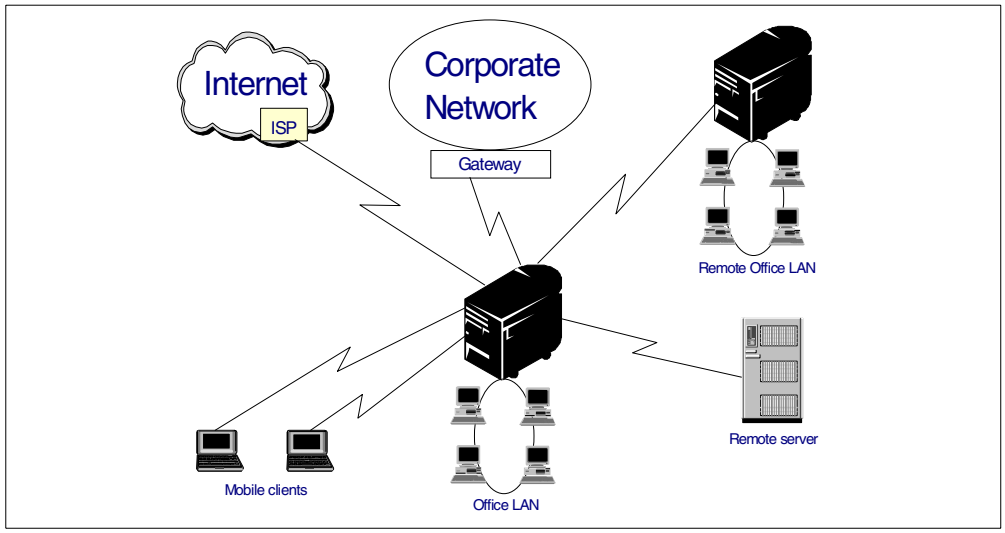


Figure 90. Complex PPP network

OS/400 V4R3 introduces the Dial-on-Demand functionality (Figure 91 and Figure 92). This enables the use of ad hoc connections to several destinations.

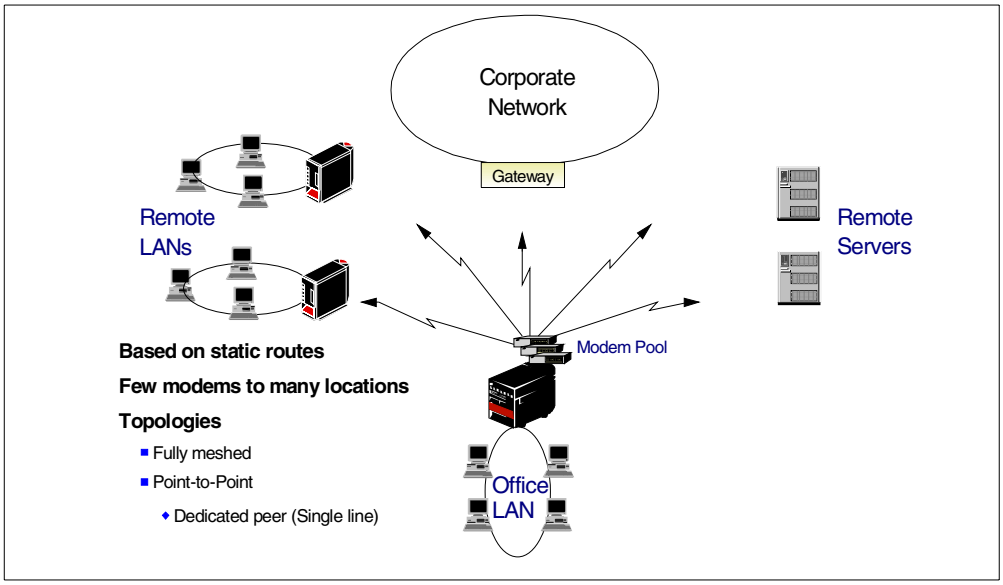


Figure 91. OS/400 TCP/IP PPP dial-on-demand

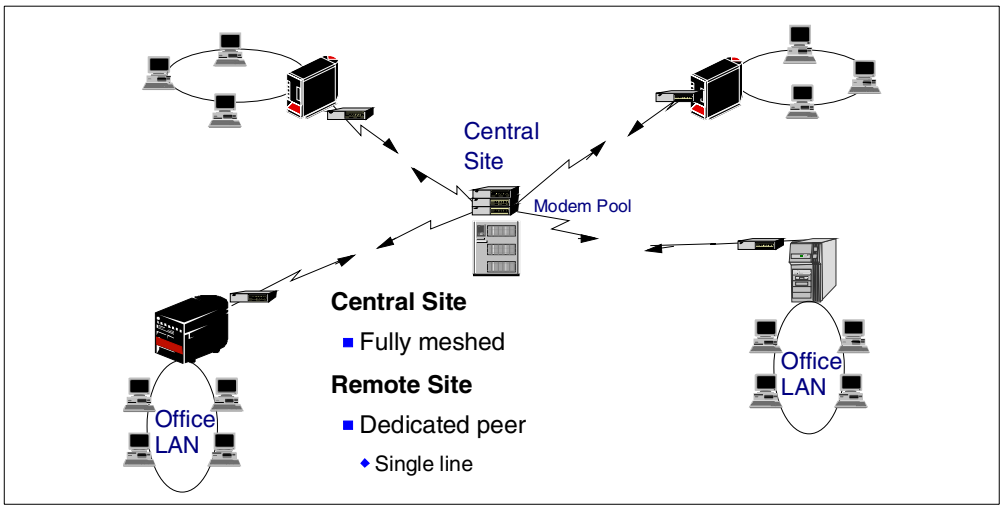


Figure 92. OS/400 PPP dial-on-demand hub and spoke

4.2.2 Advantages of PPP over SLIP

The SLIP Request for Comment (RFC) never became an Internet standard because it has several deficiencies:

- There is no standard way to define IP addressing between the two hosts, meaning that an unnumbered net cannot be used. We see later what an unnumbered net is.
- Only TCP/IP protocol was supported in SLIP. PPP supports several protocols, such as IPX and NetBIOS.
- SLIP has no support for system authentication, while PPP has two-way authentication.

- There is no support for error detection or compression. This is implemented in PPP.
- SLIP uses ICF for communication. This is eliminated in PPP, giving a better performance.

SLIP is still used today, and is still supported on the AS/400 system. However, IBM recommends that you use PPP for new setups. PPP is an Internet standard. It is more secure because of the better authentication used. It also performs better because of the compression facilities and no ICF.

4.2.2.1 PPP security options

The PPP protocol defines two types of authentication that can be used by the peers to identify each other:

- Password Authentication Protocol (PAP)
- Challenge Handshake Authentication Protocol (CHAP)

They both perform authentication of the remote devices, but PAP only provides a basic authentication level.

The PAP, shown in Figure 93, provides a simple method for the peer system to establish its identity using a two-way handshake. This is done only upon initial link establishment. After the link establishment is complete, an ID and PASSWORD pair is sent repeatedly by the peer to the authenticator until authentication is acknowledged or the connection is terminated. The PAP is not a strong authentication method. Passwords are sent over the line “in the clear”, making them easy to trace.

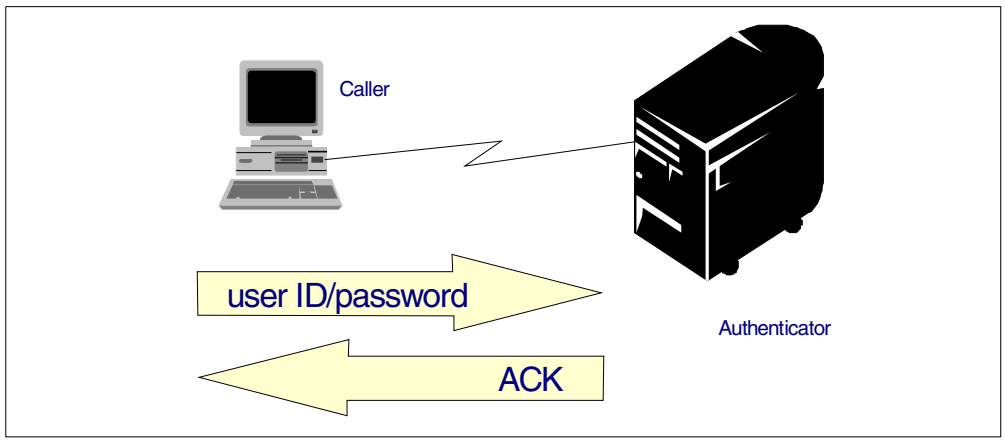


Figure 93. Password Authentication Protocol (PAP)

The CHAP, shown in Figure 94, is more secure when using a calculated value with an algorithm that is known only to the authenticator and the remote access device. The CHAP information is never sent over the link and is highly effective against playback and trial and error attempts.

Instead, the authenticator sends a “challenge” to the remote unit attempting to connect to the network. The remote unit responds with a value calculated by a common algorithm used by both devices. The authenticator checks the response against its own calculation of the expected value. If the values match, the authentication is acknowledged. Otherwise, the connection is terminated. The

PAP is a one-time authentication, while the CHAP can occur more than once during the connection.

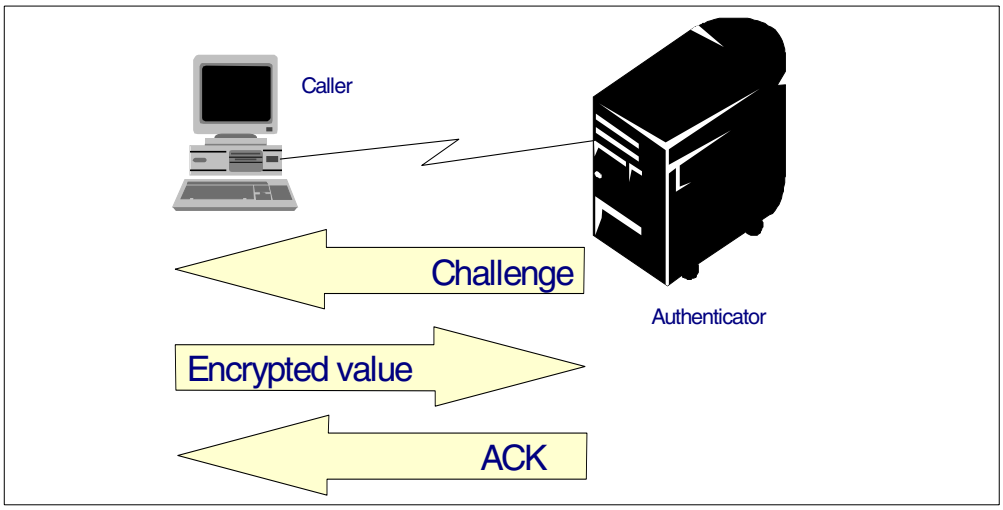


Figure 94. Challenge Handshake Authentication Protocol (CHAP)

4.2.3 IP address assignment

There are several concepts that must be understood before planning the PPP network. They are:

- **Numbered network:**

  Normally, all the interfaces connected to a network will have an IP address. If the network is a point-to-point network, each end can explicitly be given an IP address so that the PPP connection forms a separate network (Figure 95). This is called a *numbered network*.

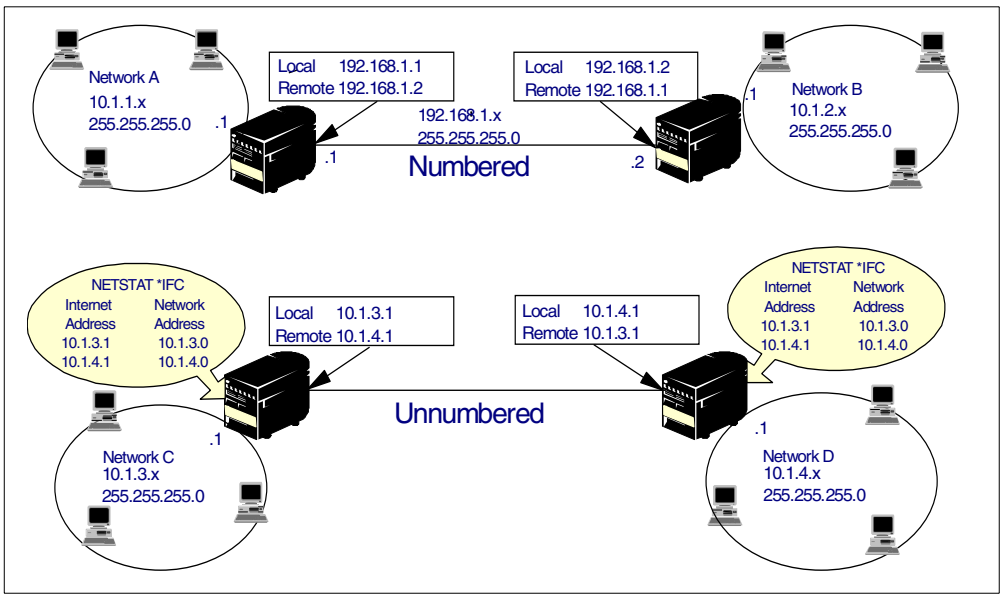


Figure 95. Unnumbered versus numbered networks

If a host on the LAN in one network wants to communicate with a host in the other network, you need to have routes defined. In our example, the AS/400 system in network A must have a route defined that points to 192.168.1.2 as the route to network B if the AS/400 system wants to communicate with a host in network B. However, there is no need to waste addresses for this because we can set the connection up as an unnumbered network.

- **Unnumbered network**:

  When two hosts are connected via an unnumbered PPP network, each end of the network does not have its own IP address, but appears to have the address of the interface to the LAN. The AS/400 PPP connection can be configured to create an IP interface whose address will be the main LAN interface address of the remote end. When we use the Work with TCP/IP Status (NETSTAT) command to display the interface status (`NETSTAT OPTION(*IFC)`), the address is seen as a normal interface. This allows the hosts on one side of the network to “talk” to the other side of the network by turning on the IP forwarding option in the Configure TCP/IP Attributes (CFGTCPA) command. The host in each network must have a route added that points to the AS/400 TCP/IP interface. For example, a host in network C (Figure 95 on page 87) must have a route added that points to the address 10.1.3.1 as the gateway to network D (10.1.4.x).

- **Transparent subnetworking**:

  Transparent subnetworking allows physically separated subnets to appear as if they are part of the same subnet (Figure 96). With this concept, a router is not required to connect the two networks.

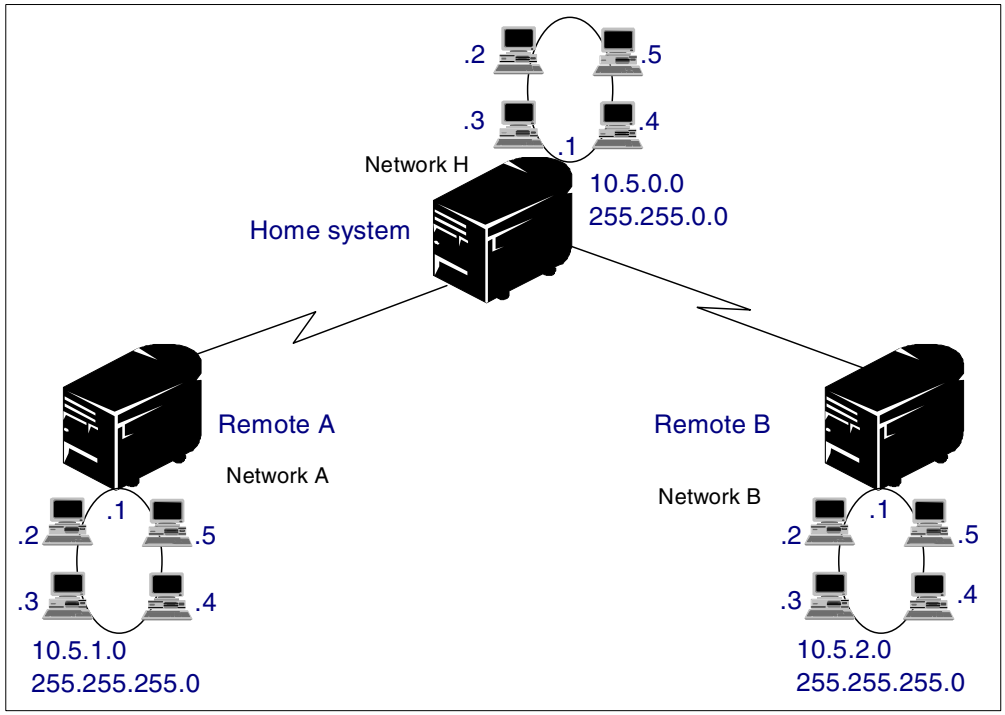


Figure 96. Transparent subnetting

To accomplish the transparent subnetworking from the home system network, you must divide the network into blocks of addresses for each remote site. In

our example, remote A has been given block 1, (10.5.1.0), and remote B has been given block 2 (10.5.2.0). Note that the subnet mask for remote A and B is 255.255.255.0. The subnet and the host are the two levels of hierarchical addressing. The boundary between them is arbitrary. In the addressing scheme used, there can be 255 partitions, each having 255 hosts available. It is then possible to manage which partitions will be assigned to remote networks. Each remote partition will be mapped to the subnet. The AS/400 system will hide the fact that the remote networks are physically located behind the AS/400 system. It will perform the Proxy ARP on behalf of the transparent subnet and forward packets to the remote networks. To the clients on the 10.5.0.0 network, it will appear that the remote clients are on the same physical network. Transparent subnetting is useful in environments where it is either impractical or impossible to run routing protocols. Such a situation may occur in a bridged network or in networks that are running unsupported proprietary dynamic routing protocols. This is best suited for point-to-point line connections when remote hosts need to communicate with one another, such as PPP.

4.3 PPP implementation on the AS/400 system

There are two protocols that can be used to connect two hosts that are running TCP/IP over, for example, a telephone line: SLIP and PPP. SLIP was the result of early attempts to connect two systems over an asynchronous line. Since it has some deficiencies, which we will see later, it is not widely used anymore.

SLIP has been supported since V3R1 of the AS/400 operating system. It can still be used on V4R2, but PPP is preferred for new setups. Support for PPP is included in OS/400 V4R3, 5769TC1, as part of WAN connectivity.

4.3.1 Line types

There are two line description types that can be used to create TCP/IP connection profiles. These are the older asynchronous line type, and the new PPP line type. The asynchronous line type can only be used for SLIP connections. The new PPP line type can be used for *both* SLIP and PPP connections. The preferred way to create the PPP line type is by using the Operations Navigator interface during creation of the connection profile.

The following green-screen commands can also be used to work with the new PPP connection profiles:

- Configure Point-to-Point TCPIP (CFGTCPPTP)
- Work with Point-to-Point TCPIP (WRKTCPPTP)
- End Point-to-Point TCP/IP (ENDTCPPTP)
- Start Point-to-Point TCP/IP (STRTCPPTP)

These commands are typically used to create and work with SLIP profiles that use the older asynchronous line type. For a PPP line type, they have limited use. You can activate or deactivate the connection, and you can display the status by using the Work with Point-to-Point TCPIP (WRKTCPPTP) command. You cannot access the configuration profiles for PPP line type. This can only be done by the Operations Navigator GUI interface.

The new PPP line type has the following advantages over the asynchronous line type:

- Support for PPP connections
- Support for advanced IP routing
- Support for both synchronous and asynchronous lines
- Support for V.25bis
- Support for Integrated Services Digital Network (ISDN) terminal adapters like the Courier I modem

4.3.2 Hardware requirements

The new PPP line type is only supported with the following IOA types. These IOAs can be used for SLIP as well:

- 2699 – Two-line WAN IOA
- 2720 – PCI WAN/Twinaxial IOA
- 2721 – PCI Two-line WAN IOA

The Asynchronous line type (SLIP) can be used on:

- 2609 – Two-line EIA 232/V24 adapter
- 2612 – One line EIA 232/V24 adapter

These two adapters do not support the new PPP line type.

4.3.3 Starting Operations Navigator for PPP configuration

From V4R2, the Operations Navigator is the interface used to configure PPP. It is not possible to use the green-screen commands to configure PPP profiles. However, the green-screen interface can still be used to start and stop the PPP connections.

The first step in configuring any part of PPP is to get to the correct portion of Operations Navigator. Refer to Figure 97 as you run the following procedure to get the PPP portion of Operations Navigator:

1. Start Operations Navigator by clicking **Start->Programs->IBM AS400 Client Access->AS/400 Operations Navigator**. The AS/400 Operations Navigator window appears (Figure 97).

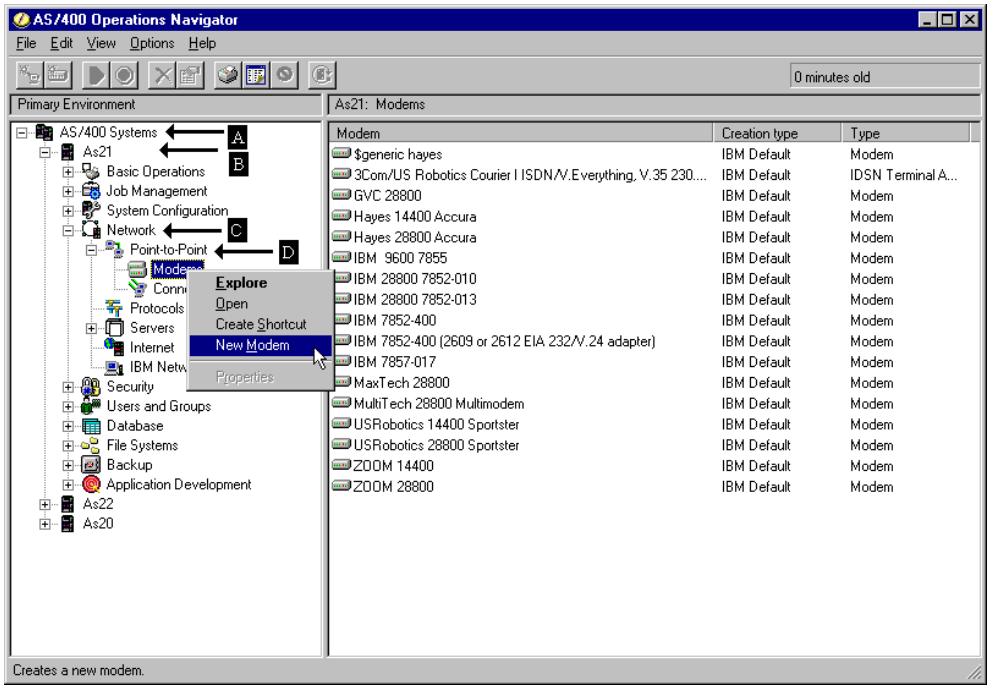


Figure 97. Accessing a PPP configuration using Operations Navigator

2. Double-click the AS/400 Network icon (A).
3. Double-click the system icon (B) for the AS/400 system that you are configuring. The system components appear.
4. Double-click the **Network** icon (C). The network components appear.
5. Double-click the **Point-to-Point** icon (D).

You are now ready to proceed with your PPP configuration procedure.

4.3.4 Defining modem types

Today's asynchronous modems use a command set invented by the Hayes company. The modems are sold as “Hayes compatible”. However, the initial Hayes command set has been extended by various vendors, and those extensions are often proprietary to the modem model. The command set used in a modem is normally described in the modem documentation. The commands are used to reset and initialize the modem, and to tell the modem to dial the phone number of the remote system. Because different modem models have different initialization command strings, each modem model has to be defined before it can be used with PPP connection profile.

The AS/400 system has many modem models predefined, but new models can be defined using the Operations Navigator Interface. An existing definition can be used as a base for the new type to be defined. If you are not sure what commands your modem is using, or you do not have access to the modem documentation, start with the Generic Hayes modem definition. The predefined shipped definitions cannot be changed. However, additional commands can be added to the existing initialization command or dial string.

Use the following steps to configure a modem definition:

1. Use the procedure in 4.3.3, “Starting Operations Navigator for PPP configuration” on page 90, to access the PPP configuration tree.
2. Click the **Modems** item. The available modems appear in the right window.
3. Right-click **Modems** to show the menu. Select **New Modem**. The New Modem Properties window appears. The values entered in Figure 98 are only an example.

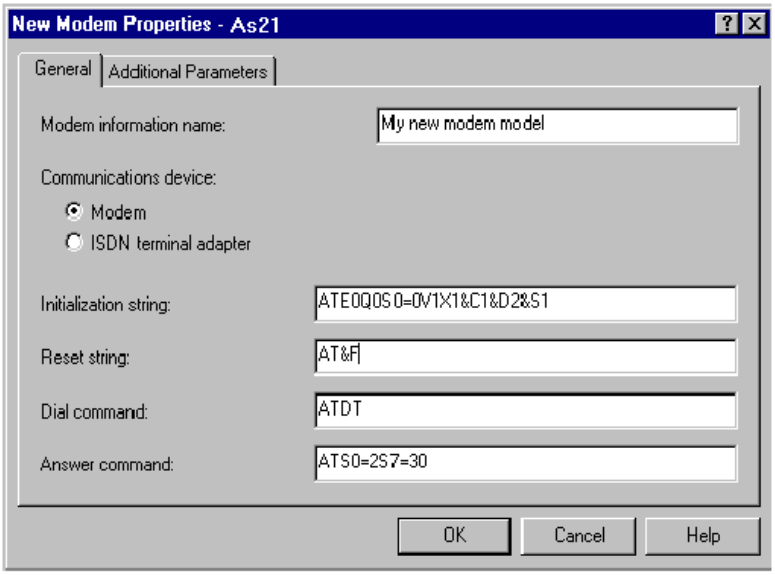


Figure 98. Creating a new modem type: Modem properties

4. In the Modem information name field, enter the name of your modem.
5. Check that *modem* is selected as Communications device.
6. In the Initialization string field, enter the Initialization string for your modem. See the modem documentation.
7. In the Reset string field, enter the command string for resetting your modem. See the modem documentation.
8. In the Dial command field, enter the dial command string. Do not enter the phone number here as part of the dial string. It will be entered in the PPP profile.
9. In the Answer command field, enter the string of commands to set your modem in the answer mode.
10. Click **OK** to save the modem definition.

4.3.5 Overview of PPP configuration using Operations Navigator

The following procedure provides an overview of using Operations Navigator. It is intended to explain the functions found in the windows by providing sample windows. The windows are not complete, and only give you an idea of the functions available. A detailed description of the functions is given in 4.6, “Scenario overview” on page 98. Refer to the correct scenario when you are ready to configure a connection.

Use the following steps to start configuration of a PPP connection:

1. Use the procedure in 4.3.3, “Starting Operations Navigator for PPP configuration” on page 90, to access the PPP configuration tree.
2. Click the **Connection Profiles** item. The available profiles appear in the right window.
3. Right-click **Connection Profiles** to show the menu. Select **New Profile**. The New Point-to-Point Profile Properties window appears. The values entered in Figure 99 are only an example.

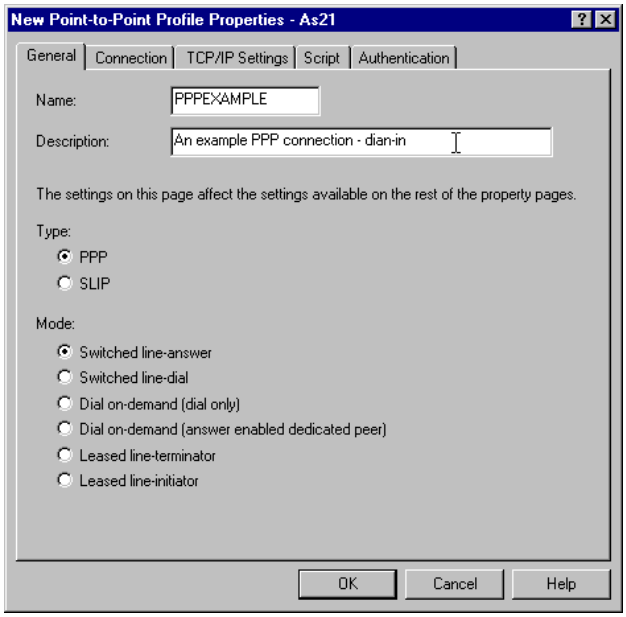


Figure 99. General properties on the PPP connection

4. Enter a name and description for the PPP profile. Select the type **PPP** and the mode **Switched line-answer**, or **Switched line-dial**, **Dial on-demand**, or **Leased line**. Click the **Connection** tab (Figure 100 on page 94).

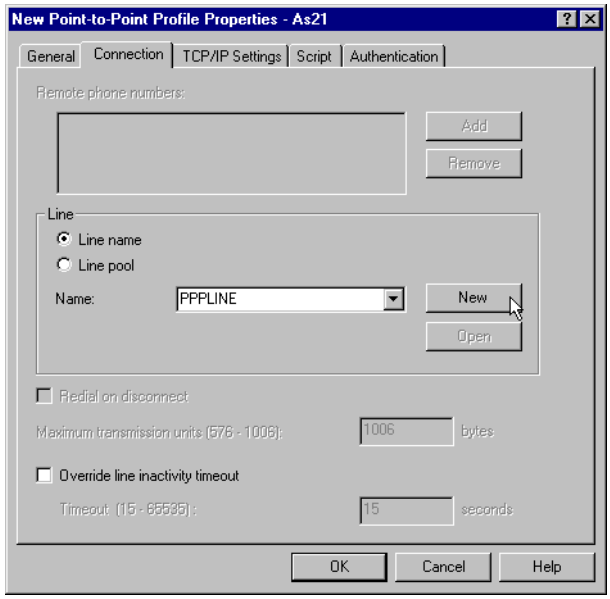


Figure 100. Connection properties on the PPP connection

5. Here you enter a remote connection number in case of a dial connection. You also specify what PPP line on the AS/400 system should be used for this connection profile. If none exists, enter the name, and click the **New** button to create a new PPP line on the AS/400 system. If you are using an existing PPP line on the AS/400 system, go directly to the TCP/IP settings. Figure 101 shows the creation of a new PPP line.

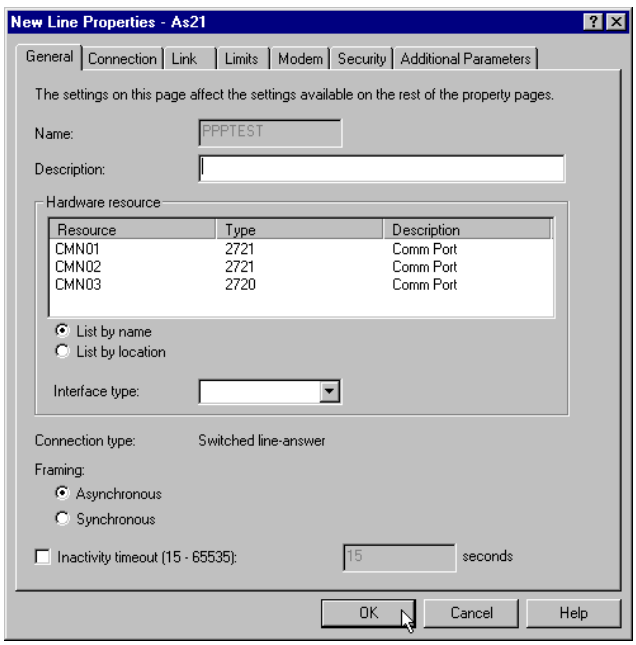


Figure 101. Creating a new PPP line on the AS/400 system

6. TCP/IP settings (Figure 102) is used to configure the TCP/IP settings on the local and the remote system. If a special configuration is required in relation to routing, either static or dynamic, click the **Routing** button.

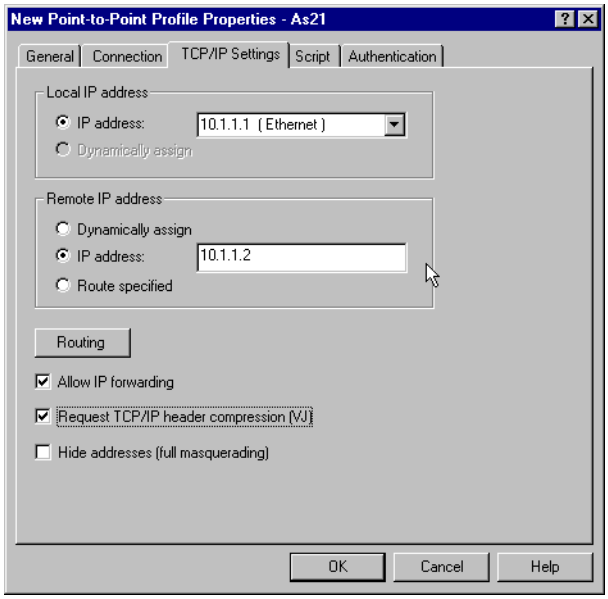


Figure 102. TCP/IP Settings on the PPP connection

7. The Script settings (Figure 103) are only relevant when configuring SLIP. Do *not* use scripting when configuring PPP.

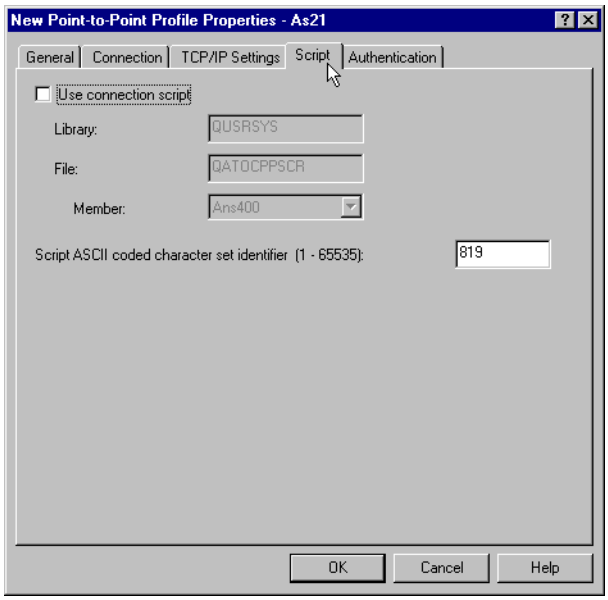


Figure 103. Script settings: Only for use when configuring SLIP

8. The **Authentication** tab (Figure 104 on page 96) allows the administrator to configure security related issues: remote and local and the kind of security that is required, such as PAP or CHAP.

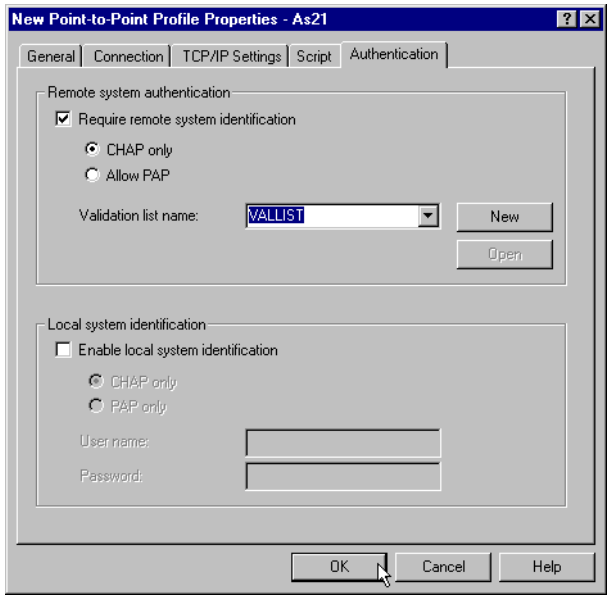


Figure 104. Authentication settings on the PPP connection

4.3.6 PPP jobs on the AS/400 system

Figure 105 shows an example of the PPP jobs running on the AS/400 system. In the QSYSWRK subsystem, the jobs *QTPANSnnn* or *QTPDIALnn* indicate a PPP connection profile that is ready.

```
                          Work with Active Jobs                    AS8
                                                        11/20/98  13:27:22
CPU %:     .9     Elapsed time:   00:00:09     Active jobs:   129

Type options, press Enter.
  2=Change   3=Hold   4=End   5=Work with   6=Release   7=Display message
  8=Work with spooled files   13=Disconnect ...

Opt  Subsystem/Job  User        Type  CPU %  Function        Status
       QTODDHCPS    QTCP        BCH      .0  PGM-QTODDSVR     SELW
       QTPOP00151   QTCP        BCH      .0                   DEQW
       QTPOP00165   QTCP        BCH      .0                   DEQW
       QTPOP00235   QTCP        BCH      .0                   TIMW
       QTPPANS008   QTCP        BCH      .0  PGM-QTOCPPPM     DEQW
       QTSMTPBRCL   QTCP        BCH      .0  PGM-QIMSBRDG     DEQW
       QTSMTPBRSR   QTCP        BCH      .0  PGM-QIMSBRSR     DEQW
       QTSMTPCLNT   QTCP        BCH      .0  PGM-QIMSCLCP     DEQW
       QTSMTPSRVR   QTCP        BCH      .0  PGM-QIMSSRCP     SELW
                                                                More...
Parameters or command
===>
F3=Exit   F5=Refresh      F7=Find      F10=Restart statistics
F11=Display elapsed data  F12=Cancel   F23=More options   F24=More keys
```

Figure 105. The PPP jobs in the QSYSWRK subsystem

4.4 Serial Line Interface Protocol (SLIP)

Serial Line Interface Protocol (SLIP) is the result of early attempts to connect two systems using TCP/IP over an asynchronous line. SLIP is not an Internet standard.

SLIP was developed in 1984 by Rick Adams for Berkley UNIX Version 4.2. It became a de facto standard for running TCP/IP over point-to-point serial connections. With the advent of high-speed dial-up modems, the interest in serial line communications grew dramatically, for example, for connecting home computers to the Internet through the RS232 serial port. Thus, it became necessary to develop a standard physical layer protocol for serial lines.

SLIP is a very simple protocol that merely defines a framing method for IP packets flowing over a serial line. It is described in RFC 1055 as a *Nonstandard for Transmission of IP Datagrams Over Serial Lines*, which suggests that it is not an Internet standard. SLIP has some major deficiencies in the following areas that are also described in the RFC:

- Addressing: Both computers in a SLIP link need to know each other's IP addresses for routing purposes. SLIP currently provides no mechanism for hosts to communicate addressing information over a SLIP connection.
- Packet type identification: Only one protocol can be run over a SLIP connection. While SLIP is a “Serial Line IP”, if a serial line connects two computers, those computers should be able to use more than one protocol over the line if it is needed. SLIP does not allow them to do so.
- To contrast SLIP's inability to allow more than one protocol over the single serial line (for example, a LAN protocol such as Token-Ring), Token-Ring allows many protocols such as SNA, TCP/IP, IPX, and so on, to all physically share one adapter and coexist down at the physical wire. SLIP only allows IP packets to be carried across the serial line.
- Error detection and correction: Noisy telephone lines corrupt packets in transit. Because the line speed is relatively slow, retransmitting a packet is very expensive. Error detection is not absolutely necessary at the SLIP level because any IP application should detect damaged packets (IP header and UDP and TCP checksums should be sufficient). Because it takes so long to retransmit a packet that was corrupted by line noise, it is efficient if SLIP can provide some sort of simple error correction mechanism of its own.
- Compression: Because dial-in lines are relatively slow, packet compression results in large improvements in packet throughput.

SLIP never became an Internet standard because it has several deficiencies. Some of those deficiencies are:

- No standardized mechanism for hosts to communicate addressing information
- No support for network protocols other than TCP/IP
- No support for system authentication
- No support for packet error detection, error correction, or compression

Despite these deficiencies, SLIP is still used today. It is provided as part of OS/400. However, IBM does not encourage you to use SLIP. Instead, we suggest that you use PPP. PPP is a better choice because it provides many advantages

over SLIP. PPP is the predominant connection protocol used among Internet Service Providers (ISP) today.

4.5 PPP or SLIP: Which do you choose

There is no doubt that you should use PPP for *any* WAN connectivity. Due to the lack of a standard SLIP protocol, this will save yourself a lot of problems. You should only use the SLIP protocol if the partner you are trying to communicate with cannot use the PPP protocol. Furthermore, you are forced to use the SLIP protocol if you have the old hardware IOA, as described in 4.3.2, “Hardware requirements” on page 90.

4.6 Scenario overview

The diagrams of the scenarios in the following section use the Classless Inter-Domain Routing (CIDR) method of specifying a network mask. For example, the network 10.1.1.0 subnet mask 255.255.255.0 can also be specified as network 10.1.1.0/24. The 24 identifies the number of significant bits in the IP address to use as the network. The 255.255.255.0 subnet mask identifies that the three first octets of the mask should be used. Since each octet consists of 8 bits, this is equal to the 24 bits used in the CIDR method.

4.7 PPP scenarios

The following sections show examples of the PPP configuration. Generally, to be successful, you have to accomplish the following tasks:

- Determine which hardware resources are to be used for the lines in both ends.
- Determine which IP addresses are to be used in both ends and if you are using numbered or unnumbered networks.
- Determine how the routing is to be done: either static or dynamic using RIP
- Determine which type of authentication is to be used: PAP or CHAP. The CHAP protocol is more secure.

Note

The MF20711 PTF should be applied before configuring *any* PPP dial-on-demand connection profiles.

Assigning a dynamic IP address at both ends of a *dial-on-demand* is not supported, even if the Operations Navigator interface lets you specify it. The ISP has to assign a *fixed* IP address to your connection.

4.7.1 Scenario 1: AS/400 answer and Windows PC dial

This scenario shows a situation where a PC user located at a remote site wants to connect to the AS/400 system and the network (Figure 106). The AS23 system is configured as answer, and the Windows PC uses the Microsoft Dial-up Networking (DUN) to establish the PPP connection to the AS/400 system. This can be done using Windows 9x and Windows NT, as shown in this scenario. The

PC is assigned an IP address that the system administrator has especially reserved for dial-in connections (10.1.1.200). The PPP connection from the Windows PC is established when the users at the PC request it.

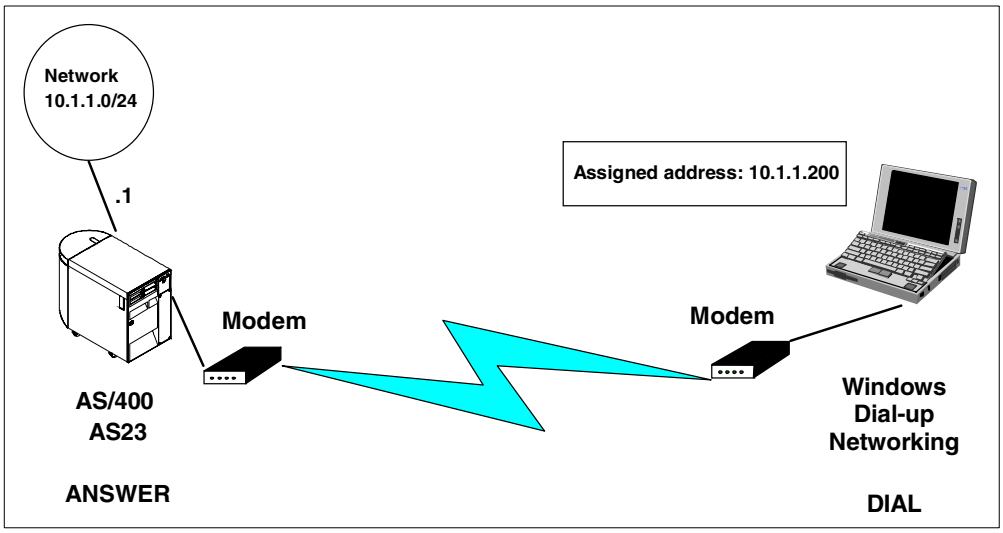


Figure 106. AS/400 AS23 answer and Windows PC dial using PPP

4.7.2 Scenario 2: AS/400 dial and AS/400 answer

This scenario shows the situation, where an AS/400 site wants to connect to another AS/400 site (Figure 107). This allows users on the network where AS23 is located to access resources on the network where AS08 is located. The AS23 system is configured as the dial and the AS08 system is configured as the answer. The scenario uses unnumbered networks on the WAN between the two AS/400 systems to save IP address space. The PPP connection from AS23 to AS08 must be established manually using the Operations Navigator. Refer to 4.7.3, “Scenario 3: AS/400 dial-on-demand to AS/400 answer” on page 100, for an example of the dial-on-demand function of the PPP implementation on the AS/400 system.

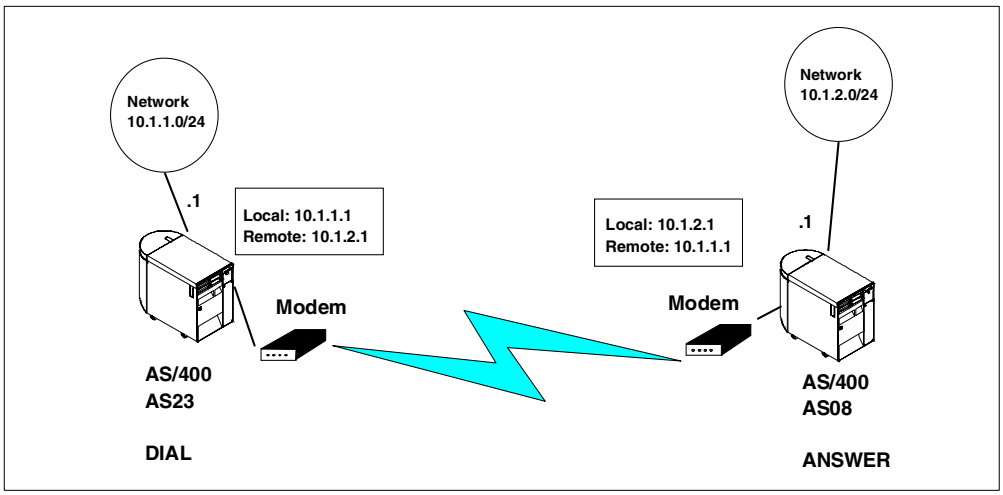


Figure 107. AS/400 AS23 dial and AS/400 AS08 answer using PPP

4.7.3 Scenario 3: AS/400 dial-on-demand to AS/400 answer

This scenario shows a situation where an AS/400 site wants to connect to another AS/400 site using the dial-on-demand feature (Figure 108). This allows users on the network where AS23 is located to access resources on the network where AS08 is located. The AS23 system is configured as the dial-on-demand and the AS08 system is configured as the answer. The scenario uses unnumbered networks on the WAN between the two AS/400 systems to save IP address space. The PPP connection from AS23 to AS08 is established automatically when required, for example, when a user from the AS23 system tries to access the AS08 system at the remote location.

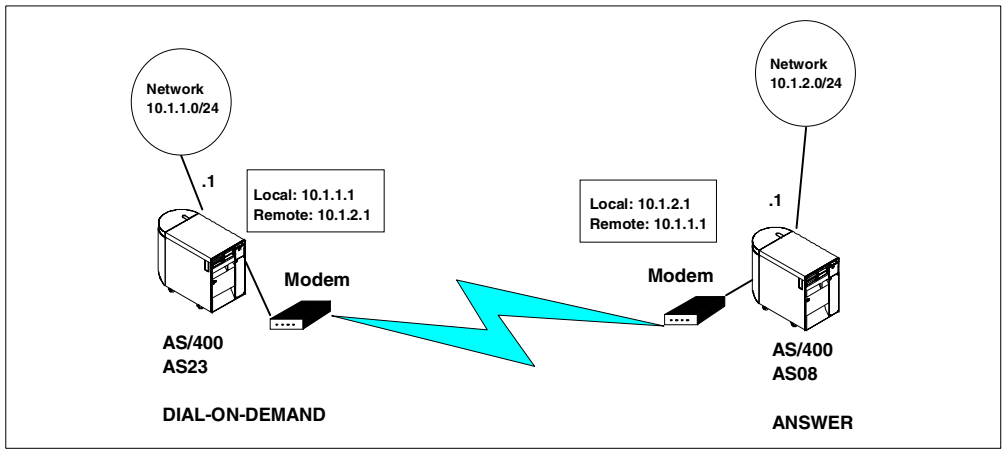


Figure 108. AS/400 AS23 dial-on-demand to AS/400 AS08 answer using PPP

4.7.4 Scenario 4: AS/400 dial to Windows NT answer

This scenario shows a situation where an AS/400 site wants to connect to network using the Windows NT Remote Access Service (RAS) (Figure 109). This allows users on the AS23 system to access resources on the network where the Windows NT system is located. The AS23 system is configured as the dial and the Windows NT system is configured as the answer. The PPP connection from AS23 to the Windows NT system is established automatically when required, for example, when a user from the AS23 system tries to access the Windows NT system at the remote location.

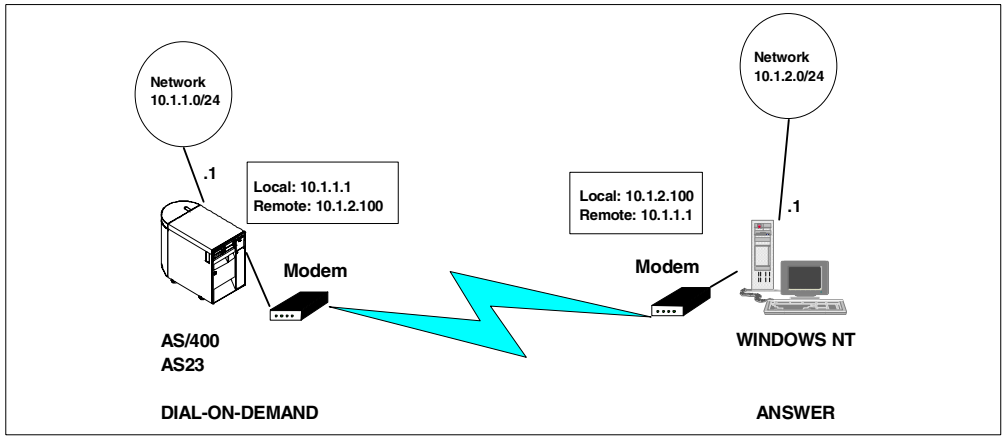


Figure 109. AS/400 AS23 dial-on-demand to Windows NT RAS answer using PPP

4.7.5 Scenario 5: AS/400 dial to an Internet Service Provider (ISP)

This scenario shows a situation where an AS/400 site wants to connect to the Internet using an Internet Service Provider (ISP) (Figure 110). This allows users on the AS20 system to access resources on the Internet. The PPP connection from AS20 to the Internet is established manually.

Note

This setup is not recommended if the AS/400 system is a production system. The AS/400 system should be protected by a firewall, either by using Firewall for AS/400 or by using IP filters on the AS/400 system.

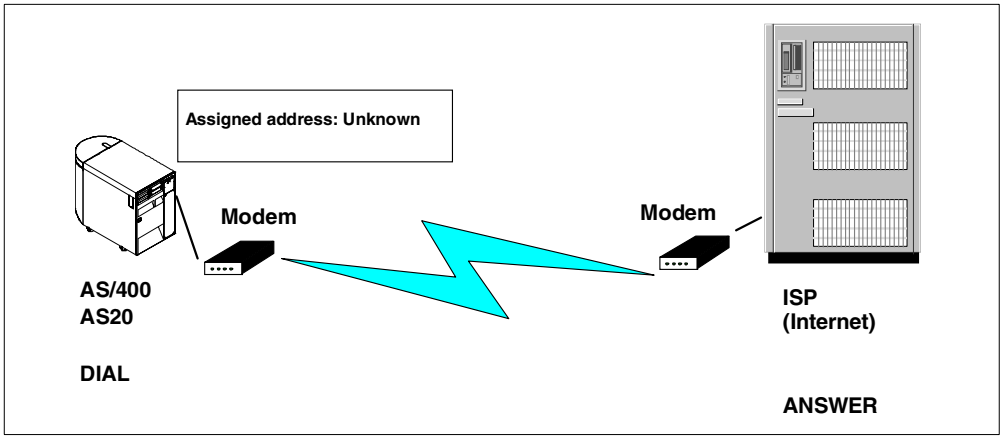


Figure 110. AS/400 dial-on-demand to the Internet (through an ISP) using PPP

4.8 SLIP scenarios

The following sections show the SLIP configuration examples we documented. We do not recommend the use of SLIP if PPP is available. We include these two scenarios for people running AS/400 systems that do not support PPP due to hardware or software restrictions.

Generally, to be successful, you have to accomplish the following tasks:

- Determine which hardware resources are to be used for the lines in both ends.
- Determine which IP addresses are to be used in both ends and if you are using numbered or unnumbered networks.

4.8.1 Scenario 6: AS/400 dial to AS/400 answer

This scenario shows a situation where an AS/400 site wants to connect to another AS/400 site (Figure 111 on page 102) using SLIP. This allows users on the network where AS23 is located to access resources on the network where AS08 is located. The AS23 system is configured as the dial and the AS08 system is configured as the answer. The SLIP connection from AS23 to AS08 must be established manually.

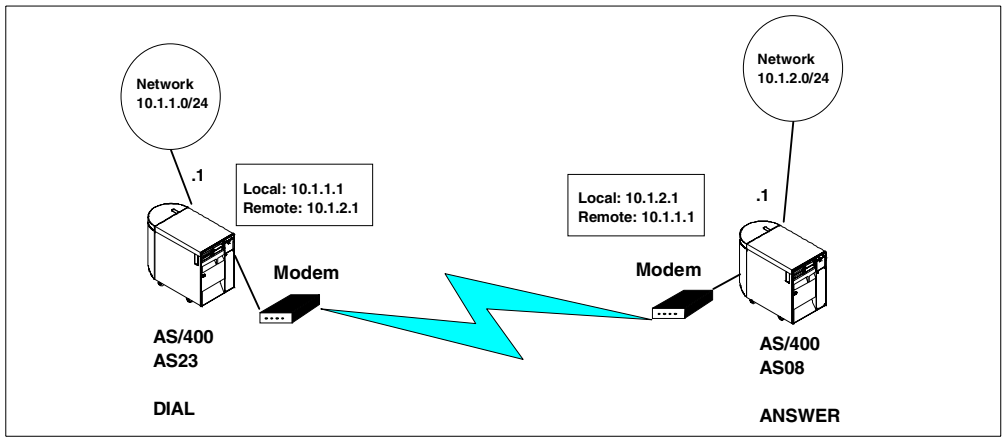


Figure 111. AS/400 dial to AS/400 answer using SLIP

4.8.2 Scenario 7: Windows 9x PC dial to AS/400 answer

This scenario shows a situation where a PC user located at a remote site wants to connect to the AS/400 system and the network (Figure 112). The AS23 system is configured as the answer and the Windows 9x PC uses the Microsoft DUN to establish the SLIP connection to the AS/400 system. The PC is assigned an IP address that the system administrator has specially reserved for dial-in connections (10.1.1.200). The SLIP connection from the Windows 9x PC is established when the users at the PC request it.

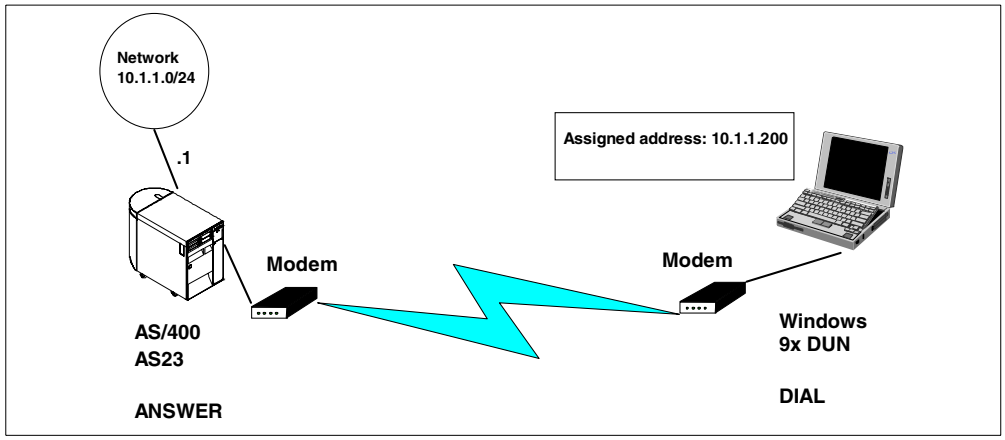


Figure 112. Windows 9x PC dial to AS/400 answer using SLIP

4.9 Scenario 1: AS/400 answer and Windows PC dial

In this scenario (Figure 106 on page 99), you perform the following tasks:

- Configure a PPP switched answer connection profile on AS23.
- Configure a PPP switched DUN connection on the Windows PC. The configuration is shown using both Windows 9x and Windows NT.

4.9.1 Configuring the AS/400 system PPP connection

Refer to Chapter 2, “TCP/IP basic installation and configuration” on page 7, to create the basic setup of the IP interfaces on the AS/400 systems. We used the IP address AS23: 10.1.1.1/24 on the 10.1.1.0/24 network on the AS/400 system.

Use the following steps to start the configuration of a PPP connection:

1. Use the procedure in 4.3.3, “Starting Operations Navigator for PPP configuration” on page 90, to access the PPP configuration tree.
2. Click the **Connection Profiles** item. The available profiles appear in the right window.
3. Right-click **Connection Profiles** to show the menu. Select **New Profile**. The New Point-to-Point Profile Properties window shown in Figure 113 appears.

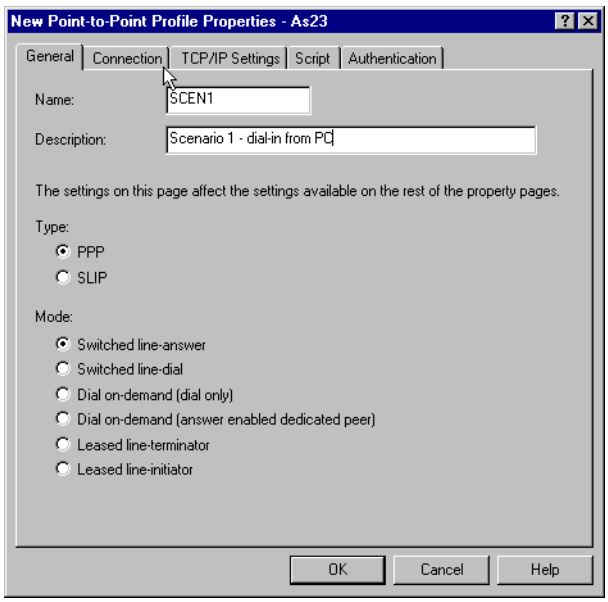


Figure 113. Selecting the PPP connection as switched answer

4. Enter a name for the profile and a description. Select a type of **PPP** and a mode of **Switched line-answer**. Click **Connection**. The display shown in Figure 114 on page 104 appears.

Figure 114. Creating a new PPP line for the connection

5. Type a line name in the Name field, and click **New** to create a new line for the connection. You may also select an existing line from the drop-down list. The window shown in Figure 115 appears.

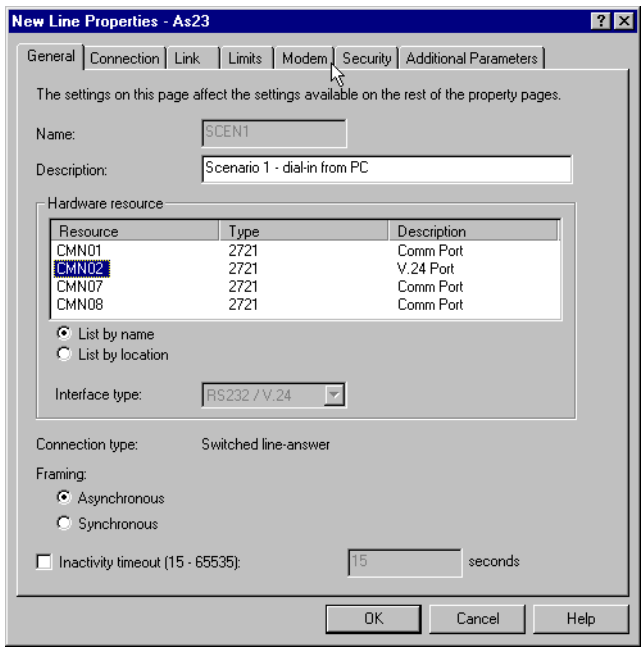


Figure 115. Selecting the correct hardware adapter to use

6. Select the appropriate hardware adapter. Click **Modem**. The display shown in Figure 116 appears.

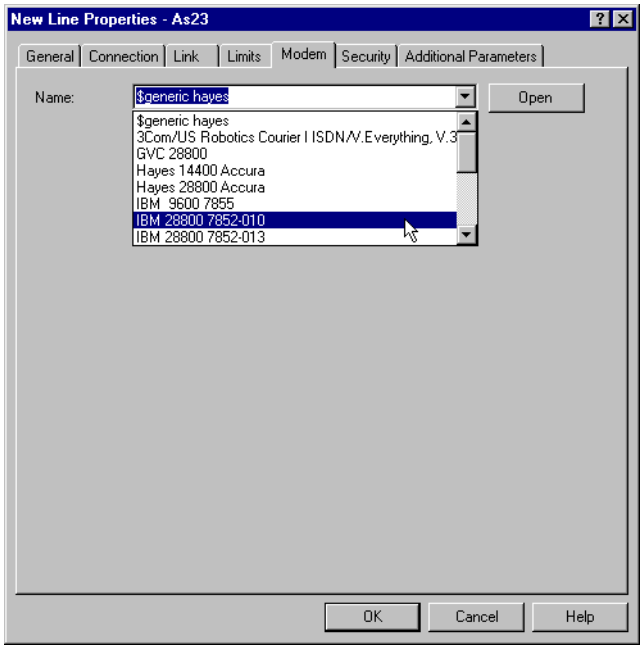


Figure 116. Selecting the modem type

7. Select the modem you are using from the list shown. Click **OK**. The display shown in Figure 114 appears. Click **TCP/IP Settings**. The display shown in Figure 117 appears.

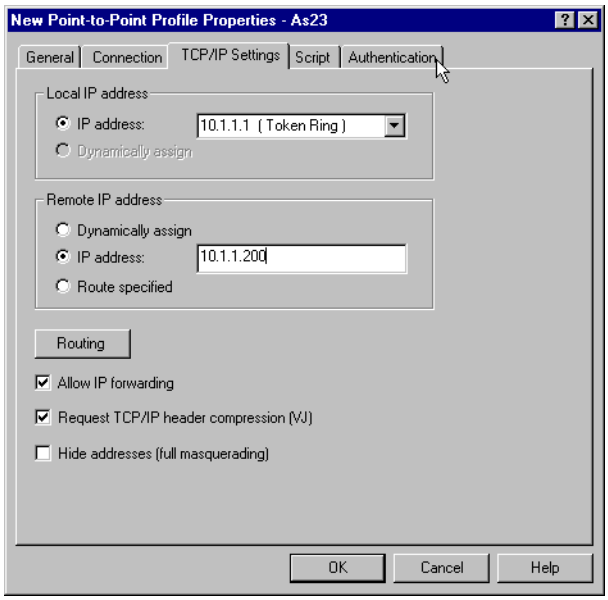


Figure 117. Configuring the TCP/IP settings for the local and the remote system

8. In this example, we set up an unnumbered network. We select **Allow IP forwarding** to enable the PC to communicate with all the systems in the 10.1.1 network. For the local IP address, we use the address of the AS/400 LAN adapter 10.1.1.1. For the remote address, we use an address that the network administrator set aside in the LAN for PPP. In this case, the address

is 10.1.1.200. After you enter the values, click **Authentication**. The display shown in Figure 118 appears.

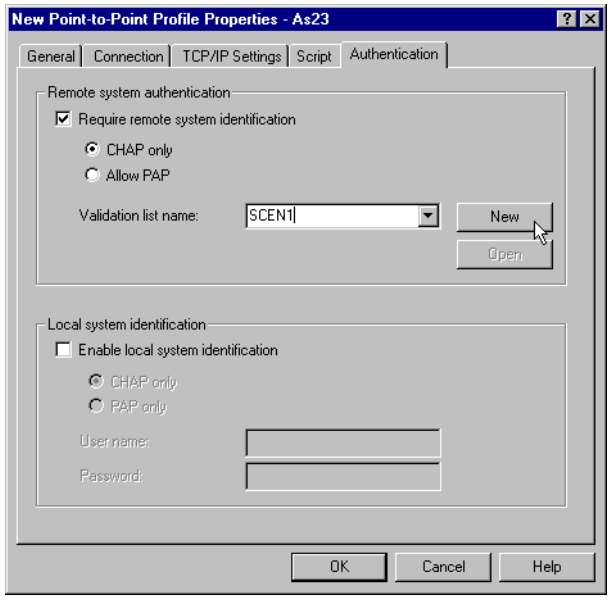


Figure 118. Creating a validation list for the remote client

9. Check **Require remote system identification**. Select the type **CHAP only** or **Allow PAP**. Enter a validation list name, and click **New** to create a new validation list for the remote users. You may use an existing validation list from the drop-down rather than creating a new list. The display shown in Figure 119 appears.

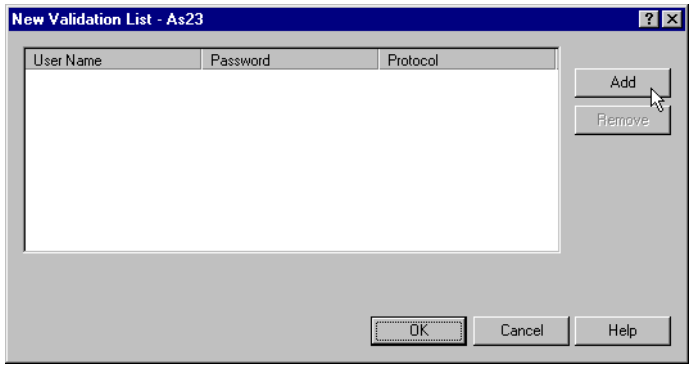


Figure 119. Adding a user to the validation list

10. Click **Add** to add a new user to the validation list. The display shown in Figure 120 appears.

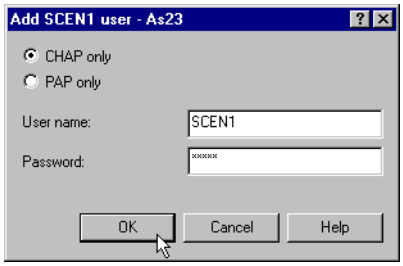


Figure 120. Specifying the user and password

11. Specify the user and the password. Click **OK**. The display shown in Figure 121 appears.

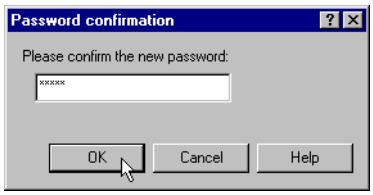


Figure 121. Confirming the password entered

12. Enter the password again to confirm the value entered. Click **OK**. The display shown in Figure 122 appears. Notice that the validation list has been added.

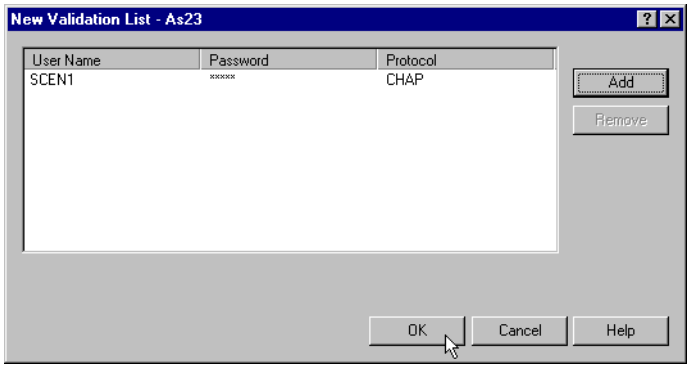


Figure 122. Confirming the creating of the validation list

13. Click the **OK** button to confirm the new validation list. The display shown in Figure 123 on page 108 appears.

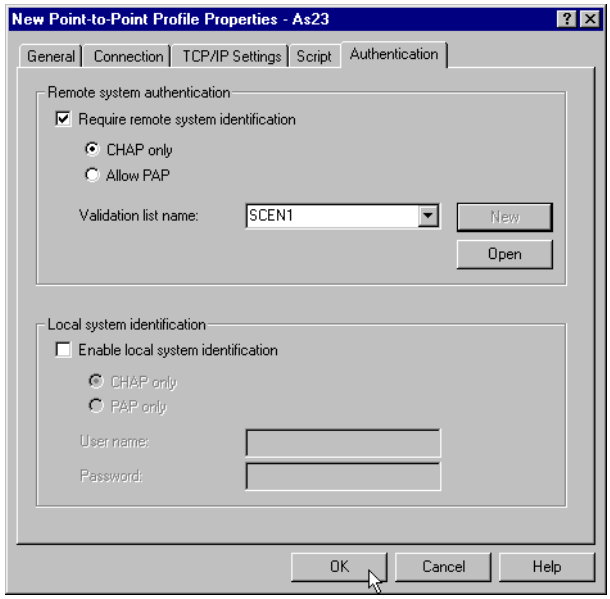


Figure 123. Confirming the creation of the PPP connection profile

14. Click **OK** to confirm the creation of the new profile. The display shown in Figure 124 appears. Notice that your profile has been added in the right-hand window.

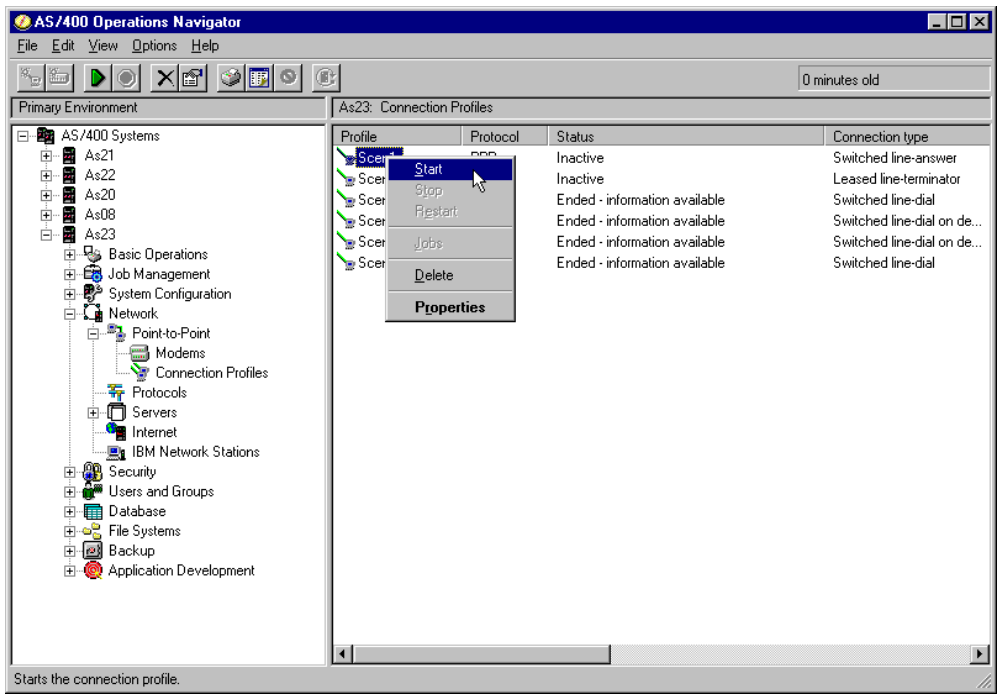


Figure 124. Starting the PPP connection profile on the AS23 system

15. To start the connection profile, right-click on the profile name. Select **Start** from the pop-up menu. The status changes to `Waiting for incoming call` as shown in Figure 125. You may need to refresh the window contents by pressing F5.

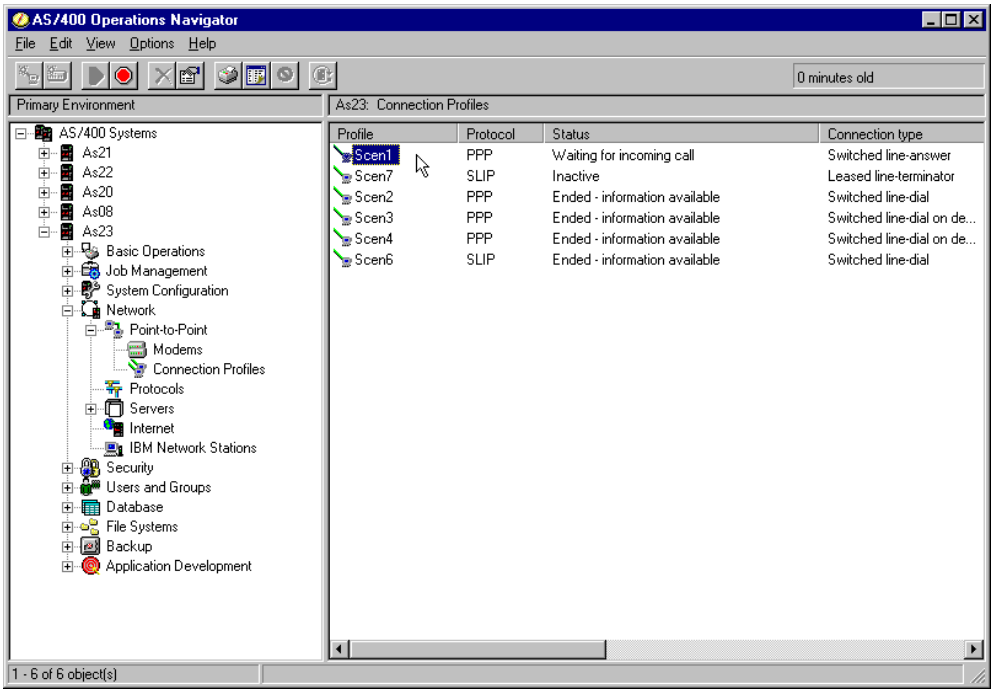


Figure 125. The ready-to-use PPP connection profile on the AS23 system

The AS/400 system is ready for the PPP connection.

4.9.2 Configuring the Windows 9x PPP connection

This section describes how to configure the PPP dial-up network connection on the Windows 9x system. Please refer to the Windows documentation for further information. If you are using Window NT, go to 4.9.3, “Configuring the Windows NT PPP connection” on page 118.

Note

During this procedure, you may need your Windows operating system media. Insert the media, and point to the drive when prompted. You may also be asked to restart your Windows system. Follow the directions on the screen and restart as requested.

4.9.2.1 Installing Windows 9x Dial-Up Networking

If the workstation does not have Dial-Up Networking support installed, you should follow this procedure. If the Dial-Up Networking support is installed, skip to 4.9.2.2, “Configuring a new Windows connection for PPP” on page 115.

Perform the following procedure to install Dial-Up Networking on the workstation:

1. Click **Start->Settings->Control Panel** to access the Windows Control Panel (Figure 126 on page 110).

Figure 126. Selecting the Network icon from the Control Panel

2. In the Control Panel, double-click the **Network** icon. The display shown in Figure 127 appears.

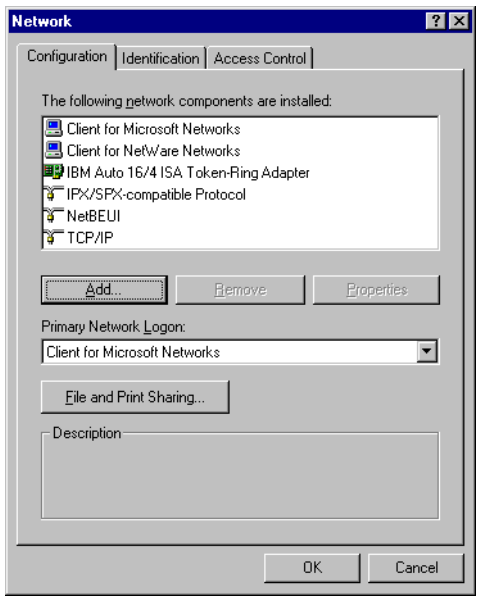


Figure 127. Using the Add button

3. Click **Add**. The display shown in Figure 128 appears.

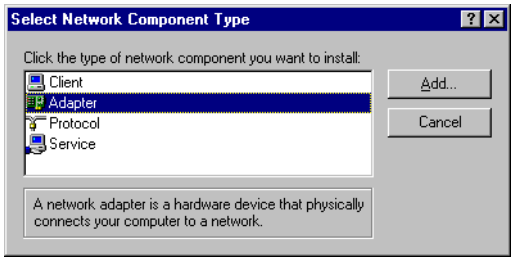


Figure 128. Adding a new adapter

4. Select the **Adapter** item, and click **Add**. The display shown in Figure 129 appears.

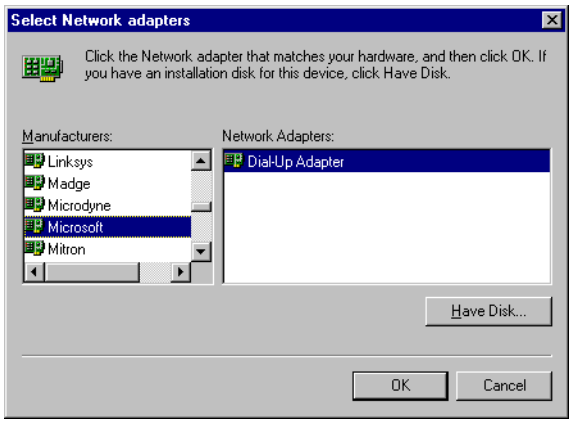


Figure 129. Selecting an adapter type

5. Select the **Microsoft** item and then the **Dial-Up Adapter** item. Click **OK**. You return to the Windows Control Panel (Figure 130).

Figure 130. Selecting the Modem icon from the Control Panel

6. Double-click the **Modem** icon in the Control Panel. The display shown in Figure 131 on page 112 appears.

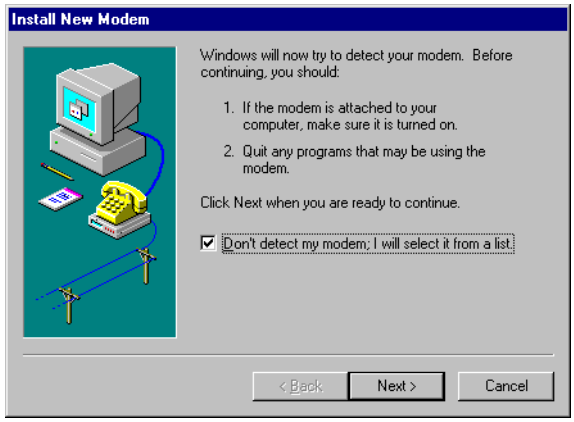


Figure 131. Installing a new modem

7. Select the checkbox, and click **Next**. The display shown in Figure 132 appears.

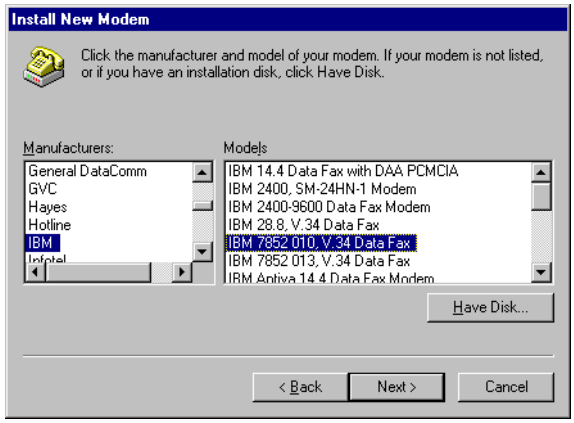


Figure 132. Selecting the correct modem type

8. Select your modem manufacturer and modem type from the list. In our example, we select IBM as the manufacturer and IBM 7852 as the modem type. Click **Next**. The display shown in Figure 133 appears.

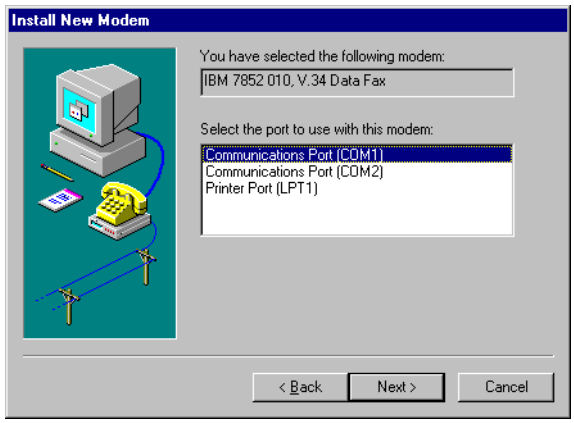


Figure 133. Selecting the communication ports

9. Select the communication port to use, and click **Next** (Figure 134).

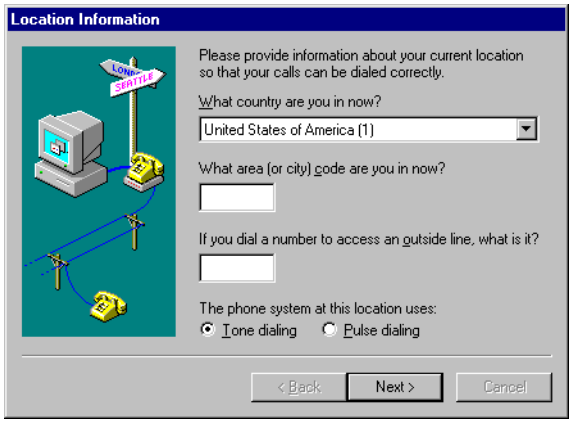


Figure 134. Entering the location information

10. Enter the location information, and click **Next**. The display shown in Figure 135 appears.

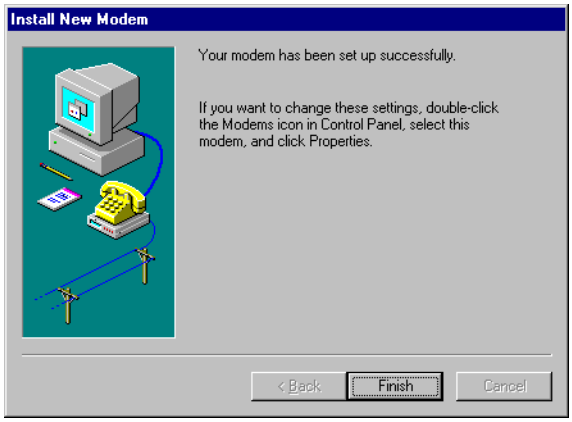


Figure 135. Completing the installation of the new modem

11. Click **Finish**. You return to the Windows Control Panel (Figure 136 on page 114).

Figure 136. Adding programs to the Windows setup

12. Double-click the **Add/Remove Programs** icon in the Control Panel. The display shown in Figure 137 appears.

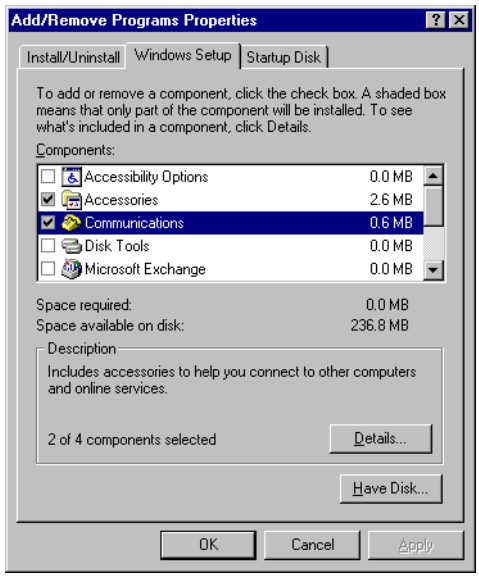


Figure 137. Adding functions to the communications

13. Select the **Communications** check box, and click **Details**. The display shown in Figure 138 appears.

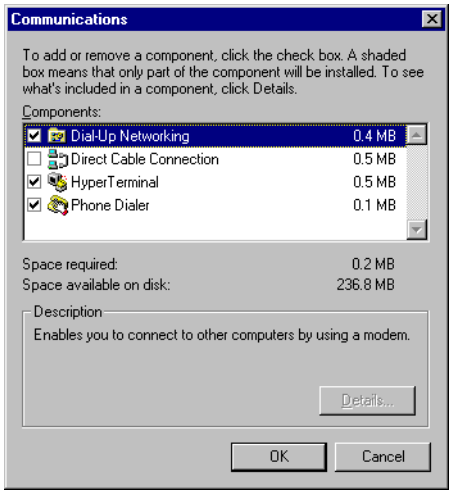


Figure 138. Installing Dial-Up Networking

14. Select the **Dial-Up Networking** item, and click the **OK** button. You may be prompted to insert your Windows CD or diskette. Follow the directions in the windows to complete the installation process. You may also be required to restart the workstation.

4.9.2.2 Configuring a new Windows connection for PPP

After Dial-Up Networking support is installed, you must configure a PPP connection to communicate with the AS/400 system. Refer to Figure 139 as a starting point while following this procedure to configure the connection.

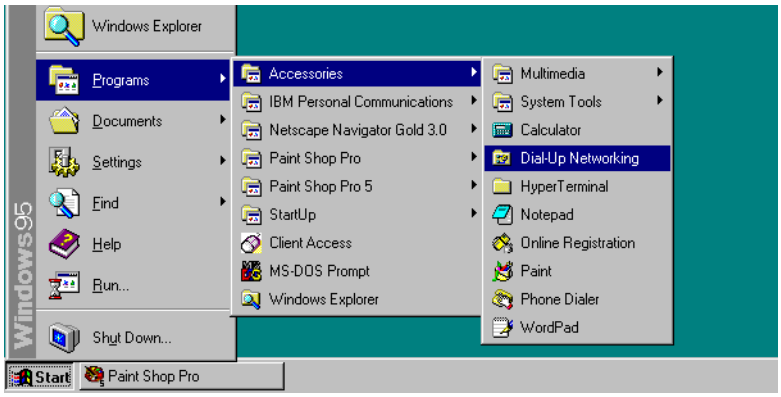


Figure 139. Starting the Dial-Up Networking

1. On the Windows menu bar, click **Start->Programs->Accessories->Dial-Up Networking** to start the Dial-Up Networking configuration. If this is the first configuration, the display shown in Figure 140 on page 116 appears. If there are existing configurations, the Dial-Up Networking window appears. If you receive the Dial-Up Networking window, double-click the **Make New Connection** icon. The display shown in Figure 141 on page 116 appears. Skip to step 3 on page 116.

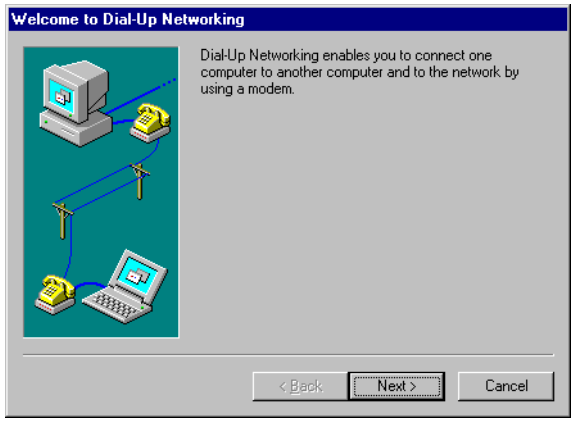


Figure 140. Starting a new dial-up configuration

2. Click **Next**. The display shown in Figure 141 appears.

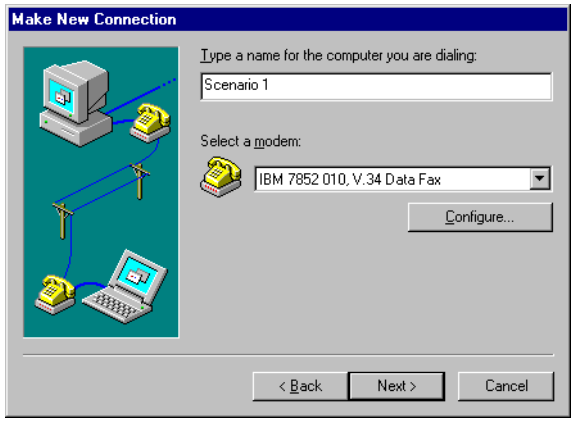


Figure 141. Specifying a name and a modem to use

3. Enter the name of the new connection, and select the modem. Click **Next**. The display shown in Figure 142 appears.

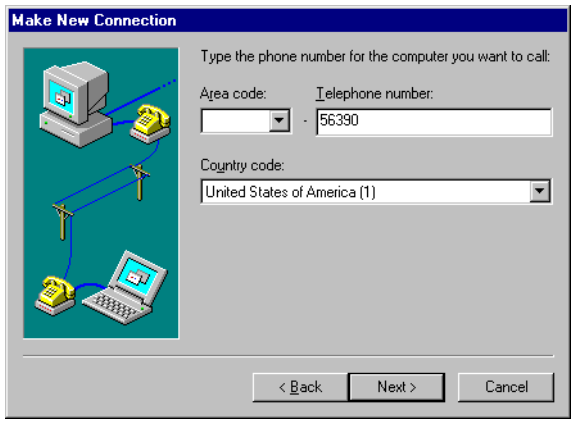


Figure 142. Specifying the phone number to use

4. Enter the phone number with which you are going to connect. Click **Next**. The display shown in Figure 143 appears.

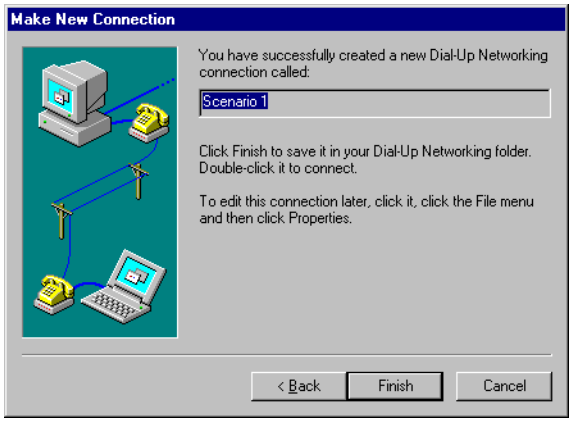


Figure 143. Successful installation of a new connection

5. Click the **Finish** button. The display shown in Figure 144 appears.

Figure 144. Modifying the properties of the connection

6. You now need to specify to which server type to connect. Right-click the icon added for your connection. Select **Properties** from the pop-up menu. A window similar to the one shown in Figure 145 appears. The contents of the window may be different depending on your level of Windows.

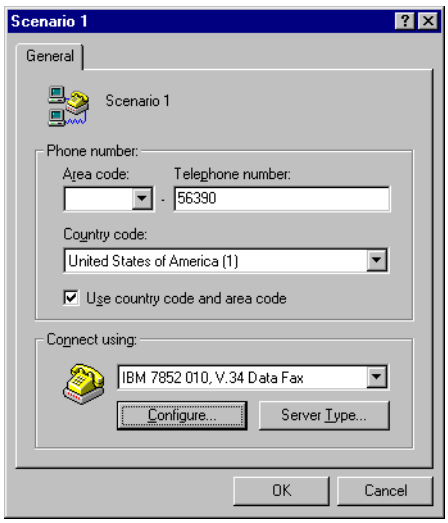


Figure 145. Specifying the server type

7. Click the **Server Type** button or tab. The display shown in Figure 146 on page 118 appears.

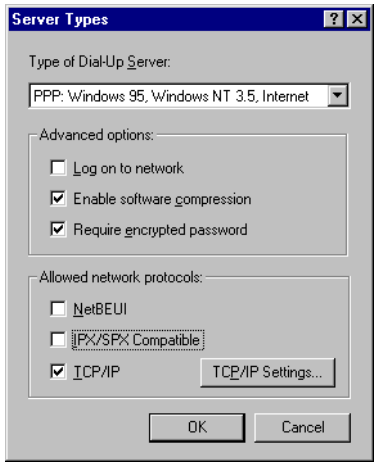


Figure 146. Setting the Server Type parameters

8. Specify the parameters shown in Figure 146, and click **TCP/IP Settings**. The display shown in Figure 147 appears.

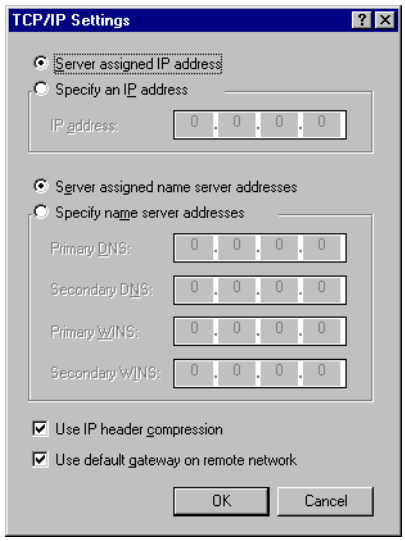


Figure 147. TCP/IP settings on the PC dial-up connection

9. Specify the parameters shown in Figure 147, and click **OK**.

This completes the Windows 9x configuration.

4.9.3 Configuring the Windows NT PPP connection

This section describes how to configure Dial-Up Networking on a Windows NT system. Remote Access Service (RAS) must be installed on the Windows NT system.

Note

During this procedure, you may need your Windows operating system media. Insert the media and point to the drive when prompted. You may also be asked to restart your Windows system. Follow the directions on the screen and restart as requested.

4.9.3.1 Configuring Remote Access Support (RAS)

Ensure that the Remote Access Service is installed and configured to allow dial-out connections by following these steps:

1. Click **Start->Settings->Control Panel** to access the Control Panel (Figure 148).

Figure 148. Selecting the Network icon on the Control Panel

2. Double-click the **Network** icon. The display shown in Figure 149 on page 120 appears.

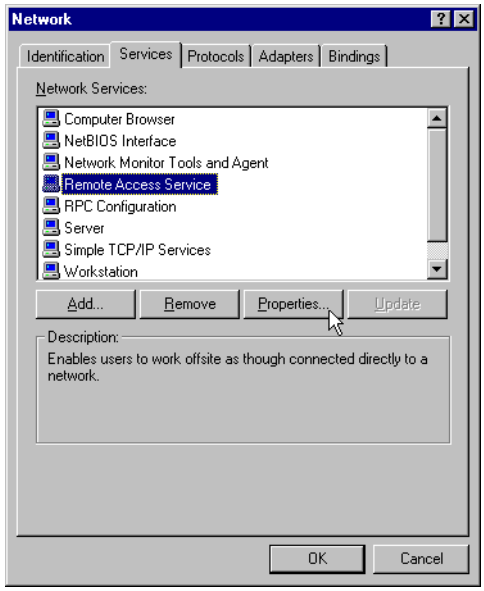


Figure 149. Modifying the properties of the Remote Access Service

3. Click **Services->Remote Access Service**. Click **Properties**. The display shown in Figure 150 appears. If Remote Access Service (RAS) does not appear in the list, you must install it. Refer to 4.12.3, “Installing Remote Access Service (RAS)” on page 150, for more information on installing the Remote Access Service.

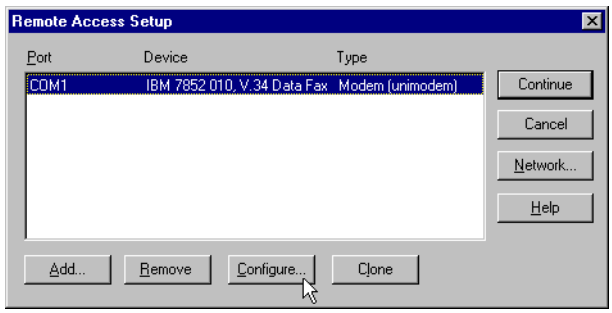


Figure 150. Changing the configuration of the Remote Access Service

4. Select the communications port to use from the list presented. Click **Configure**. The display shown in Figure 151 appears.

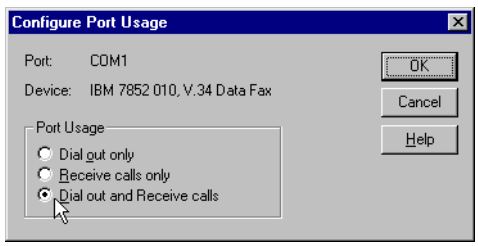


Figure 151. Selecting the RAS as both dial and answer

5. Select how you want this port to be used. In our example, we selected Dial out and Receive calls. This allows the system to start outgoing calls and answer

incoming PPP connection requests. If you want the system to only support outgoing calls, select **Dial out only**. If you want the system to only support incoming calls, select **Receive calls only**. After you make your selection, click **OK**. The display shown in Figure 150 appears.

6. Click **Network** to change the TCP/IP settings for this port. The display shown in Figure 152 appears.

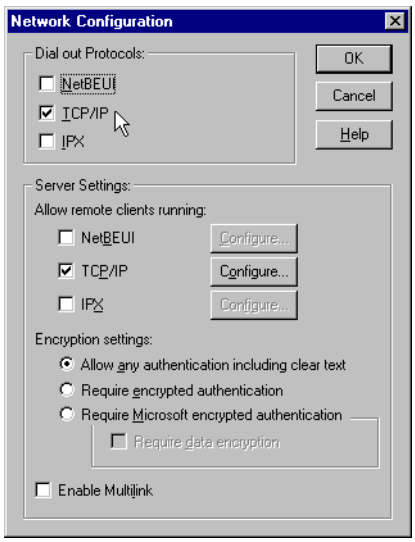


Figure 152. Selecting TCP/IP for dial-out

7. Select **TCP/IP** under Dial out Protocols and Server Settings. Click **OK**.

You have now configured the Remote Access Service support.

4.9.3.2 Configuring the Windows NT dial connection

To configure a dial connection, follow these steps:

1. Double-click **My Computer** on the desktop. The display shown in Figure 153 appears.

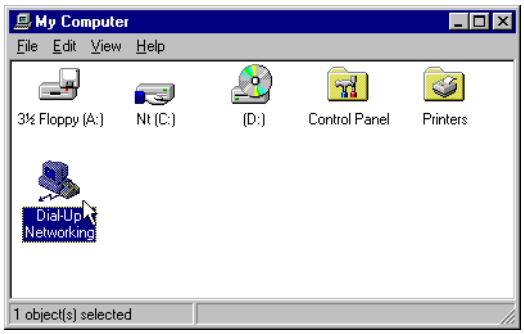


Figure 153. Selecting the Dial-up Networking icon

2. Double-click the **Dial-Up Networking** icon. If the phone book is empty, The display shown in Figure 154 on page 122 appears.

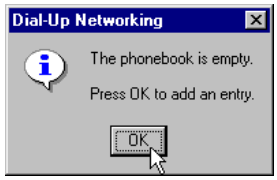


Figure 154. Adding an entry to the empty phonebook

3. Click **OK** to add a new entry to the empty phonebook. The display shown in Figure 155 appears.

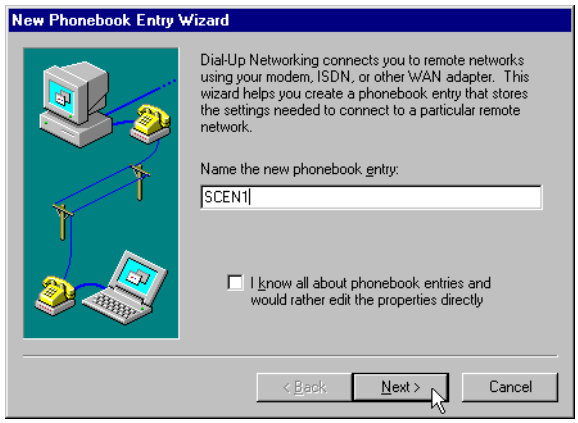


Figure 155. Supplying a name to the new connection

4. Supply a name, and click **Next**. The display shown in Figure 156 appears.

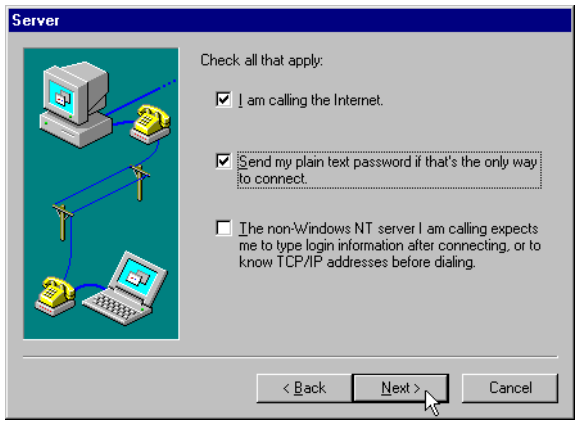


Figure 156. Configuring PPP settings for the new connection

5. Select **I am calling the Internet** and **Send my plain text password if that's the only way to connect**. Click **Next**. The display shown in Figure 157 appears.

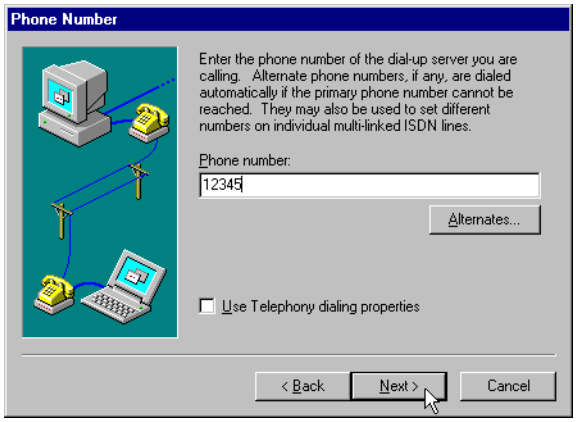


Figure 157. Entering the remote phone number

6. Enter the phone number of the remote location. Click **Next**. The display shown in Figure 158 appears.

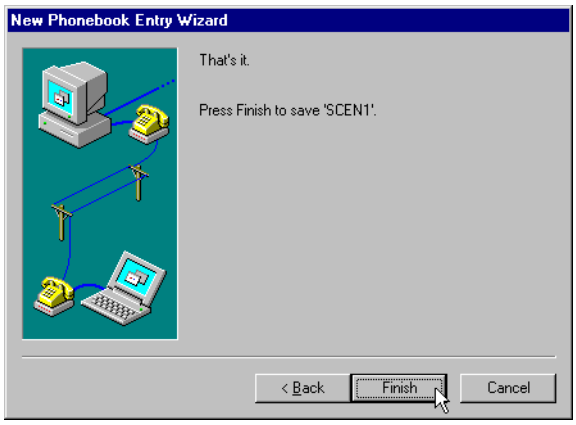


Figure 158. Completing the initial configuration

7. Click **Finish.** This completes addition of the phonebook entry. The display shown in Figure 159 appears.

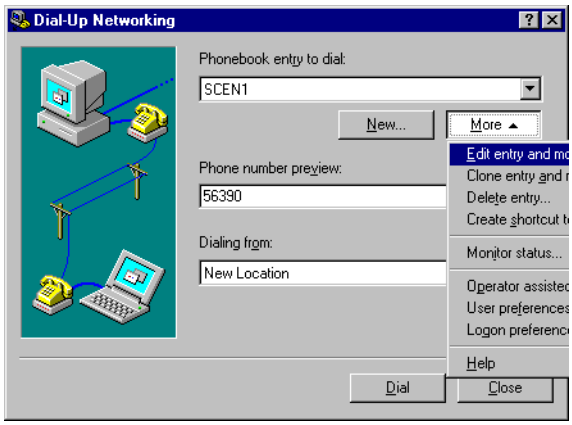


Figure 159. Changing the new connection

8. In the Dial-Up Networking window, select **More->Edit entry...**. The display shown in Figure 160 appears.
9. Use the settings shown in Figure 160, and click **TCP/IP Settings**.

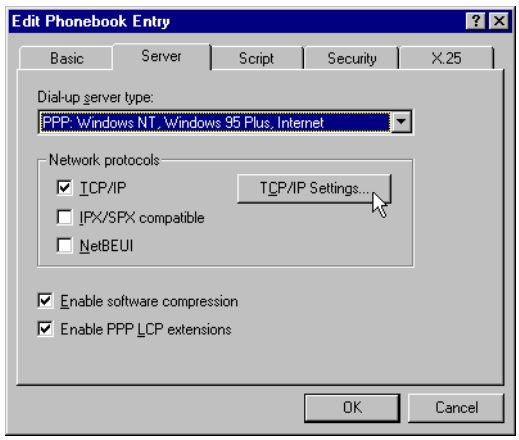


Figure 160. Checking the TCP/IP settings (Part 1)

10. Select TCP/IP in the network protocols box. Select **Enable software compression** and **Enable PPP LCP extensions**. Click **TCP/IP Settings**. The display shown in Figure 161 appears.

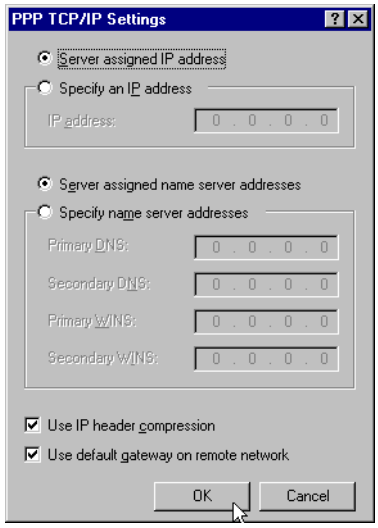


Figure 161. Checking the TCP/IP settings (Part 2)

11. Configure the settings as shown in Figure 161. Click **OK**.

This completes the configuration of the Windows NT system.

4.9.4 Testing the scenario

This section shows how to test the connection that you have defined. It consist of two parts:

- Starting the dial connection
- Starting a TCP/IP application such as TELNET

4.9.4.1 Starting a dial connection with Windows 9x

Use the following procedure to start the dial connection using Windows 9x:

1. From the Dial-Up Networking window, select the connection. The display shown in Figure 162 appears.

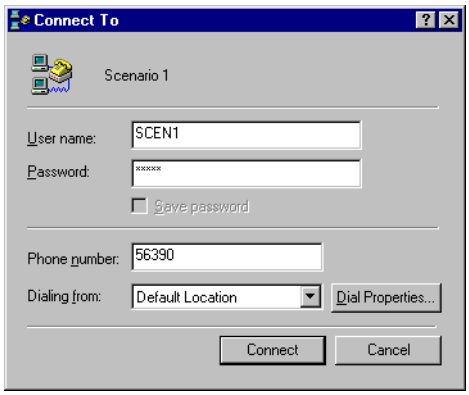


Figure 162. Starting the connection to the AS23 system

2. Enter the User name and the Password, and click **Connect**. If a successful connection is made, the window shown in Figure 163 appears.

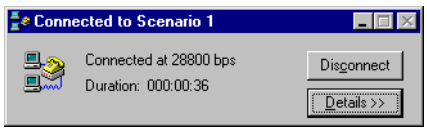


Figure 163. Successful connection to the AS23 system

Go to 4.9.4.3, “Starting a TCP/IP application” on page 127 to continue the testing.

4.9.4.2 Starting a dial connection with Windows NT

Use the following procedure to start the dial connection using Windows NT:

1. From the Dial-Up Networking window, select the connection. The display shown in Figure 164 appears.

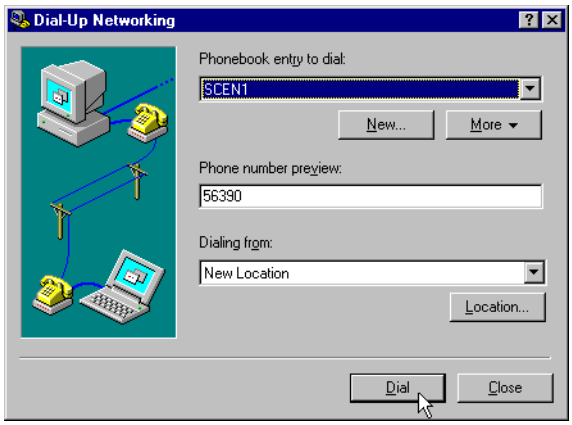


Figure 164. Starting the Windows NT connection

2. Select your phonebook entry. Click **Dial**. The display shown in Figure 165 appears.

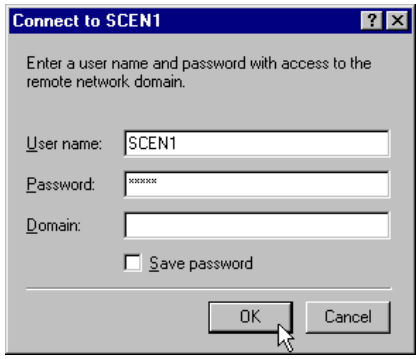


Figure 165. Specifying the user information

3. Enter the user and password information. *Do not* enter a Domain name. Click **OK**. While the connection is trying to connect to the remote system, you see windows similar to the ones shown in Figure 166 and Figure 167.

Figure 166. Dialing the remote system

Figure 167. Verifying the connection parameters

4. When the connection is established, the display shown in Figure 168 appears. Click **OK**.

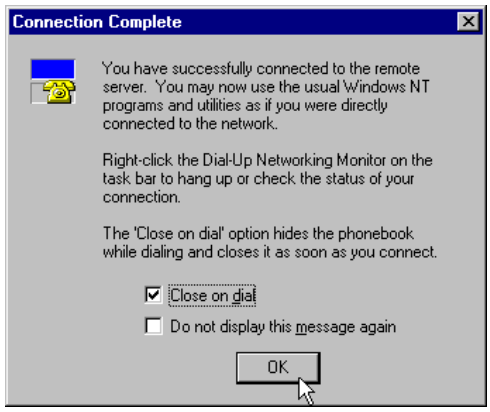


Figure 168. Successful connection to the remote system

You are now ready to start an application.

4.9.4.3 Starting a TCP/IP application

Use the following procedure to start the TCP/IP application TELNET to test the TCP/IP connection. Here we show the use of PC5250 to start a TN5250 session on the AS23 system:

1. Start the PC5250 session using your normal startup procedure. The display shown in Figure 169 appears.

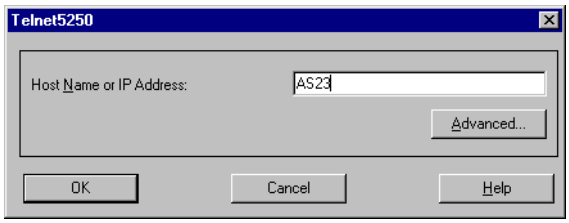


Figure 169. Testing the connection to the AS23 using PC5250

2. Type the name or address of the system. Click **OK**. The display shown in Figure 170 appears.

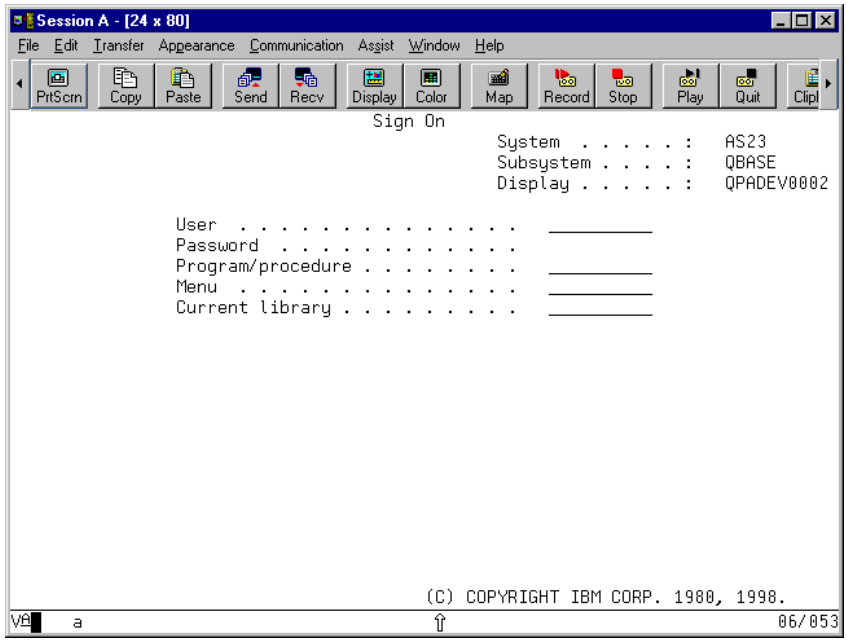


Figure 170. Successful connection to the AS23 system

4.10 Scenario 2: AS/400 dial and AS/400 answer

In this scenario (Figure 107 on page 99), we show the steps used to communicate between two AS/400 systems using TCP/IP PPP. The connection must be manually started before communications can occur. To make this work, you accomplish the following tasks:

- Configure a PPP switched answer connection profile on AS08.
- Configure a PPP switched dial connection profile on AS23.

Refer to Chapter 2, “TCP/IP basic installation and configuration” on page 7, to create the basic setup of the IP interfaces on the AS/400 systems. We used the following IP address on the AS/400 system:

- AS23: 10.1.1.1/24 on the 10.1.1.0/24 network
- AS08: 10.1.2.1/24 on the 10.1.2.0/24 network

4.10.1 Configuring the AS08 system to answer a PPP connection

First we configure the AS08 system to accept the incoming call. Use the following steps to start configurating the PPP connection:

1. Use the procedure in 4.3.3, “Starting Operations Navigator for PPP configuration” on page 90, to access the PPP configuration tree.
2. Click the **Connection Profiles** item. The available profiles appear in the right window.
3. Right-click **Connection Profiles** to show the menu. Select **New Profile**. The **New Point-to-Point Profile Properties** window shown in Figure 171 appears.

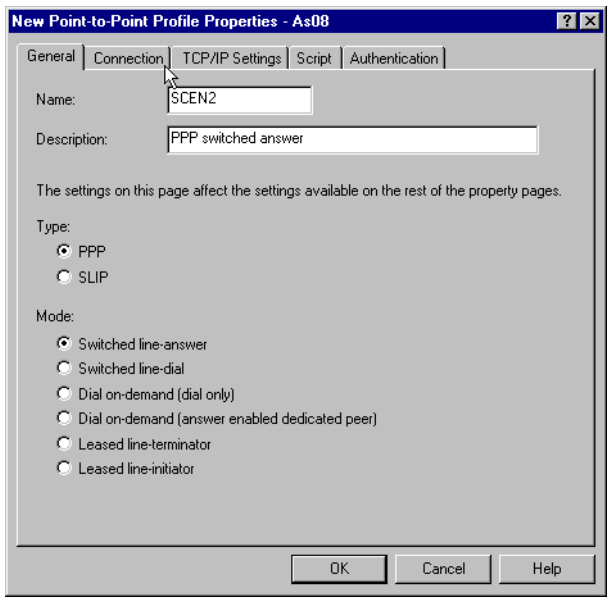


Figure 171. Entering a name, description, type, and mode

4. Enter a name for the profile and a description. Select a type of **PPP** and a mode of **Switched line-answer**. Click **Connection**. The display shown in Figure 172 appears.

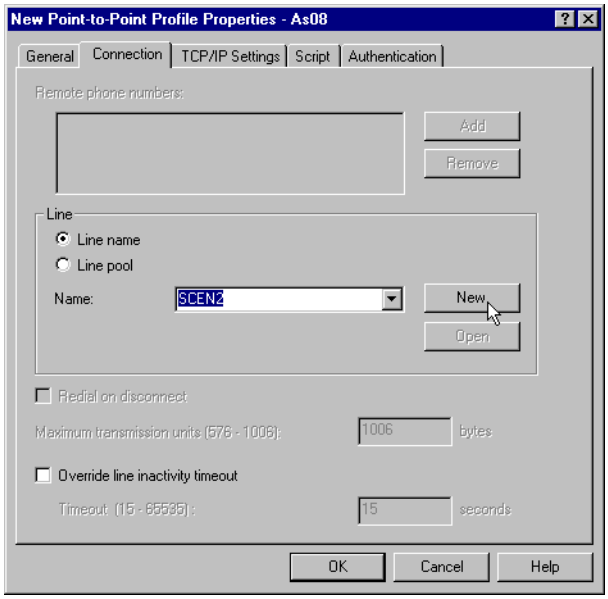


Figure 172. Entering a new line name

5. Type a line name in the Name field, and click **New** to create a new line for the connection. The display shown in Figure 173 appears.

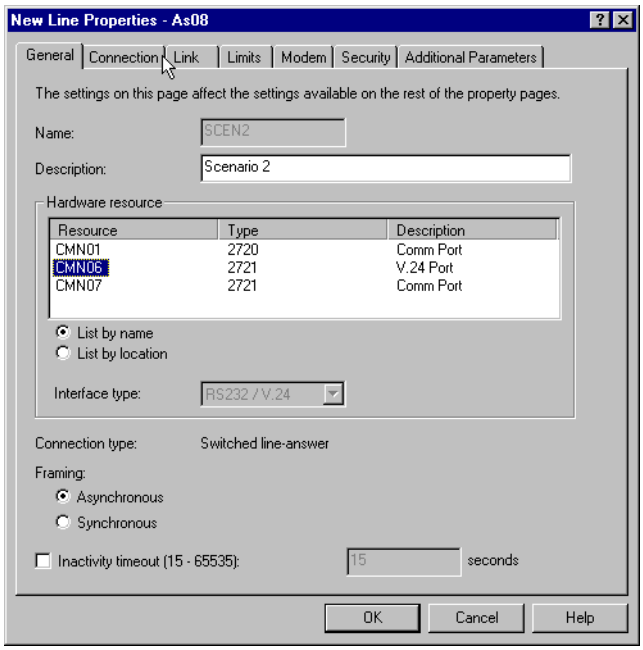


Figure 173. Selecting the correct hardware resource

6. Select the appropriate hardware adapter. Click **Modem**. The display shown in Figure 174 on page 130 appears.

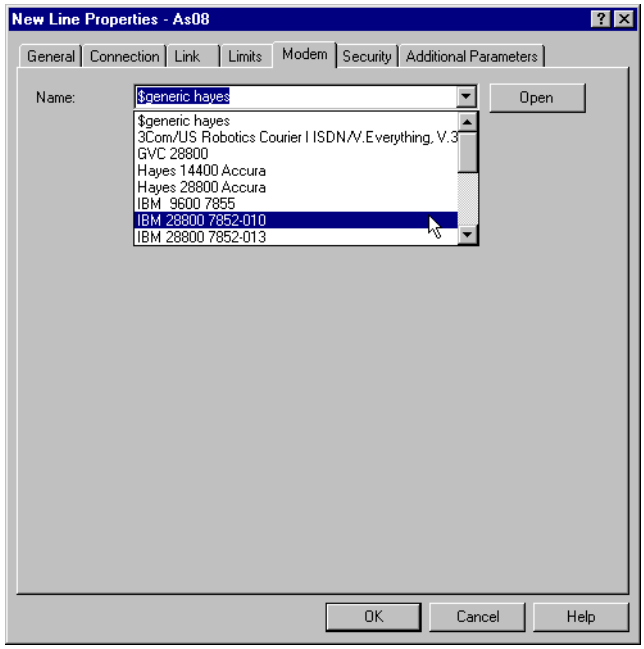


Figure 174. Selecting the modem used

7. Select the modem you are using from the list shown. Click **OK**. The display shown in Figure 172 on page 129 appears. Click **TCP/IP Settings**. The display shown in Figure 175 appears.

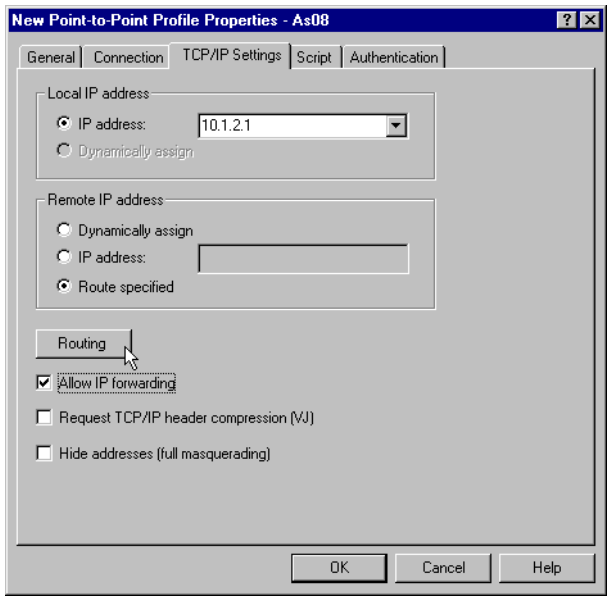


Figure 175. Entering the IP address information

8. In this example, we set up an unnumbered network. We specify Allow IP forwarding to enable other systems in the network to use AS08 and AS23 as routers. To complete this connection, AS23 must also be configured to allow the connect back to the entire 10.1.2 network. For the local IP address, we use the address of the AS/400 LAN adapter 10.1.2.1. For the remote address, we use an address based on the route information. This lets us specify an

address and subnet value for the remote interface rather than just a host address at the other end of the connection. In this case, the address is 10.1.1.1 with a subnet mask of 255.255.255.0. Specify the remote IP address as Route specified and click **Routing**. The display shown in Figure 176 appears.

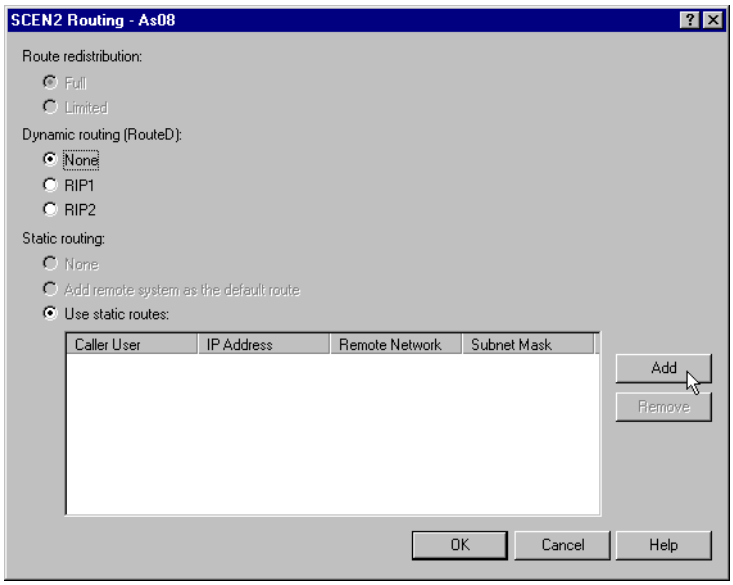


Figure 176. Adding a new static route for the particular user

9. Select **Use static routes**. Click **Add**. The display shown in Figure 177 appears.

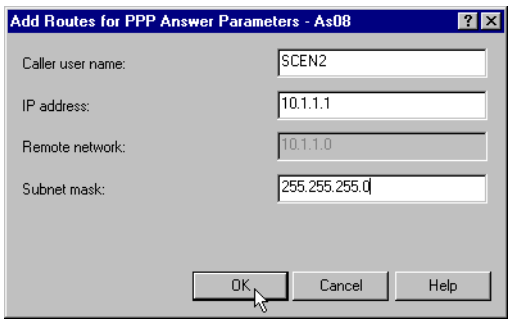


Figure 177. The IP address information depends on the remote user

10. Add the IP address information for this particular user, and click **OK**. In this example, we use the IP address of the LAN adapter on system AS23. The subnet mask is used to define the network address for the adapter. This allows AS08 to communicate with all the systems in the 10.1.1 remote network. The display shown in Figure 178 on page 132 appears.

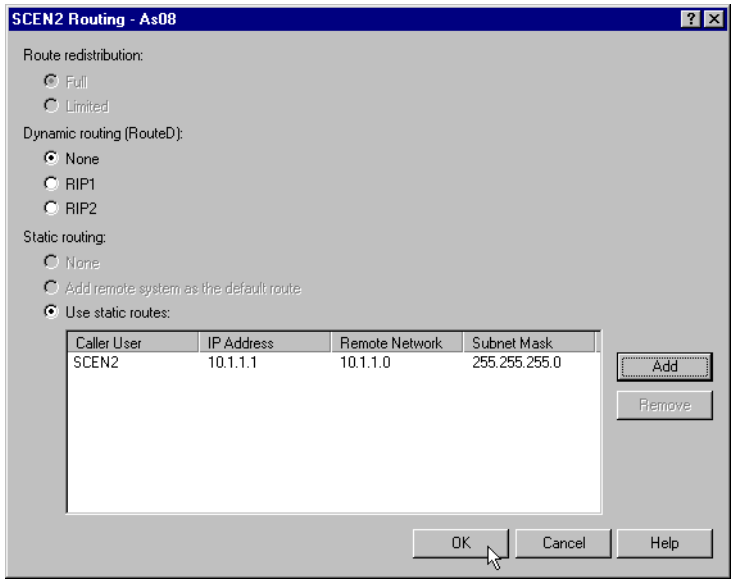


Figure 178. Click OK to finish the routing configuration

11. Notice that the new information has been added to the window. Verify that it is correct, and click **OK** to end the routing configuration. The display shown in Figure 175 on page 130 appears. Click **Authentication**. The display shown in Figure 179 appears.

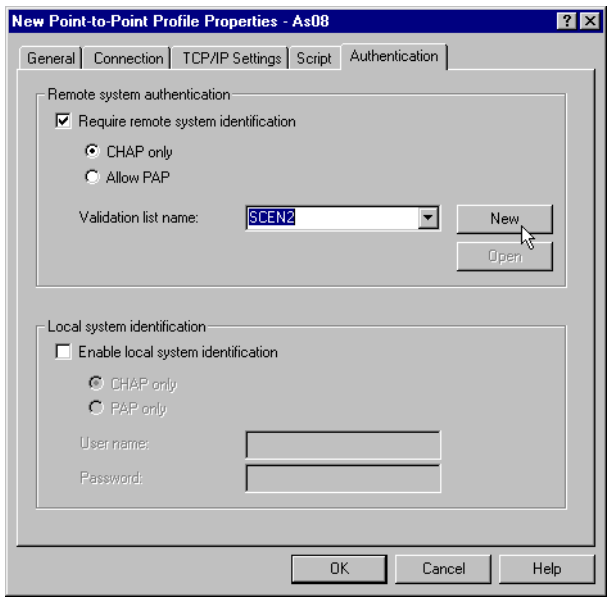


Figure 179. CHAP is required: Entering a new validation list name

12. Check **Require remote system identification**. Select the type **CHAP only** or **Allow PAP**. Enter a validation list name, and click **New** to create a new validation list for the remote users. You may use an existing validation list from the drop-down menu rather than creating a new list. The display shown in Figure 180 appears.

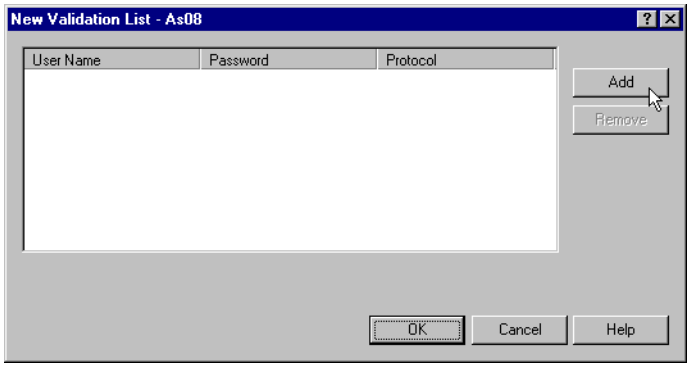


Figure 180. Adding a new remote user using the Add button

13. Click **Add** to add a new user to the validation list. The display shown in Figure 181 appears.

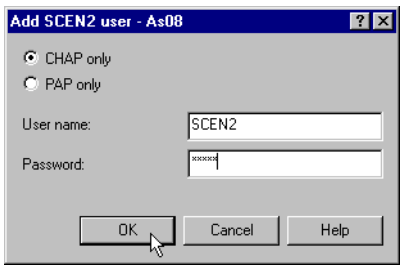


Figure 181. Entering the user name and the password

14. Specify the type, user, and the password. Click **OK**. The display shown in Figure 182 appears.

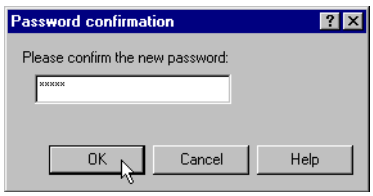


Figure 182. Confirming the password entered

15. Enter the password again to confirm the value entered. Click **OK**. The display shown in Figure 183 on page 134 appears. Notice that the validation list has been added.

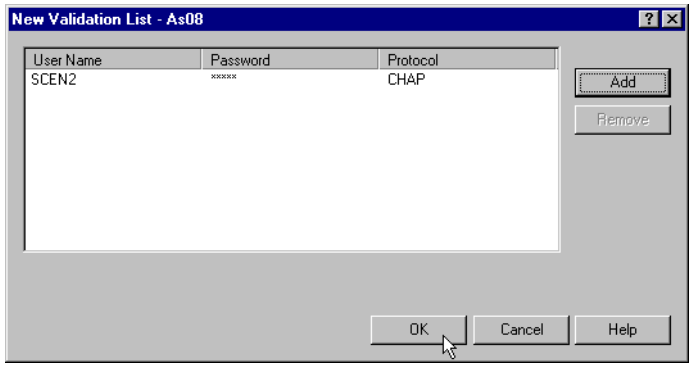


Figure 183. Click OK to finish the configuration

16. Click **OK** to confirm the new validation list. The display shown in Figure 184 appears.

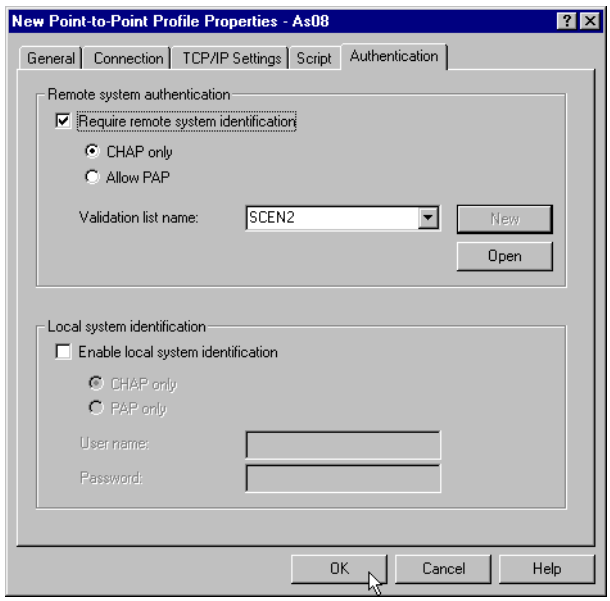


Figure 184. Click OK to confirm the configuration

17. Click **OK** to confirm the creation of the new profile. The display shown in Figure 185 appears. Notice that your profile has been added in the right-hand window.

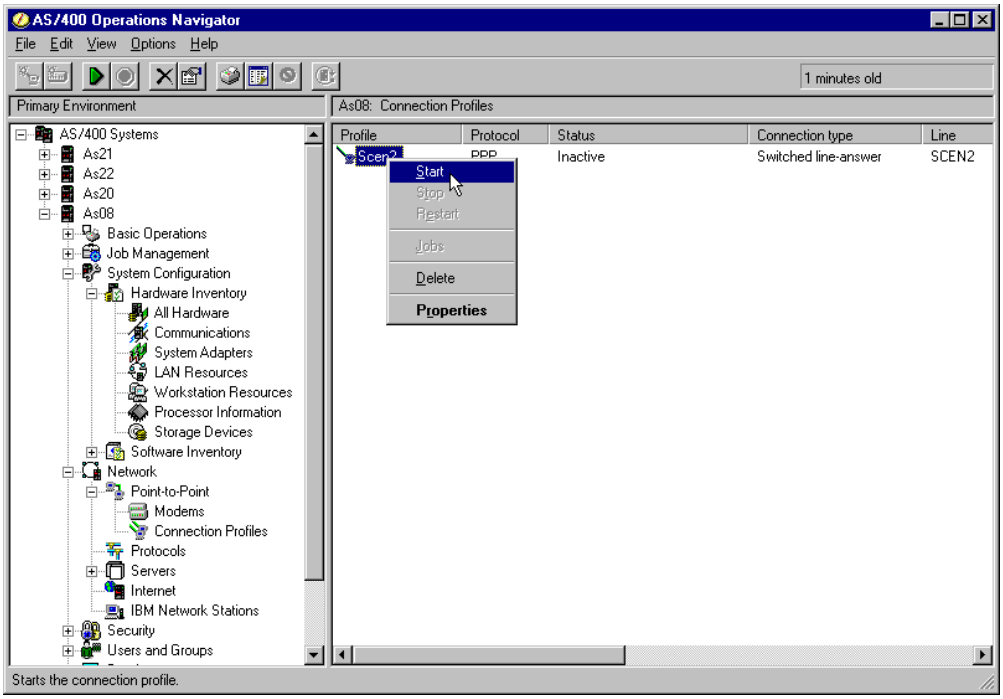


Figure 185. Start the PPP connection profile using the pop-up menu and select Start

18. To start the connection profile, right-click on the profile name. Select **Start** from the pop-up menu. The status changes to `Waiting for incoming call` as seen in Figure 186. You may need to refresh the window contents by pressing F5.

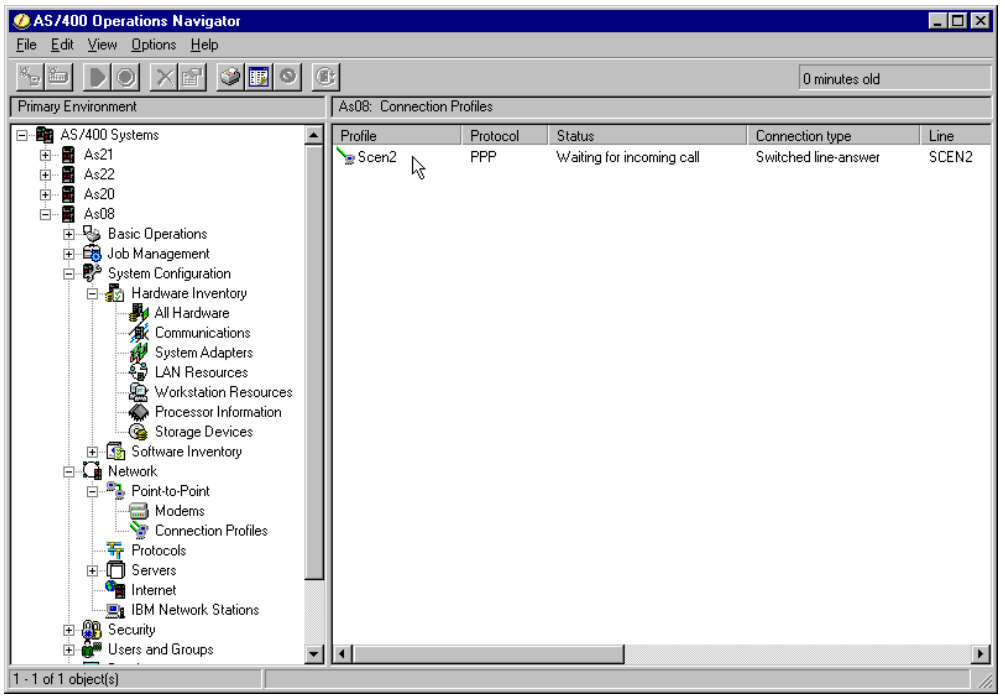


Figure 186. Connection ready for dial-in connections

The AS/400 system is ready for the PPP connection.

4.10.2 Configuring the AS23 system dial PPP connection

In this section, we configure the AS23 system to place the outgoing call. Use the following steps to configure the PPP connection:

1. Use the procedure in 4.3.3, “Starting Operations Navigator for PPP configuration” on page 90, to access the PPP configuration tree.
2. Click the **Connection Profiles** item. The available profiles appear in the right window.
3. Right-click **Connection Profiles** to show the menu. Select **New Profile**. The **New Point-to-Point Profile Properties** window shown in Figure 187 appears.

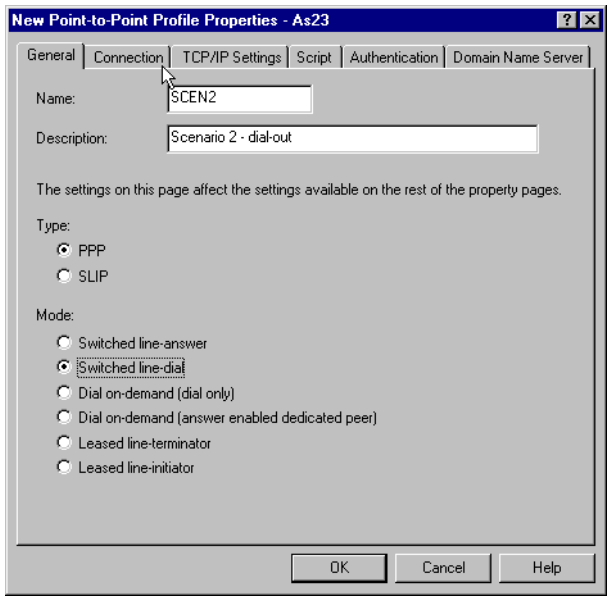


Figure 187. Specifying name, description, and dial mode

4. Enter a name for the profile and a description. Select a Type of **PPP** and a Mode of **Switched line-dial**. Click **Connection**. The display shown in Figure 188 appears.

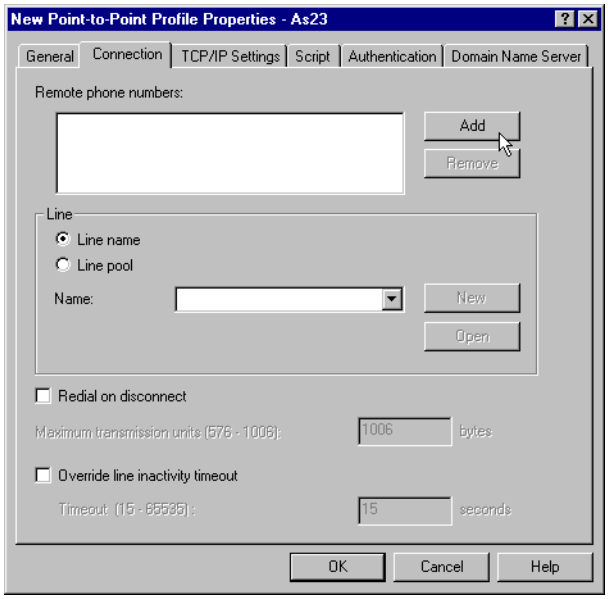


Figure 188. Using the Add button to add a connection number

5. Click **Add**. The input area for the phone number is made available for input. Type the phone number to dial to reach the remote system. Be sure to include any prefix that may be required. Select the **Line Name** field. Type the name of the new line you are creating or select an existing line to use. To create a new line, click **New**. The display shown in Figure 189 appears.

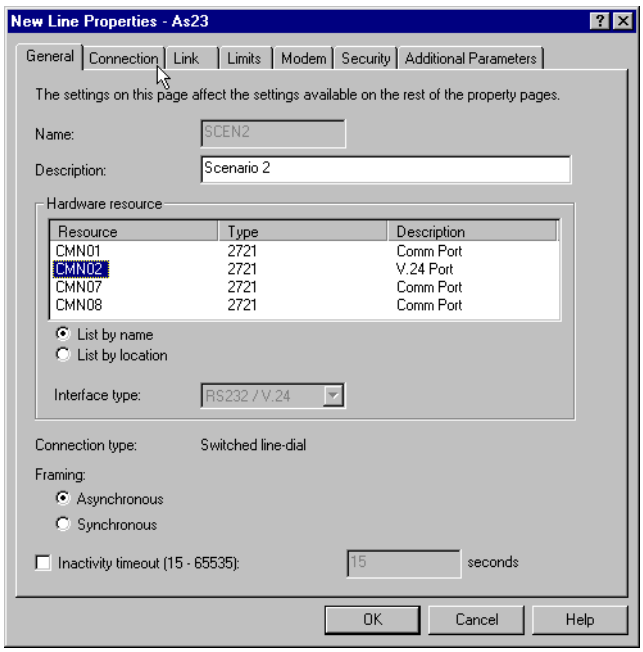


Figure 189. Selecting the hardware adapter

6. Select the appropriate hardware adapter. Click **Modem**. The display shown in Figure 190 on page 138 appears.

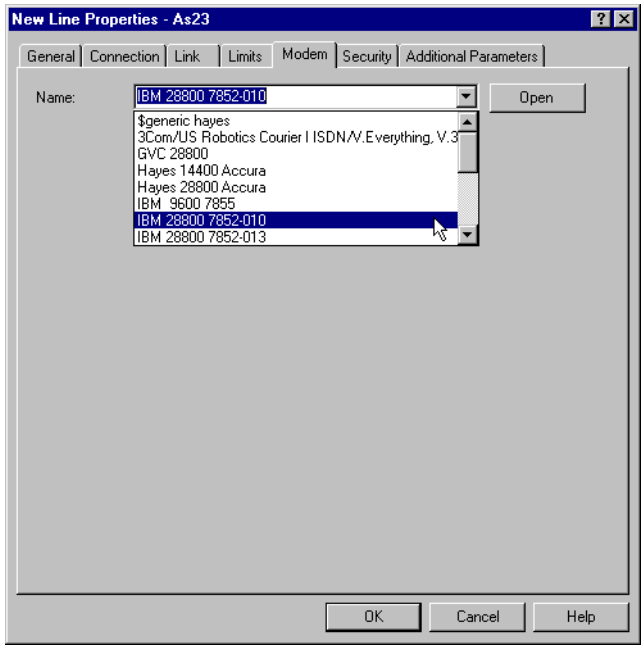


Figure 190. Selecting the appropriate modem

7. Select the modem you are using from the list shown. Click **OK**. The display shown in Figure 188 on page 137 appears. Click **TCP/IP Settings**. The display shown in Figure 191 appears.

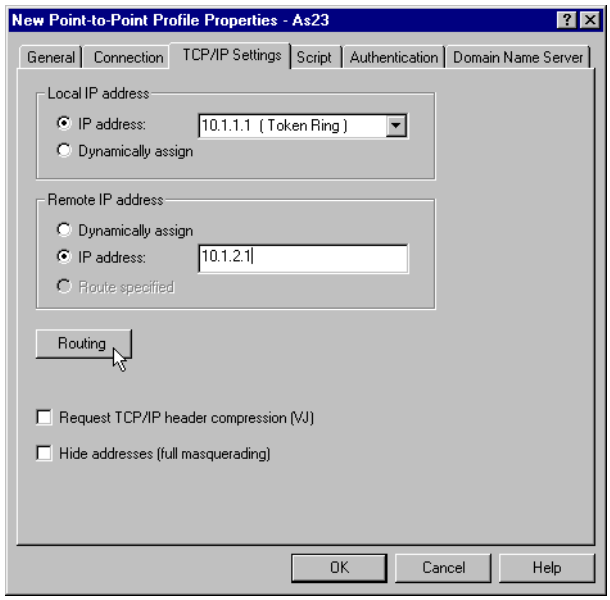


Figure 191. Specifying the local and the remote IP address

8. In this example, we set up an unnumbered network. For the local IP address, we use the address of the AS/400 LAN adapter 10.1.1.1. For the remote address, we use the address of the LAN adapter on the remote system. In this case, the address is 10.1.2.1. In this configuration, only AS23 can communicate with AS08. If we want to allow communications with other systems in the 10.1.2 network, we have to configure this side of the

connection as Route specified as we did in step in of 4.10.1, “Configuring the AS08 system to answer a PPP connection” on page 128. We would use the IP address of 10.1.2.1 and a subnet mask of 255.255.255.0. After you enter the values, click **Authentication**. The display shown in Figure 192 appears.

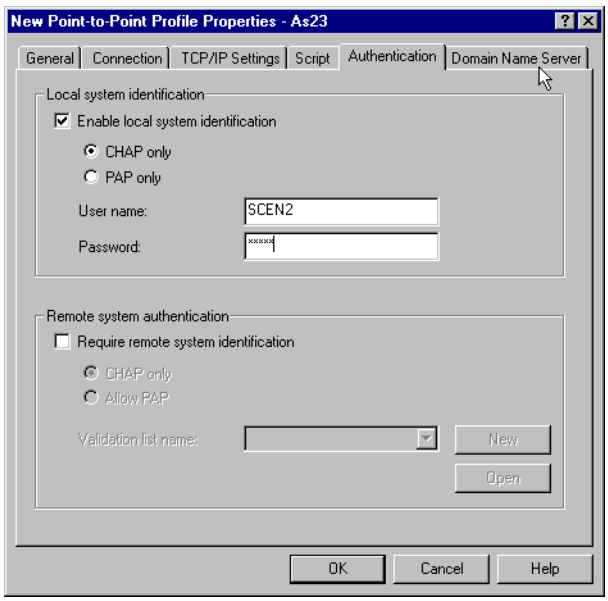


Figure 192. Specifying user and password: Do not get the case wrong

9. Specify the user and the password. Pay attention to the entered values. Mistyping can cause problems at the remote end. The display shown in Figure 193 appears.

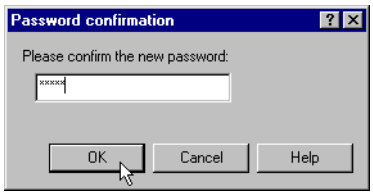


Figure 193. Confirming the password

10. Enter the password again to confirm the value entered. Click **OK**. The display shown in Figure 192 appears. Click **OK** to confirm the creation of the new profile. The display shown in Figure 194 on page 140 appears. Notice that your profile has been added in the right-hand window.

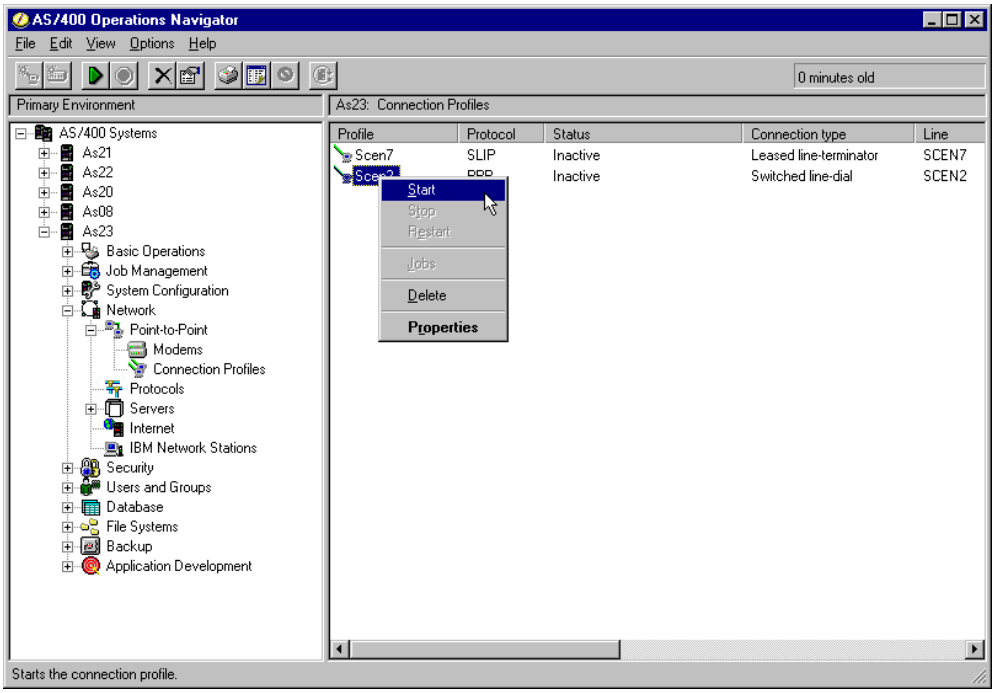


Figure 194. Now start the PPP connection profile

11. To start the connection profile, right-click on the profile name. Select **Start** from the pop-up menu. If you are near the modem and the modem speaker is on, you may hear some dial tones. Monitor the call progress by using the F5 key to refresh the window contents. After some time, you should see the status changes to `Active` as seen in Figure 195.

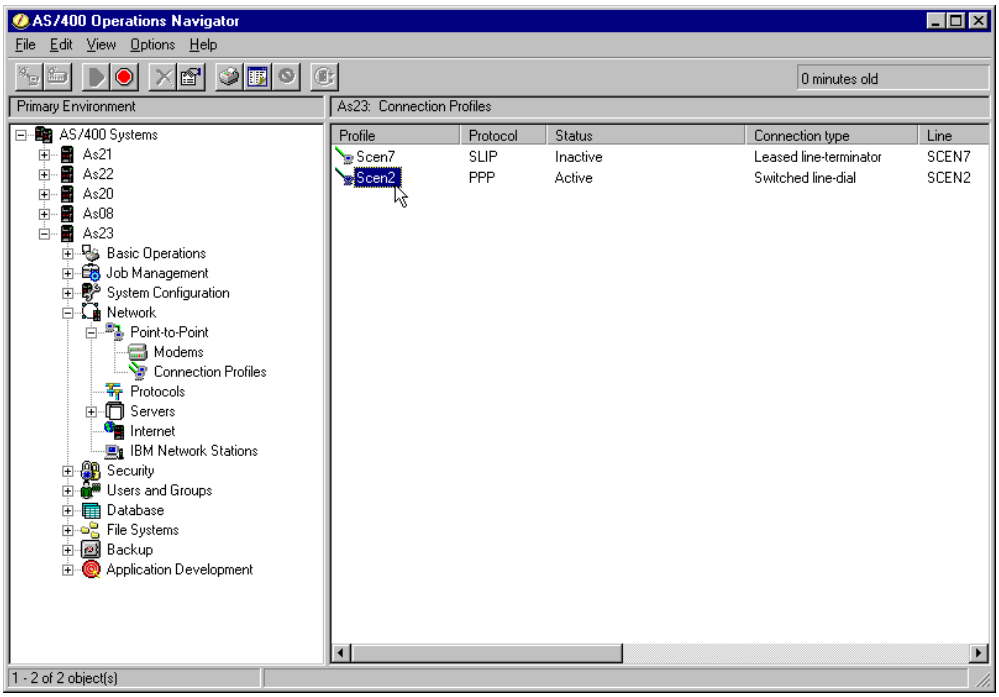


Figure 195. The connection is established and active

You are now ready to use the link.

4.10.3 Testing the scenario

The test of the scenario is performed by using of the Start TCP/IP TELNET (TELNET) command. To perform the test, complete the following steps:

1. Sign on to the AS23 system.
2. On the AS/400 command line, issue the command:

   ```
   TELNET RMTSYS('10.1.2.1')
   ```

   In this example, 10.1.2.1 is the address of the remote system (AS08) that we want to access. Press Enter. If the connection is working, you should be presented with a Sign On screen from the remote system (Figure 196).

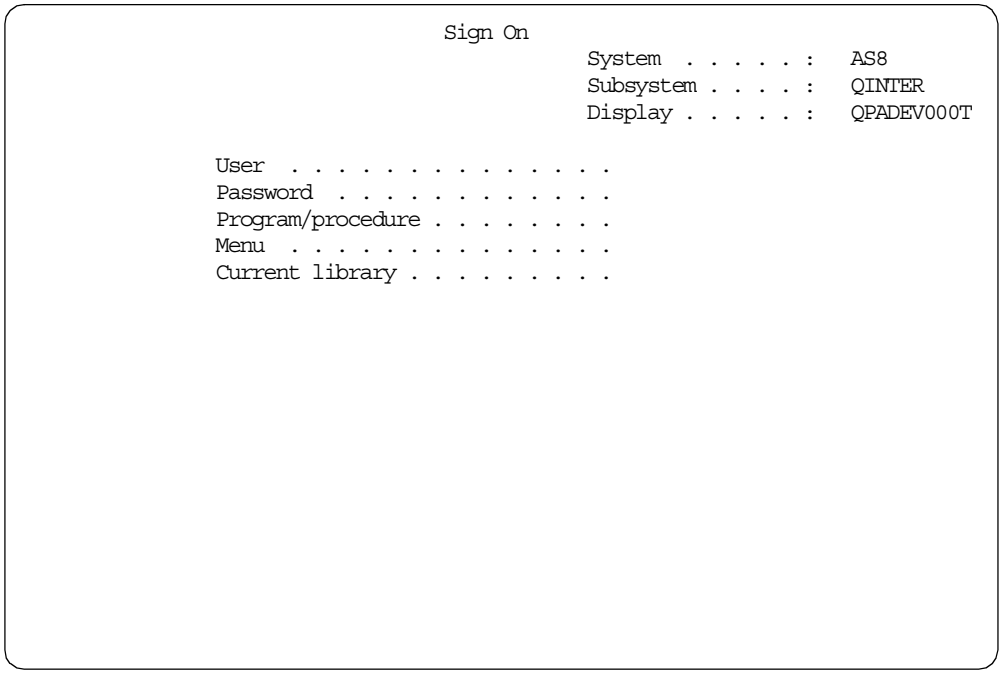


Figure 196. The connection to the AS08 system is successful

3. Sign on and test the link. After you complete the test, sign off and end the Telnet session.

4.11 Scenario 3: AS/400 dial-on-demand to AS/400 answer

In this scenario (Figure 108 on page 100), we show the steps used to communicate between two AS/400 systems using TCP/IP PPP. This time the connection will start when needed. To make this work, you must accomplish the following tasks:

- Configure a PPP switched answer connection profile on AS08.
- Configure a PPP switched dial-on-demand connection profile on AS23.

Refer to Chapter 2, “TCP/IP basic installation and configuration” on page 7, to create the basic setup of the IP interfaces on the AS/400 systems. We used the following IP address on the AS/400 system:

- AS23: 10.1.1.1/24 on the 10.1.1.0/24 network
- AS08: 10.1.2.1/24 on the 10.1.2.0/24 network

4.11.1 Configuring the AS08 system answer PPP connection

We configure the AS08 system to accept the incoming call. Follow the procedure in 4.10.1, “Configuring the AS08 system to answer a PPP connection” on page 128, to build the AS08 PPP connection. For this configuration, we use the value SCEN3 rather than SCEN2.

4.11.2 Configuring the AS23 system dial PPP connection

In this section, we configure the AS23 system to place the outgoing call. Complete the following steps to configure the PPP connection:

1. Use the procedure in 4.3.3, “Starting Operations Navigator for PPP configuration” on page 90, to access the PPP configuration tree.
2. Click the **Connection Profiles** item. The available profiles appear in the right window.
3. Right-click **Connection Profiles** to show the menu. Select **New Profile**. The New Point-to-Point Profile Properties window shown in Figure 197 appears.

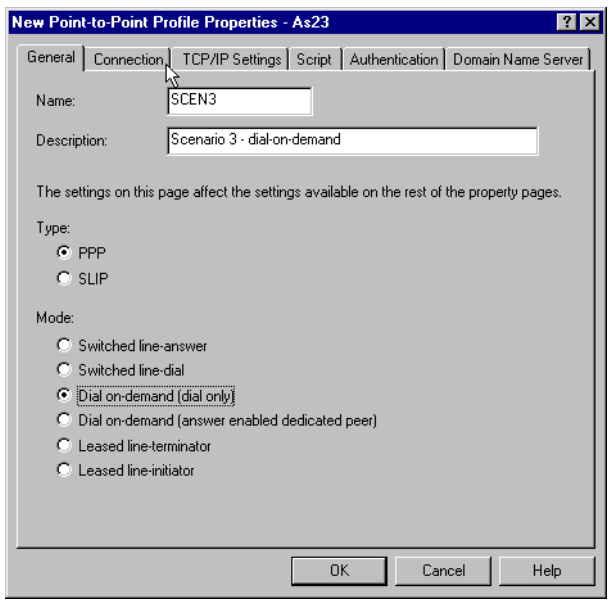


Figure 197. Remember to specify dial-on-demand

4. Enter a name for the profile and a description. Select a type of **PPP** and a mode of **Dial on-demand (dial only)**. Click **Connection**. The display shown in Figure 198 appears.

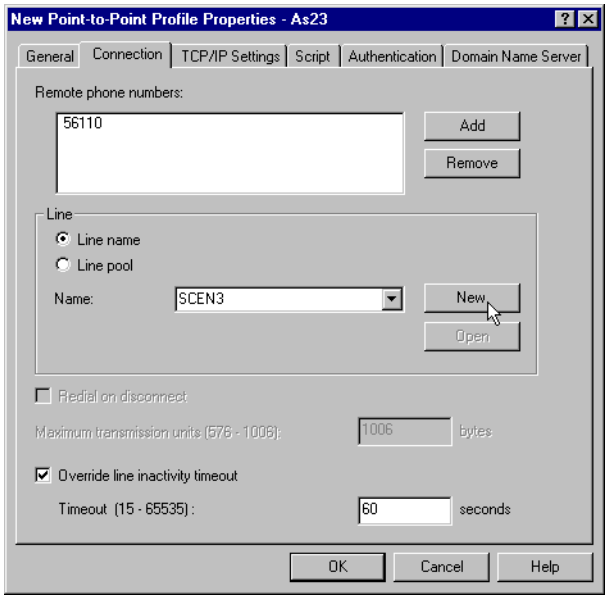


Figure 198. Creating a new PPP line for the connection

5. Click **Add**. The input area for the phone number is made available for input. Type the phone number to dial to reach the remote system. Be sure to include any prefix that may be required. Select **Override line inactivity timeout**, and specify a timeout value. In our example, we choose to drop the PPP connection after being idle for 60 seconds. Select the **Line Name** field. Type the name of the new line you are creating or select an existing line to use. To create a new line, click **New**. The display shown in Figure 199 appears.

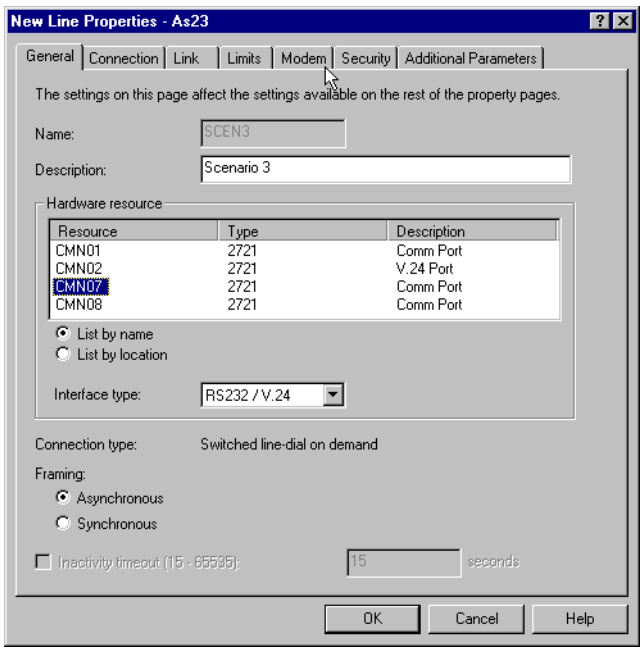


Figure 199. Selecting the hardware adapter to use

6. Select the appropriate hardware adapter. Click **Modem**. The display shown in Figure 200 on page 144 appears.

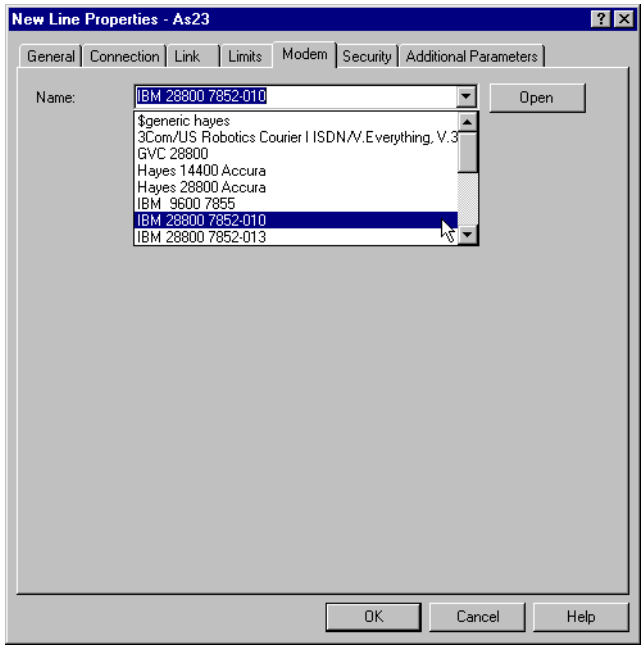


Figure 200. Selecting the modem you are using

7. Select the modem you are using from the list shown. Click **OK**. The display shown in Figure 198 on page 143 appears. Click **TCP/IP Settings**. The display shown in Figure 201 appears.

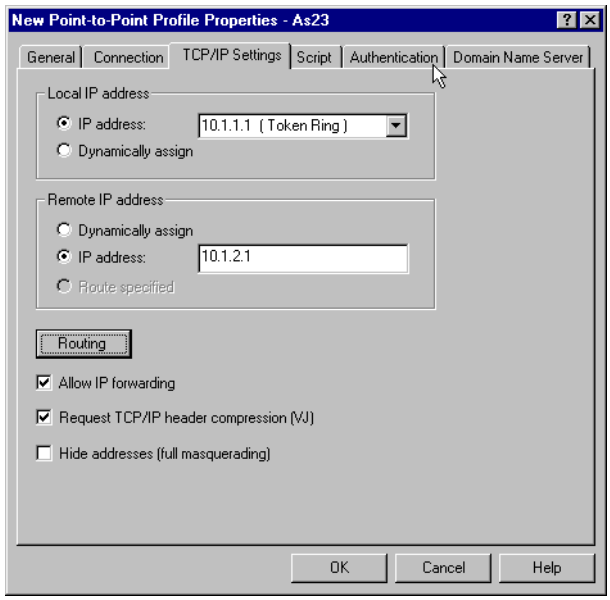


Figure 201. Specifying the IP addresses for the local and the destination system

8. In this example, we set up an unnumbered network. For the local IP address, we use the address of the AS/400 LAN adapter 10.1.1.1. For the remote address, we use the address of the LAN adapter on the remote system. In this case, the address is 10.1.2.1. In this configuration, only AS23 can communicate with AS08. If we want to allow communications with other systems in the 10.1.2 networks, we would have to configure this side of the

connection as Route specified as we did in step 8 on page 130 4.10.1, “Configuring the AS08 system to answer a PPP connection” on page 128. We would use the IP address of 10.1.2.1 and a subnet mask of 255.255.255.0 After you enter the values, click **Authentication**. The display shown in Figure 202 appears.

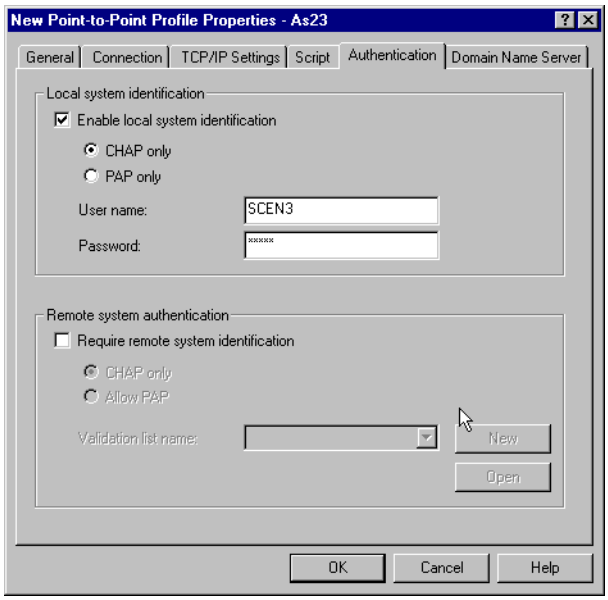


Figure 202. Specifying the user ID and password

9. Specify the user and the password. Pay attention to the entered values. Mistyping can cause problems at the remote end. The display shown in Figure 203 appears.

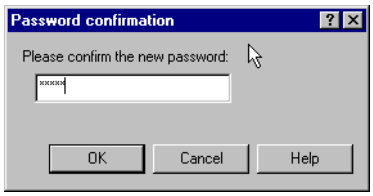


Figure 203. Confirming the password

10. Enter the password again to confirm the value entered. Click **OK**. The display shown in Figure 202 appears. Press **OK** to confirm the creation of the new profile. The display shown in Figure 204 on page 146 appears. Notice that your profile has been added in the right-hand window.

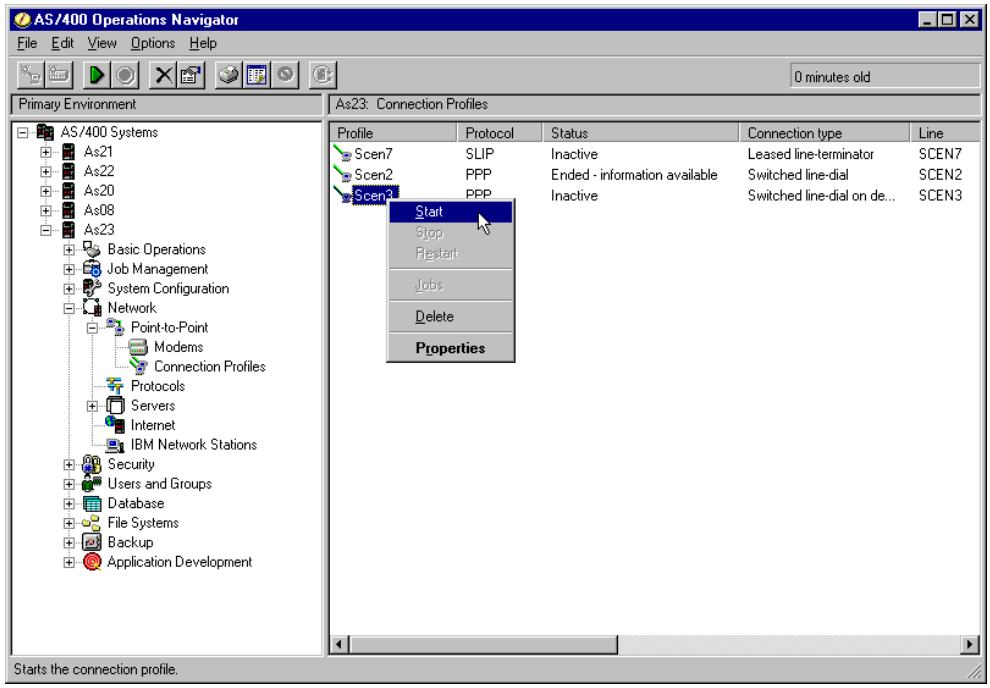


Figure 204. Start the PPP connection

11. To start the connection profile, right-click on the profile name. Select **Start** from the pop-up menu. The status changes to `Waiting for dial - Switched line on demand` as shown in Figure 205. You may need to refresh the window contents by pressing F5.

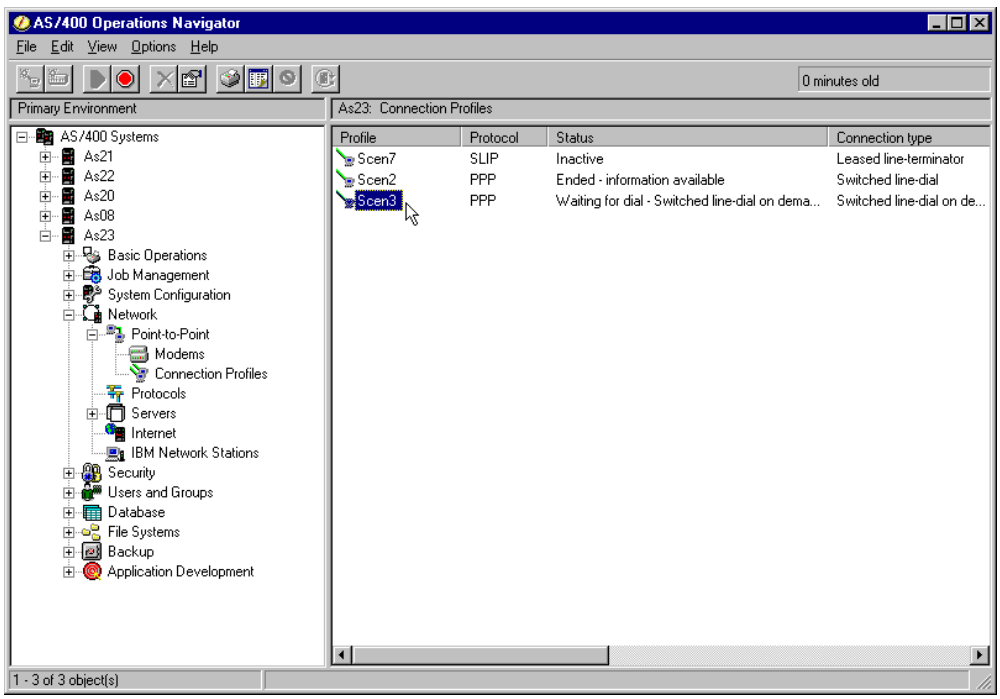


Figure 205. The PPP connection is now ready for dial-on-demand

You are now ready to test the connection.

4.11.3 Testing the scenario

The test of the scenario is done using the Start TCP/IP File Transfer Protocol (FTP) command from the AS23 system to the AS08 system (10.1.2.1). To perform the test, complete the following steps:

1. Sign on to the AS23 system.
2. On the AS/400 command line, issue the command:

   ```
   FTP RMTSYS('10.1.2.1')
   ```

   In this example, 10.1.2.1 is the address of the remote system (AS08) that we want to access. Press Enter. This causes the dial-on demand function to dial the remote system.
3. Switching to the Operations Navigator (Figure 206) and pressing F5 (Refresh) shows the progress as the AS23 system initiates the dial-on-demand connection to the AS08 system. After a few minutes, you should see the status of the connection change to `Active` (Figure 207 on page 148).

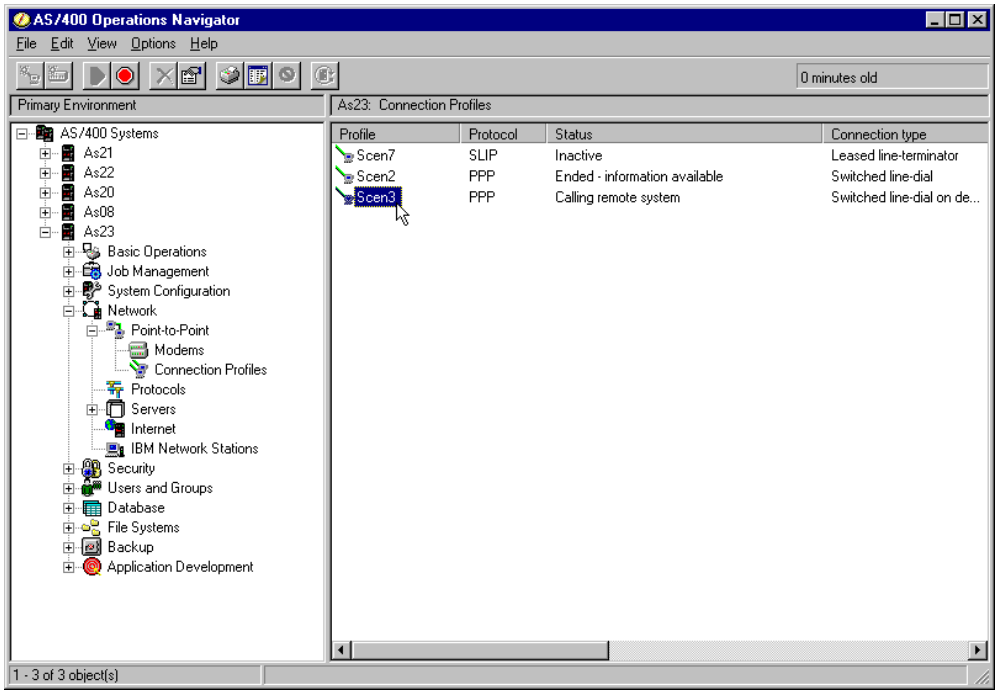


Figure 206. The dial-on-demand connection is connecting to AS08

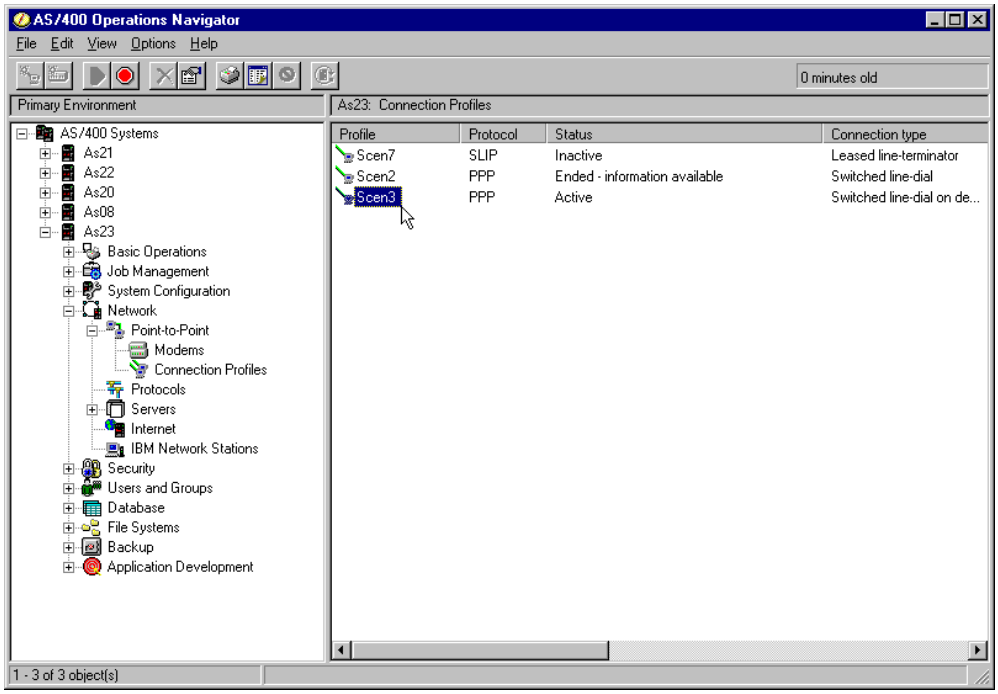


Figure 207. The PPP connection is now ready

4. Switch back to the green-screen interface (Figure 208), which shows that the AS23 system has connected to the AS08 (10.1.2.1) FTP server. Note that the time-out-value (5 minutes) shown in Figure 208 has nothing to do with the PPP disconnect value. The value shown in the figure only applies to the FTP application.

```
                          File Transfer Protocol

Previous FTP subcommands and messages:
  Connecting to remote host 10.1.2.1 using port 21.
  220-QTCP at 10.1.2.1.
  220 Connection will close if idle more than 5 minutes.









Enter login ID (userid):
===>



F3=Exit      F6=Print       F9=Retrieve
F17=Top    F18=Bottom    F21=CL command line
```

Figure 208. The FTP session is started after the connection is made

After the FTP session has ended and the PPP connection has been idle for more that 60 seconds, the PPP connection automatically disconnects. This is shown in the system operators message queue in Figure 209. The two messages at the bottom show the SNMP trap generated when the PPP connections was activated and when the connection was disconnected.

```
                              Display Messages
                                                          System:   AS23
Queue . . . . . :   QSYSOPR                Program . . . . :   *DSPMSG
  Library . . . :     QSYS                   Library . . . :
Severity  . . . :   90                     Delivery  . . . :   *HOLD

Type reply (if required), press Enter.
  Starting TCP/IP point-to-point session for profile SCEN3.
  SNMP linkUp trap generated.
  SNMP linkDown trap generated.
                                                                        Bottom
F3=Exit             F11=Remove a message                 F12=Cancel
F13=Remove all      F16=Remove all except unanswered     F24=More keys
```

Figure 209. QSYSOPR messages: SNMP traps

4.12 Scenario 4: AS/400 dial to Windows NT answer

In this scenario (Figure 109 on page 100), we show how an AS/400 system can dial a Windows NT system. This may be useful to transfer data between the systems without requiring a LAN connection or a dedicated line between the systems. We use the AS/400 dial-on demand function to start the connection as needed.

This scenario accomplishes these tasks:

- Configure a PPP dial-on-demand connection profile on AS23.
- Configure a PPP switched answer connection on the Windows NT system using RAS.

Refer to Chapter 2, “TCP/IP basic installation and configuration” on page 7, to create the basic setup of the IP interfaces on the AS/400 systems. We used the following IP address on the AS/400 system:

- AS23: 10.1.1.1/24 on the 10.1.1.0/24 network
- AS08: 10.1.2.1/24 on the 10.1.2.0/24 network

4.12.1 Configuring the AS23 system dial PPP connection

We configure the AS23 system to place the outgoing call when needed. Follow the procedure in 4.11.2, “Configuring the AS23 system dial PPP connection” on page 142, to build the AS23 PPP connection. For this configuration, we use the value SCEN4, rather than SCEN3.

4.12.2 Configuring the Windows NT system answer PPP connection

This section describes how to configure Dial-Up Networking on an Windows NT system. Remote Access Service (RAS) must be installed on the Windows NT system. For a procedure to check for the presence of RAS, refer to steps 1 through 3 in 4.9.3, “Configuring the Windows NT PPP connection” on page 118.

> **Note**
>
> During this procedure, you may need your Windows operating system media. Insert the media and point to the drive when prompted. You may also be asked to restart your Windows system. Follow the directions on the screen, and restart as requested.

4.12.3 Installing Remote Access Service (RAS)

This section provides you with a procedure for installing RAS on the Windows NT system. The configuration includes instructions for installing RAS on the Windows NT system. If RAS is already installed, you can skip this section.

1. Click **Start->Settings->Control Panel** to access the Windows Control Panel (Figure 210).

Figure 210. Selecting the Network icon in the Control Panel

2. Double-click the **Network** icon. The display shown in Figure 211 appears.

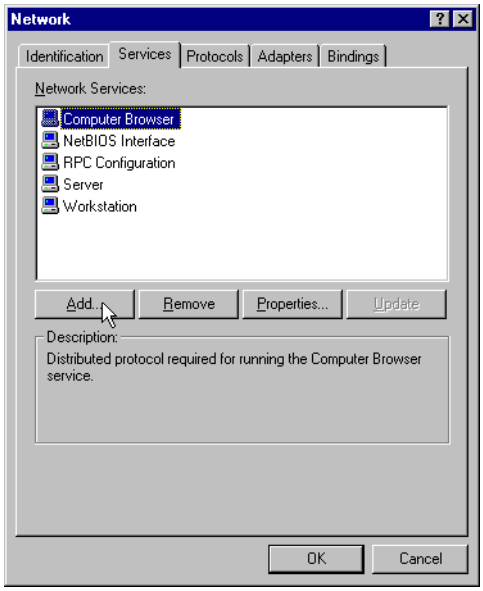


Figure 211. Adding the RAS service to the Windows NT system

3. Click the **Services** tab, and click **Add**. The display shown in Figure 212 appears.

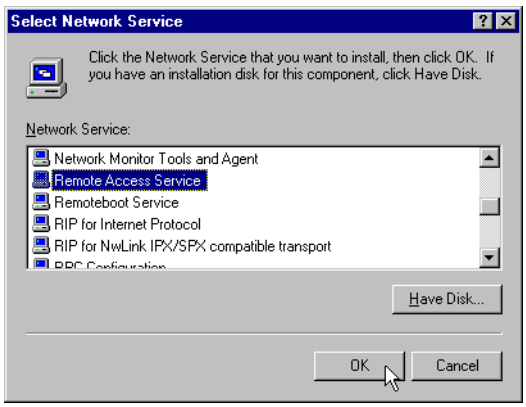


Figure 212. Selecting the RAS service

4. Select **Remote Access Service**, and click **OK**. The display shown in Figure 213 appears.

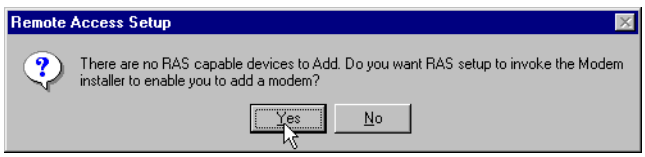


Figure 213. RAS setup question

5. Click **Yes**. The display shown in Figure 214 on page 152 appears.

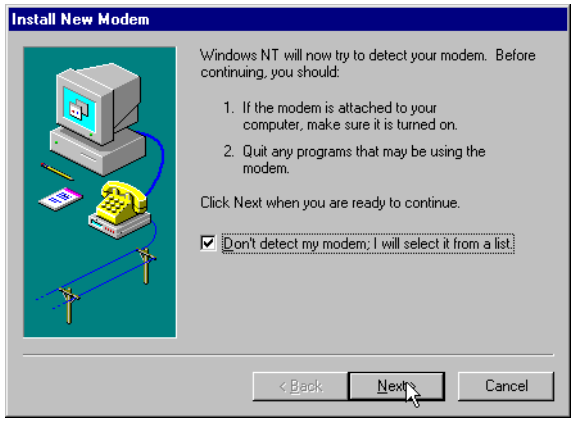


Figure 214. Installing a new modem

6. Select **Don't detect my modem. I will select it from a list**, and click **Next**. The display shown in Figure 215 appears.

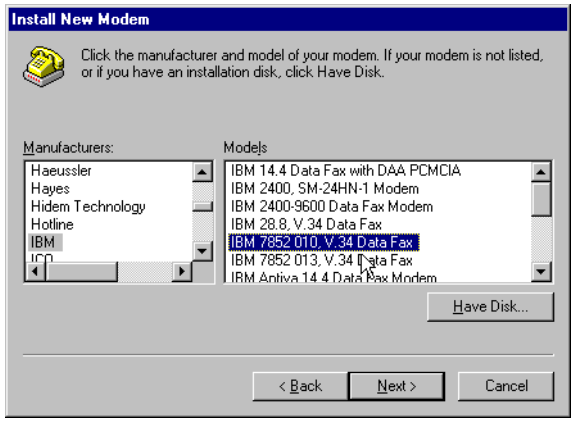


Figure 215. Specifying a modem

7. Select your modem manufacture and model from the list. In our case, we selected IBM in the left window and the IBM 7852 010 modem in the right window. Click **Next**. The display shown in Figure 216 appears.

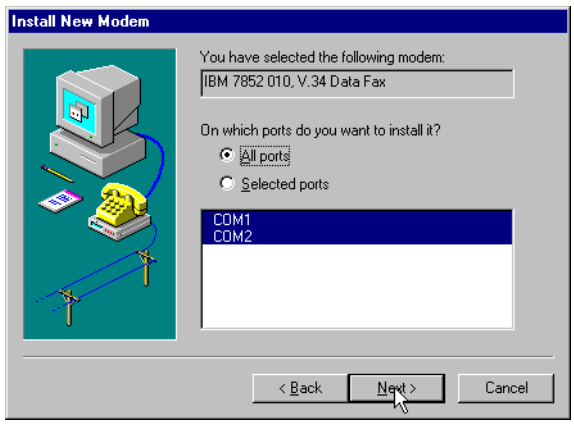


Figure 216. Selecting ports to use

8. Select **All ports**. Click **Next.** The display shown in Figure 217 appears.

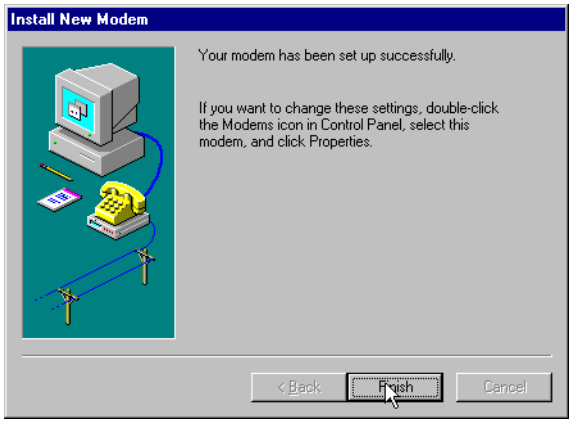


Figure 217. Finishing the modem setup

9. Click **Finish**. The display shown in Figure 218 appears.

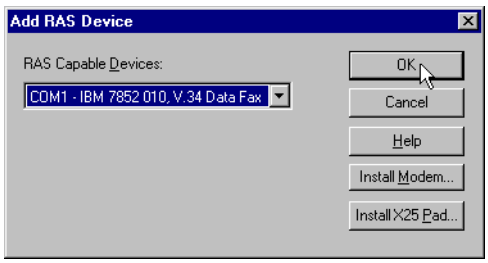


Figure 218. Adding a RAS device

10. Click **OK**. The display shown in Figure 219 appears.

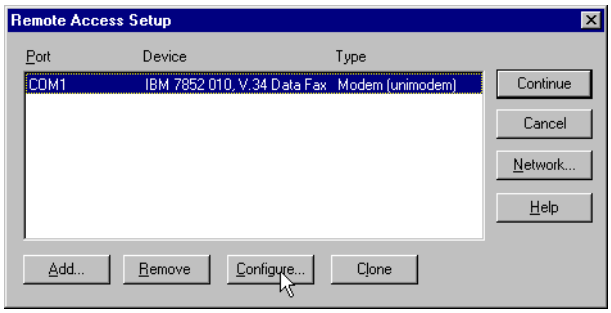


Figure 219. Configuring RAS

11. Select the port you want to configure from the list, and click **Configure**. The display shown in Figure 220 on page 154 appears.

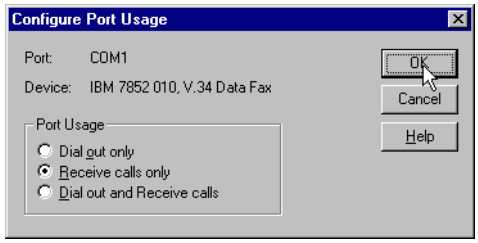


Figure 220. Selecting Port Usage

12. Select how you want this port to be used. We do not want this system to dial out, so we select **Receive calls only**, and click **OK**. The display shown in Figure 219 on page 153 appears. Click **Continue**. The display shown in Figure 221 appears.

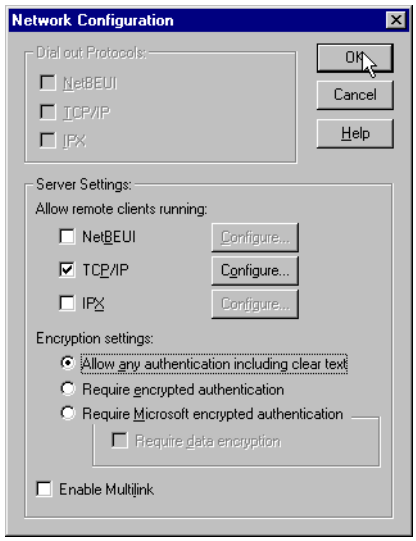


Figure 221. Specifying the network configuration

13. Select **TCP/IP** in the Server Settings section. Select the Encryption setting **Allow any authentication including clear text**. Click **Configure**. The display shown in Figure 222 appears.

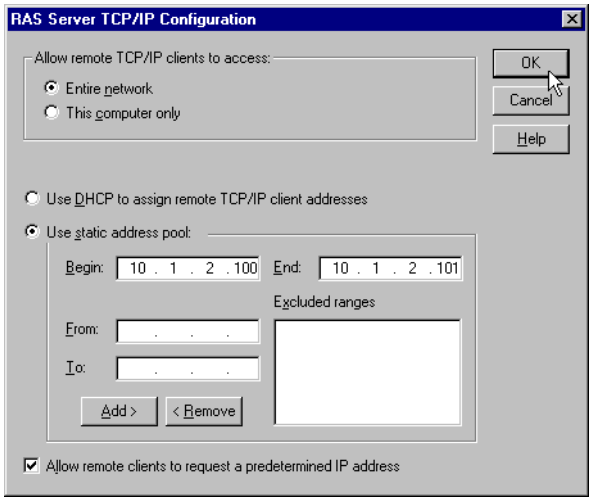


Figure 222. Specifying IP addresses

14. Specify the level of access you want to provide to the inbound connection. In our example, we want to allow the AS/400 system access to all the systems in the network. We select **Entire network**. We want to use a static address that the AS/400 system will provide so we select **Use static address pool**. In this example, we used 10.1.2.100 on the AS/400 system as the remote address. Here we specify an address range that allows that address. Because the AS/400 system is sending the address to the Windows NT system, we check **Allow remote clients to request a predetermined IP address**. Click **OK**. The display shown in Figure 223 appears.

Figure 223. Completing the RAS installation

15. Click **OK**. This completes the RAS installation.

16. You use the Remote Access Administrator tool to check the status of a RAS connection. On the Windows NT menu bar, click **Start->Programs->Administrative Tools (Common)->Remote Access Admin** to check the status of a RAS connection. The display shown in Figure 224 on page 156 appears.

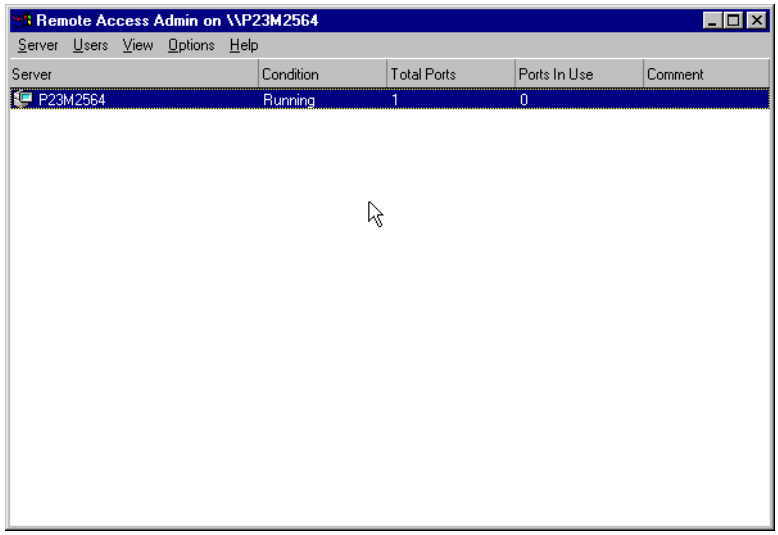


Figure 224. The Remote Access Service is running

You may use the menu options to check aspects of the connection. In this example (Figure 224), we see that the connection is running.

You now must set up a user.

4.12.4 Setting up a dial-in user on Windows NT

To make the connection work, you must set up a user to be used for dial-in access. Use the following procedure to setup the user:

1. On the Windows NT menu bar, click **Start->Programs->Administrative Tools (Common)->User Manager for Domains.** The display shown in Figure 225 appears.

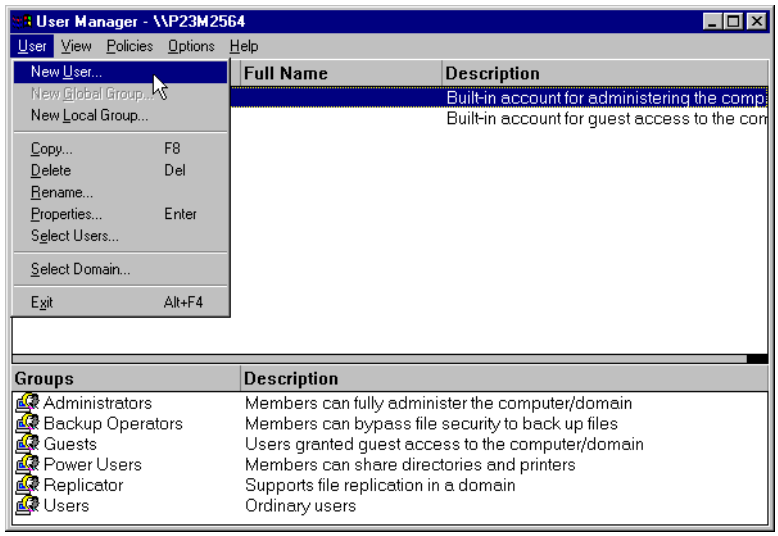


Figure 225. Creating a new user (Part 1)

2. Click **User->New User** to create a new user. The display shown in Figure 226 appears.

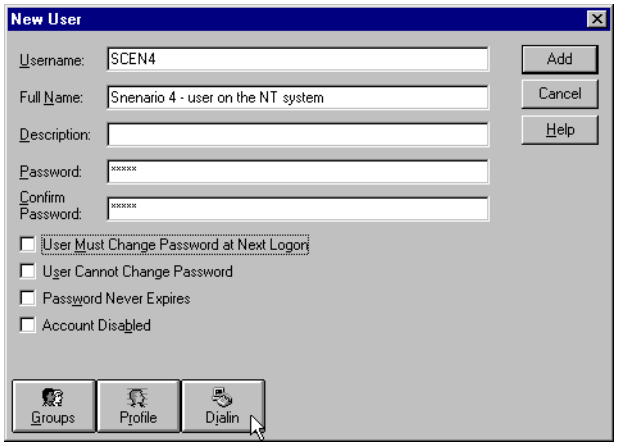


Figure 226. Creating a new user (Part 2)

3. Enter the user information. Be sure that the user name and password match exactly with the value entered in the configuration of the PPP profile on the AS/400 system. These values were configured in steps 9 and 10 of 4.11.2, “Configuring the AS23 system dial PPP connection” on page 142. After you enter the values, click **Dialin**. The display shown in Figure 227 appears.

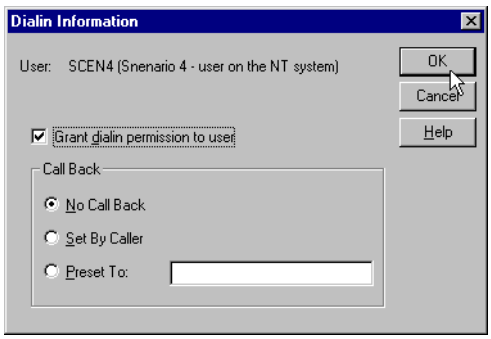


Figure 227. Granting the user dial-in access

4. Select **Grant dialin permission to user**. This makes this user a valid dial-in user. Click **OK** to complete the creation of the user.

4.12.5 Testing the scenario

The test of the dial-on-demand connection to the Windows NT system can be done by using the Start TCP/IP TELNET (TELNET) command. This requires that a Telnet daemon is running on the Windows NT system. In this case, no Telnet daemon is running on the Windows NT system, but we can still try to connect to the NT system with the Telnet command. Instead of using the well-known Telnet TCP port (23), we are going to connect to well-known TCP port (19). If it is installed at the NT system, a CHARGEN (Character Generator) service is running, listening at port 19.

If you cannot test the connection with the Telnet command, you can use the PING command to test the connection. Please note that the WAITTIME parameter on the PING command should be set to 60 seconds to allow the dial-on-demand connection between the two systems to be established.

To perform the test, complete the following tasks:

1. Sign on to the AS23 system.
2. On the AS/400 command line, issue the command:

   ```
   TELNET RMTSYS('10.1.2.100') INZWAIT(*NOMAX) PORT(19)
   ```

   In this example, `10.1.2.100` is the address of the remote system that we want to access. Press Enter. If the connection is working, you should be presented with an echo screen from the remote system (Figure 228).

```
opqrstuvwxyz{|} !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWX
pqrstuvwxyz{|} !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXY
qrstuvwxyz{|} !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ
rstuvwxyz{|} !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[
stuvwxyz{|} !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\
tuvwxyz{|} !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]
uvwxyz{|} !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^
vwxyz{|} !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_
wxyz{|} !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`
xyz{|} !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`a
yz{|} !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`ab
z{|} !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abc
{|} !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcd
|} !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcde
} !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdef
 !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefg
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefgh
"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghi
#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghij
$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijk
%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijkl
```

Figure 228. Successful connection to Chargen service at the Windows NT system

4.13 Scenario 5: AS/400 dial to an Internet Service Provider

In this scenario (Figure 110 on page 101), we show how an AS/400 system can dial an Internet Service Provider (ISP) and access the Internet. This is useful when used from a test system, for example, to access an FTP site. This connection profile must be manually started before it can be used for access.

Note

Placing a system on the Internet exposes it to a variety of threats. You should not do this with a production system or any other system that you care about. If you are going to connect a system to the Internet, you must make sure it is secure (at least level 30 security). You should consider setting filter rules to help protect the system.

This scenario accomplishes these tasks:

- Collects PPP connection information from your ISP
- Configures a PPP switched dial connection profile on AS20

4.13.1 Gathering the ISP information

Contact your ISP to get the information you need to create the PPP connection profile. Use Table 5 to collect the information.

Table 5. ISP information

ISP information	Value
Phone number to connect to	
User name	
Password	
Authentication type (PAP or CHAP)	
DNS Server	

4.13.2 Configuring the AS20 system dial PPP connection

In this section, we configure the AS20 system to place the outgoing call to the ISP. We use the values shown is Table 6 in our example.

Table 6. Sample ISP information

ISP information	Value
Phone number to connect to	9,555-5555
User name	YourUser
Password	password
Authentication type (PAP or CHAP)	CHAP
DNS Server	1.2.3.4

Use the following steps to configure the PPP connection:

1. Use the procedure in 4.3.3, “Starting Operations Navigator for PPP configuration” on page 90, to access the PPP configuration tree.
2. Click the **Connection Profiles** item. The available profiles appear in the right window.
3. Right-click **Connection Profiles** to show the menu. Select **New Profile**. The New Point-to-Point Profile Properties window shown in Figure 229 on page 160 appears.

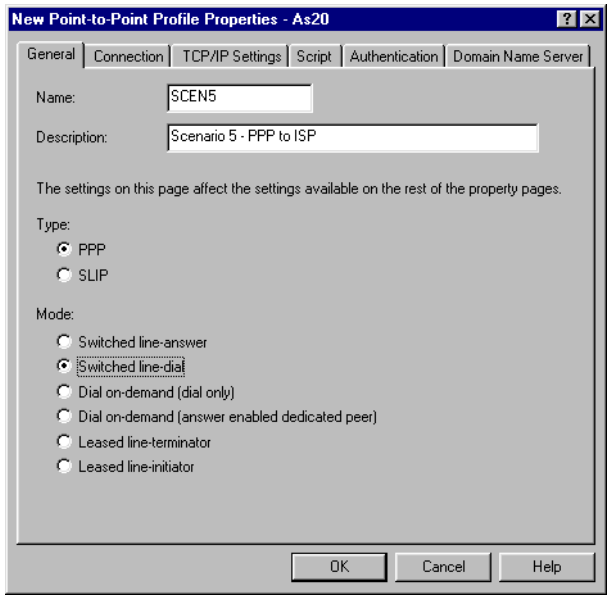


Figure 229. Entering basic settings for PPP connection to the ISP

4. Enter a name for the profile and a description off the profile. Select a type of **PPP** and a mode of **Switched line-dial**. Click **Connection**. The display shown in Figure 230 appears.

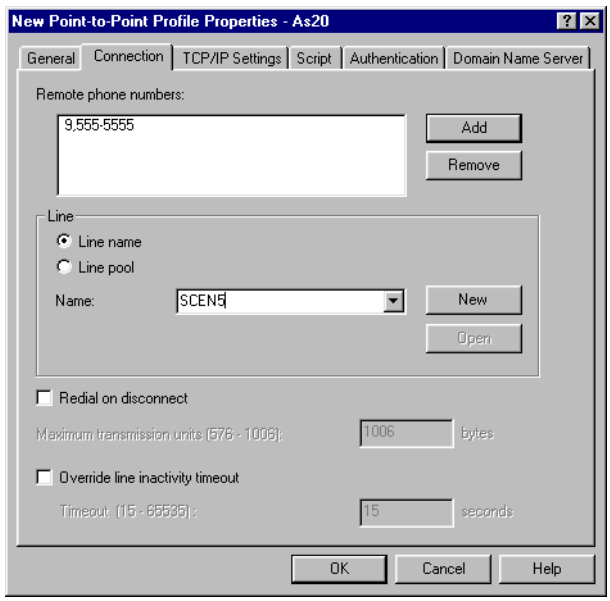


Figure 230. Entering the remote phone number and creating a new PPP line

5. Click **Add**. The input area for the phone number is made available for input. Type the phone number to dial to reach the remote system. Be sure to include any prefix that may be required. Select the **Line Name** field. Type the name of the new line you are creating or select an existing line to use. To create a new line, click **New**. The display shown in Figure 231 appears.

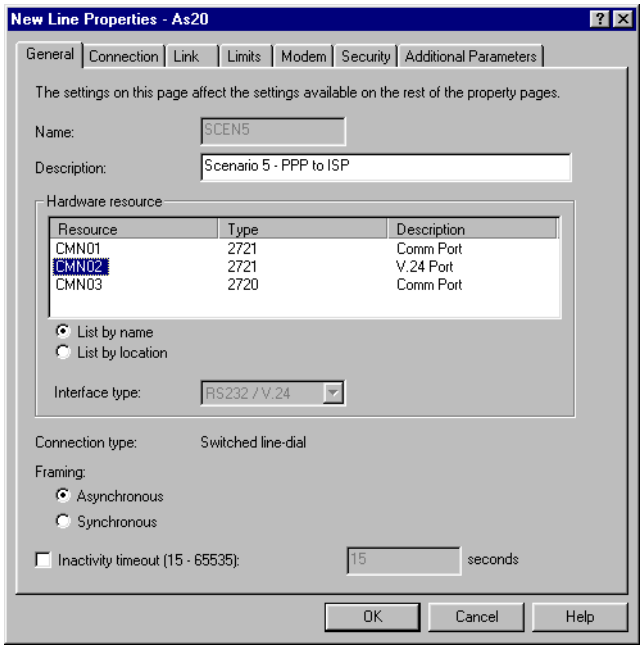


Figure 231. Selecting the hardware adapter you are using

6. Select the appropriate hardware adapter. Click **Modem**. The display shown in Figure 232 appears.

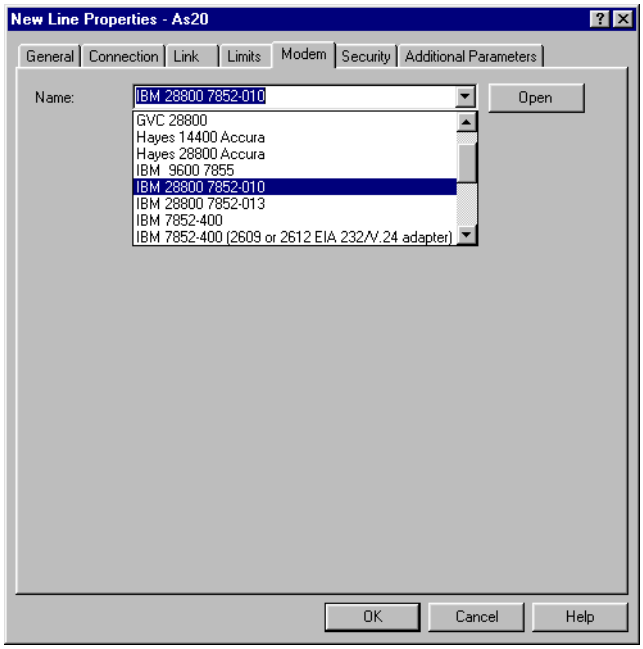


Figure 232. Selecting the appropriate modem

7. Select the modem you are using from the list shown. Click **OK**. The display shown in Figure 230 appears. Click **TCP/IP Settings**. The display shown in Figure 233 on page 162 appears.

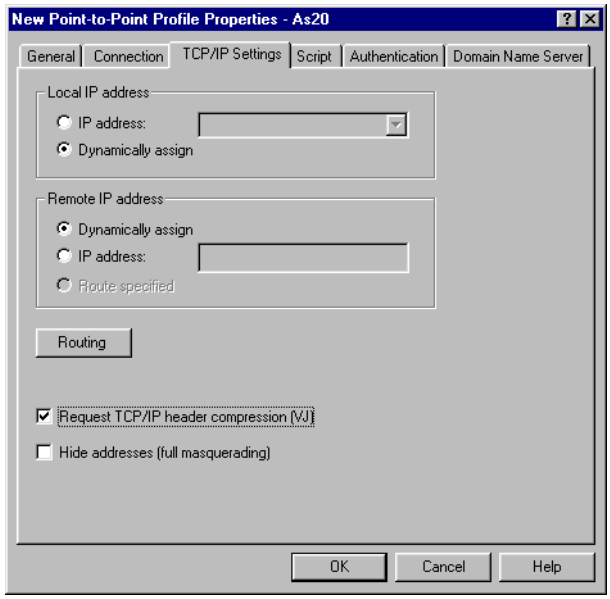


Figure 233. Specifying TCP/IP settings for the PPP connection to the ISP

8. Select **Dynamically assign** in both the Local IP address and the Remote IP address section of the window. Select the compression to speed up your connection. Click **Routing** to specify routing information. The display shown in Figure 234 appears.

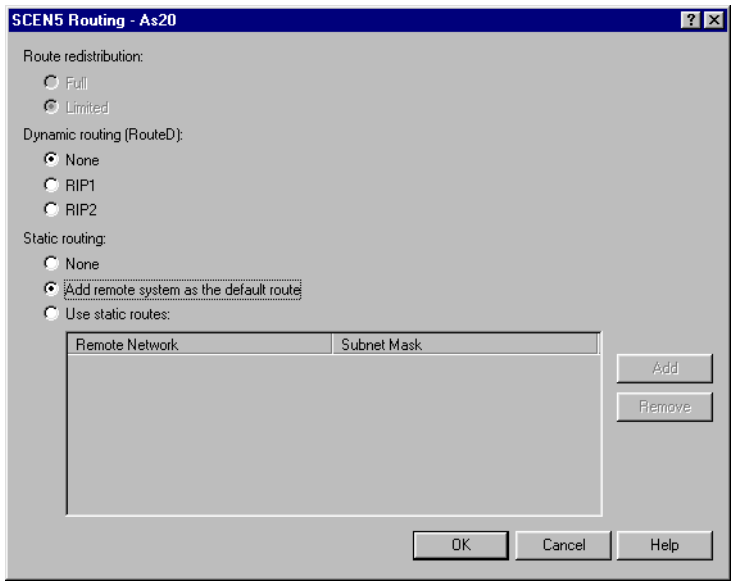


Figure 234. Adding the ISP as the default route

9. Select **Add remote system as the default route**. This makes all the traffic that is not in a directly connected LAN segment flow to the ISP. Click **OK** to end the routing configuration. The display shown in Figure 233 appears. Click **Authentication**. The display shown in Figure 235 appears.

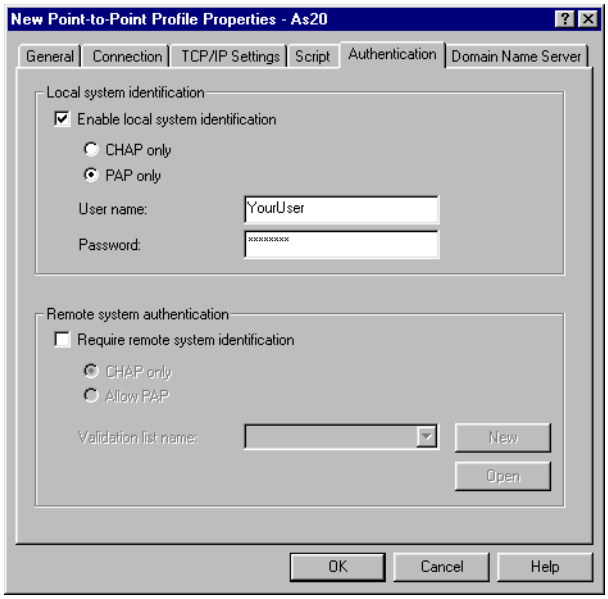


Figure 235. Specifying the user and password information

10. Select the correct Local system identification type. We are using PAP authentication. Specify the User name and Password information provided by the ISP. Click **Domain Name Server**. The display shown in Figure 236 appears.

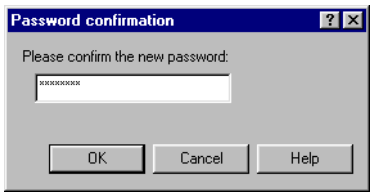


Figure 236. Confirming the entered password

11. Type the password again to verify that it is entered correctly. Click **OK**. The display shown in Figure 237 on page 164 appears.

> **Note**
>
> If you click OK rather than Domain Name Server, you will be returned to the Operations Navigator main window. To continue with the configuration of the PPP profile, right-click the PPP profile name, and select **Properties** from the pop-up menu.

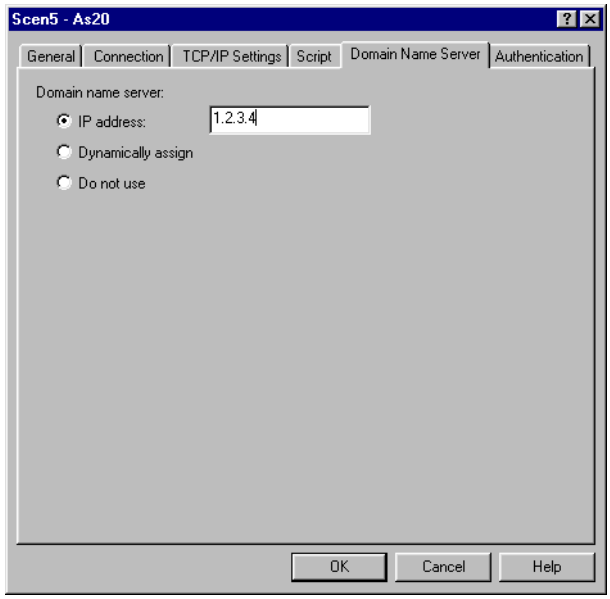


Figure 237. Configuring the DNS server to use

12. Click **IP address**, and enter the DNS information provided by the ISP. Click **OK**. The display shown in Figure 238 appears.

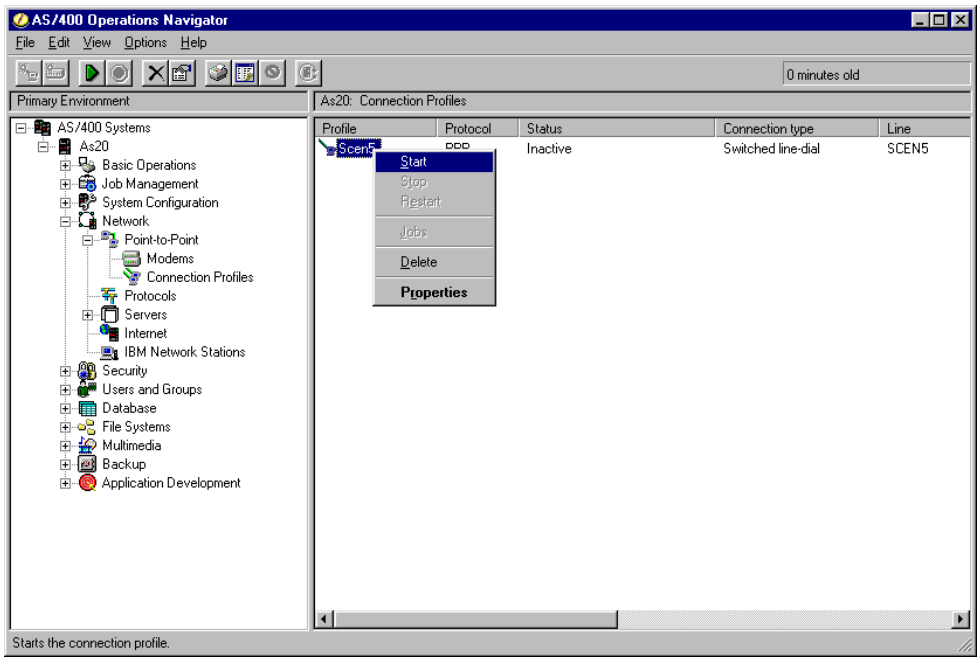


Figure 238. Starting the PPP connection to the ISP

13. Right-click the profile name, and select **Start**. When the status changes to `Active`, you are connected to the Internet. Use F5 (Refresh) to update the display.

4.13.3 Testing the scenario

The test of the scenario is shown in the following screens. The test is performed by using Telnet and FTP. First, we used Telnet to connect to the Library of Congress as shown in Figure 239 and Figure 240. This makes a TN3270 connection to the system on the Internet.

```
MAIN                          AS/400 Main Menu
                                                          System:   AS20
 Select one of the following:

      1. User tasks
      2. Office tasks
      3. General system tasks
      4. Files, libraries, and folders
      5. Programming
      6. Communications
      7. Define or change the system
      8. Problem handling
      9. Display a menu
     10. Information Assistant options
     11. Client Access/400 tasks

     90. Sign off

 Selection or command
 ===> telnet locis.loc.gov

 F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=Information Assistant
 F23=Set initial menu
```

Figure 239. Using Telnet to test the connection (Part 1)

```
        L O C I S :  LIBRARY OF CONGRESS INFORMATION SYSTEM

         To make a choice: type a number, then press ENTER

  1   Library of Congress Catalog          4   Braille and Audio

  2   Federal Legislation                  5   Foreign Law

  3   Copyright Information

*     *     *     *     *     *     *     *     *     *     *     *

  7   Searching Hours and Basic Search Commands
  8   Documentation and Classes
  9   Library of Congress General Information
 10   Library of Congress Fast Facts
 11   * * Announcements * *

      The Organizations (NRCM) file is no longer created or supported by LC.
      It has been removed from LOCIS.

 12   Comments and Logoff
      Choice:
                                                                LOCISMENU
```

Figure 240. Using Telnet to test the connection (Part 2)

Next, we used FTP to connect to the public IBM FTP server as shown in Figure 241 and Figure 242. We used the user “anonymous” and a password of our e-mail address to logon.

```
MAIN                          AS/400 Main Menu
                                                              System:   AS20
 Select one of the following:

      1. User tasks
      2. Office tasks
      3. General system tasks
      4. Files, libraries, and folders
      5. Programming
      6. Communications
      7. Define or change the system
      8. Problem handling
      9. Display a menu
     10. Information Assistant options
     11. Client Access/400 tasks

     90. Sign off

 Selection or command
 ===> ftp ftp.software.ibm.com

 F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=Information Assistant
 F23=Set initial menu
```

Figure 241. Using FTP to test the connection to the ISP (Part 1)

```
                          File Transfer Protocol

 Previous FTP subcommands and messages:
   220-*
                                                                          *
   220-*  All FTP'able software is (c) copyright International Business
            *
   220-*  Machines Corporation.  All rights reserved.  Logging is performed on
     *
   220-*  all activity.       ******* LOGON AS "anonymous" *******
                *
   220-*************************************************************************
    **
   220-
   220 service.boulder.ibm.com FTP server (Version wu-2.4.2-academ[BETA-14](1)
    Wed Aug 20 10:20:24 MDT 1997) ready.
 Enter login ID:
 ===>



 F3=Exit     F6=Print       F9=Retrieve
 F17=Top    F18=Bottom    F21=CL command line
```

Figure 242. Using FTP to test the connection to the ISP (Part 2)

4.14 Scenario 6: AS/400 dial to AS/400 answer—SLIP

In this scenario (Figure 111 on page 102), we show the steps used to communicate between two AS/400 systems using TCP/IP SLIP. The connection must be manually started before communications can occur. To make this work, you accomplish the following tasks:

- Configure a SLIP switched answer connection profile on AS08
- Configure a SLIP dial connection profile on AS23

Refer to Chapter 2, “TCP/IP basic installation and configuration” on page 7, to create the basic setup of the IP interfaces on the AS/400 systems. We used the following IP address on the AS/400 system:

- AS08: 10.1.2.1/24 on the 10.1.2.0/24 network
- AS23: 10.1.1.1/24 on the 10.1.1.0/24 network

This scenario focuses on using Operations Navigator for the configuration process. The scenario is shown using the graphical interface. This interface is preferable to the green-screen interfaces because it is much easier and intuitive. If you cannot use Operations Navigator to configure the SLIP connection due to hardware limitations (see 4.3.2, “Hardware requirements” on page 90), you have to use the green-screen interface to configure the connection. An example of this is shown in 4.15, “Scenario 7: Windows 9x PC dial to AS/400 answer—SLIP” on page 180.

4.14.1 Configuring the AS08 system answer SLIP connection

First, we configure the AS08 system to accept the incoming call. Use the following steps to start configuration of the SLIP connection:

1. Use the procedure in 4.3.3, “Starting Operations Navigator for PPP configuration” on page 90, to access the PPP configuration tree.
2. Click the **Connection Profiles** item. The available profiles appear in the right window.
3. Right-click **Connection Profiles** to show the menu. Select **New Profile**. The New Point-to-Point Profile Properties window shown in Figure 243 on page 168 appears.

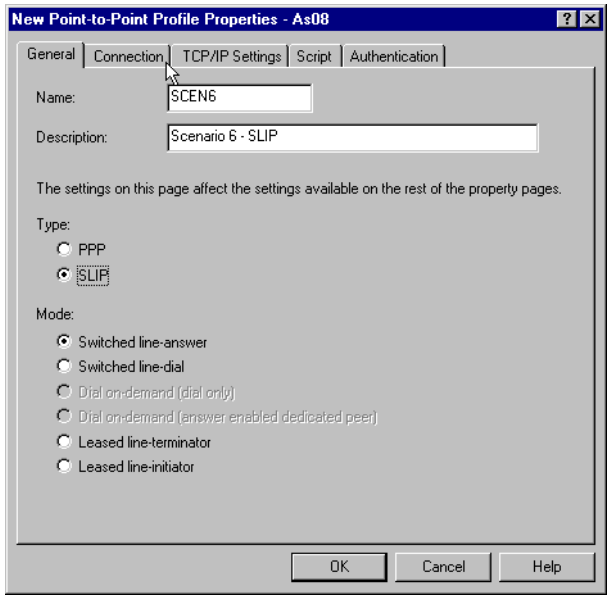


Figure 243. Specifying SLIP and answer mode

4. Enter a name for the profile and a description. Select a type of **SLIP** and a mode of **Switched line-answer**. Click **Connection**. The display shown in Figure 244 appears.

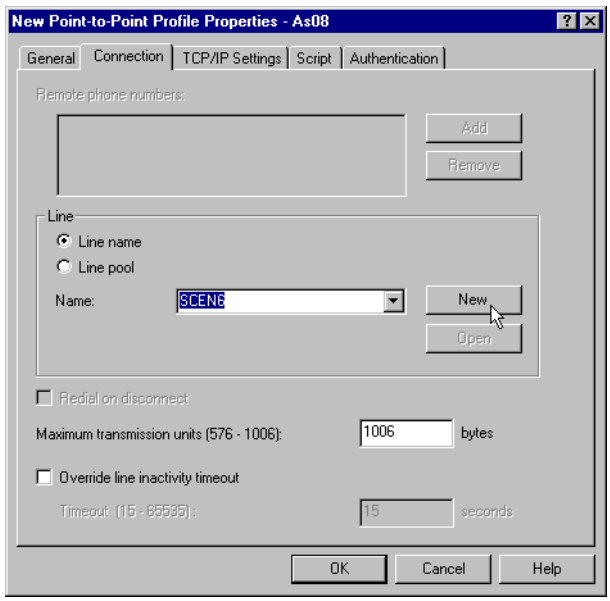


Figure 244. Creating a new line for the SLIP connection

5. Type a line name in the Name field, and click **New** to create a new line for the connection. The display shown in Figure 245 appears.

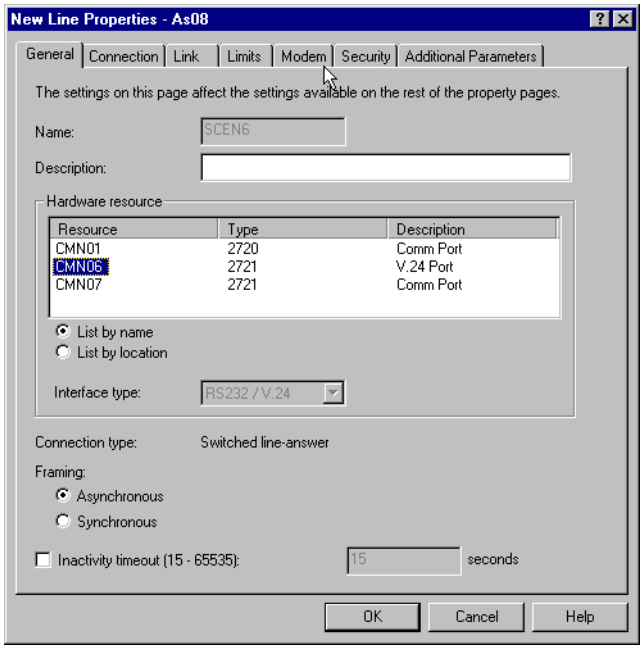


Figure 245. Selecting the hardware adapter for the SLIP connection

6. Select the appropriate hardware adapter. Click **Modem**. The display shown in Figure 246 appears.

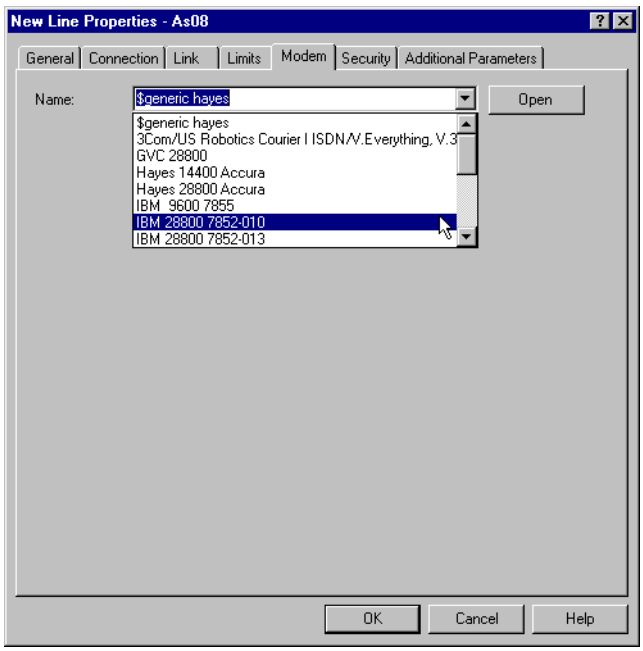


Figure 246. Selecting the modem type for the SLIP connection

7. Select the modem you are using from the list shown. Click **OK**. The display shown in Figure 244 on page 168 appears. Click **TCP/IP Settings**. The display shown in Figure 247 on page 170 appears.

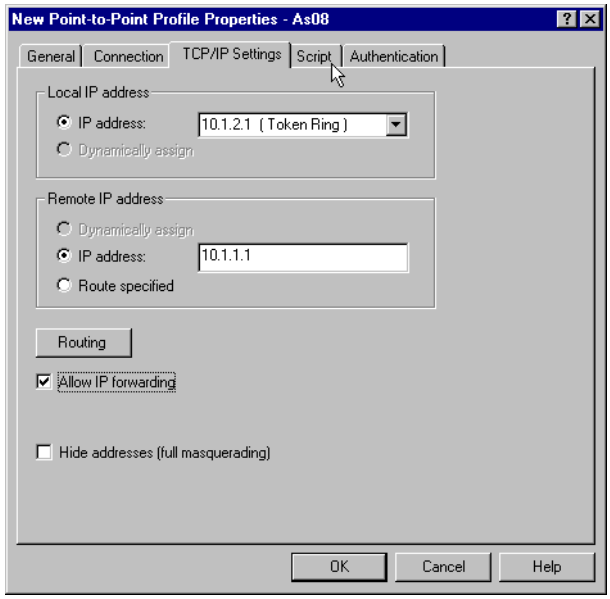


Figure 247. Specifying the TCP/IP local and remote settings

8. In this example, we set up an unnumbered network. We specify Allow IP forwarding to enable other systems in the network to use AS08 and AS23 as a router. To complete this connection, AS23 must also be configured to allow the connecting back to the entire 10.1.2 network. For the local IP address, we use the address of the AS/400 LAN adapter 10.1.2.1. For the remote address, we use an address of 10.1.1.1. Click **Script**. The display shown in Figure 248 appears.

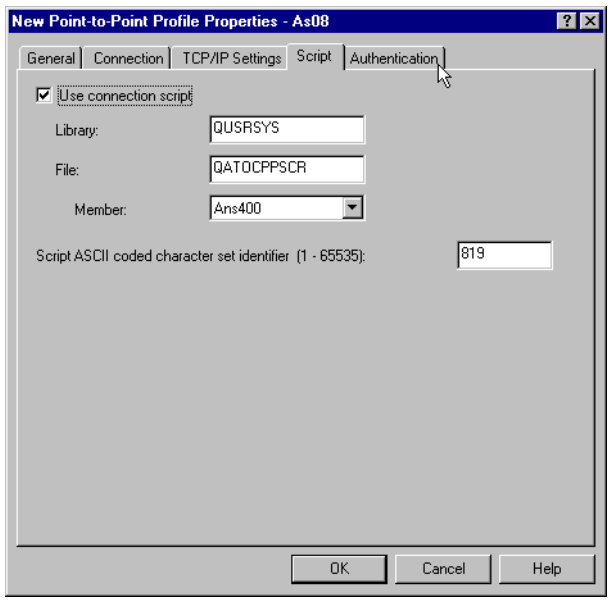


Figure 248. Selecting the use of a script when using SLIP

9. This is one place where PPP is different from SLIP. With SLIP, you use a script to pass information to the remote system. This information is required to establish the connection. The system is shipped with several standard default scripts. In this case, we select **Use connection script**, and specify the

Member for answering an AS/400 connection (**Ans400**). Click the **Authentication** tab. The display shown in Figure 249 appears.

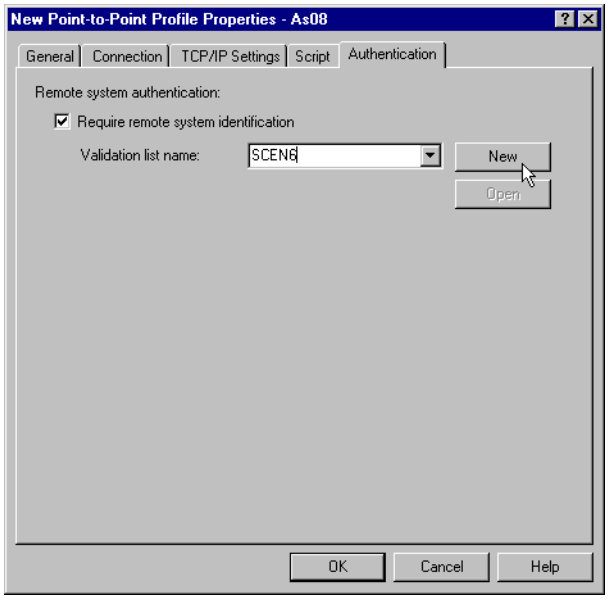


Figure 249. Specifying security

10. Check **Require remote system identification**. Enter a validation list name, and click **New** to create a new validation list for the remote users. You may use an existing validation list from the drop-down list rather than creating a new list. The display shown in Figure 250 appears.

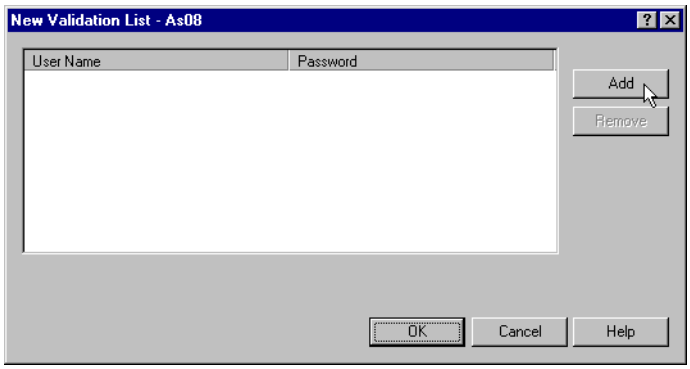


Figure 250. Adding a new user to the validation list

11. Click **Add** to add a new user to the validation list. The display shown in Figure 251 on page 172 appears.

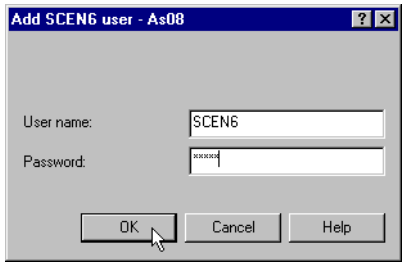


Figure 251. Entering a user ID and password for the SLIP connection

12. Specify the user name and the password. Click **OK**. The display shown in Figure 252 appears.

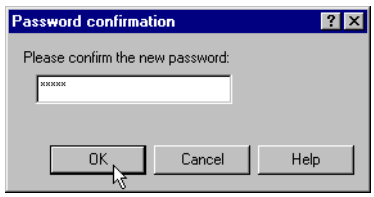


Figure 252. Confirming the password

13. Enter the password again to confirm the value entered. Click **OK**. The display shown in Figure 253 appears. Notice that the validation list has been added.

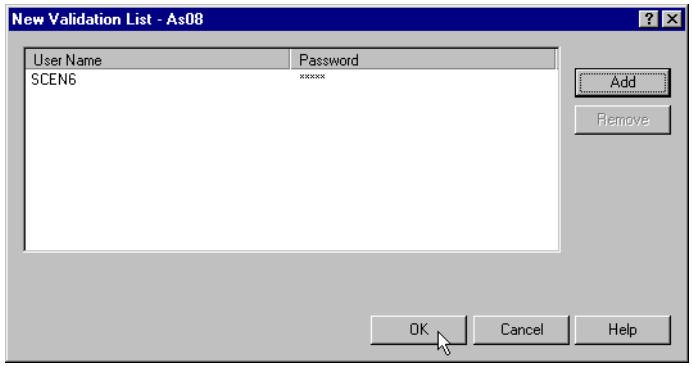


Figure 253. Confirming the users in the validation list

14. Click **OK** to confirm the new validation list. The display shown in Figure 254 appears.

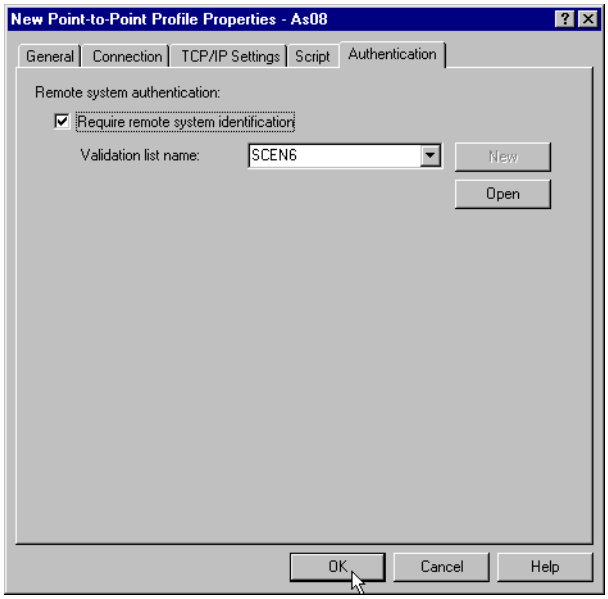


Figure 254. Confirming the creation of the SLIP connection on AS08

15. Click **OK** to confirm the creation of the new profile. The display shown in Figure 255 appears. Notice that your profile has been added in the right-hand window.

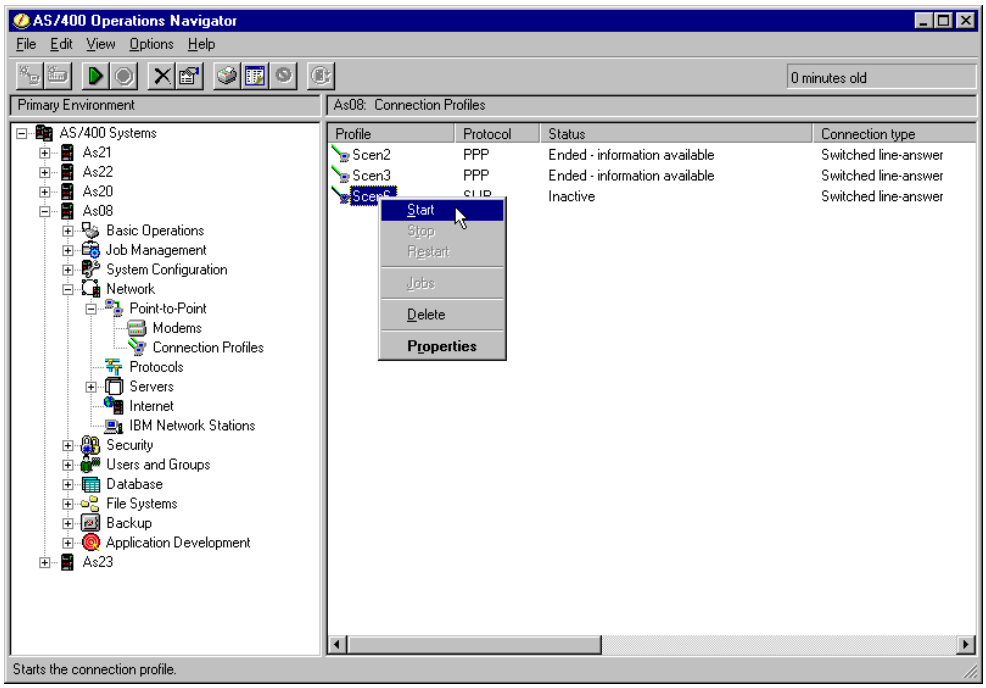


Figure 255. Starting the SLIP connection on AS08

16. To start the connection profile, right-click on the profile name. Select **Start** from the pop-up menu. The status changes to `Waiting for incoming call` as shown in Figure 256 on page 174. You may need to refresh the window contents by pressing F5.

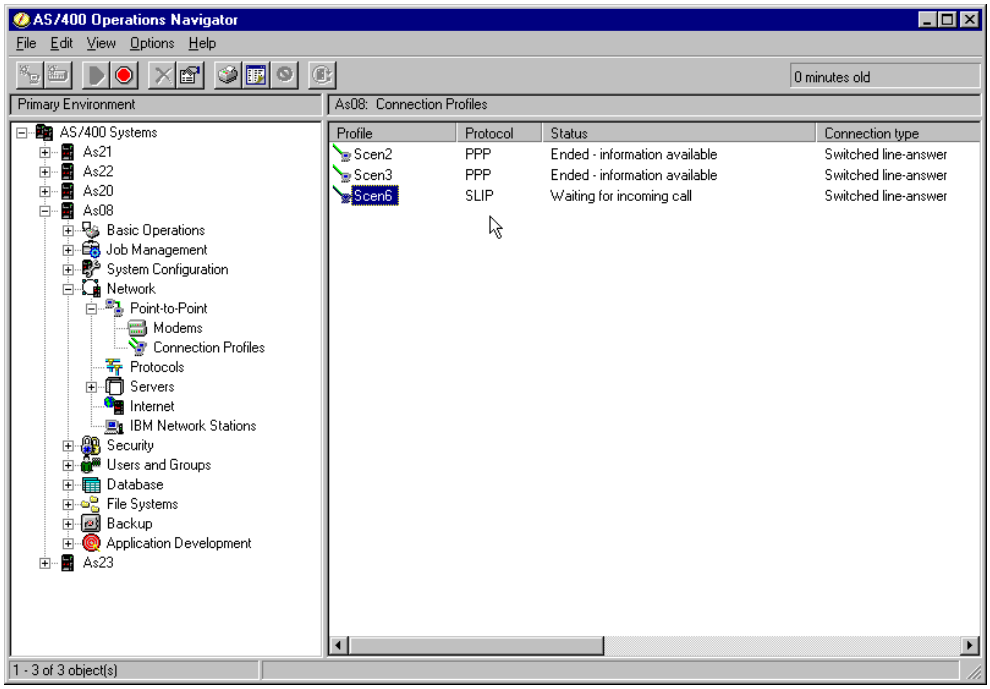


Figure 256. The SLIP connection on AS08 is now ready

The AS/400 system is ready for the SLIP connection.

4.14.2 Configuring the AS23 system dial SLIP connection

In this section, we configure the AS23 system to place the outgoing call. Use the following steps to configure the SLIP connection:

1. Use the procedure in 4.3.3, “Starting Operations Navigator for PPP configuration” on page 90, to access the PPP configuration tree.
2. Click the **Connection Profiles** item. The available profiles appear in the right window.
3. Right-click **Connection Profiles** to show the menu. Select **New Profile**. The New Point-to-Point Profile Properties window shown in Figure 257 appears.

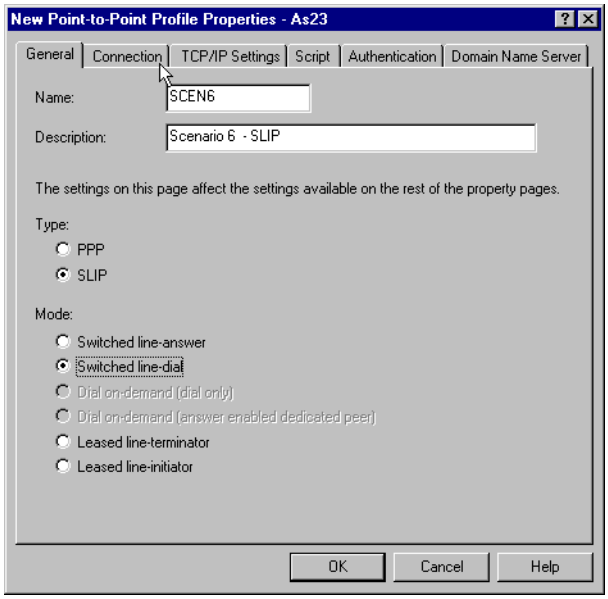


Figure 257. Selecting the SLIP type and dial-mode

4. Enter a name for the profile and a description. Select a type of **SLIP** and a mode of **Switched line-dial**. Click **Connection**. The display shown in Figure 258 appears.

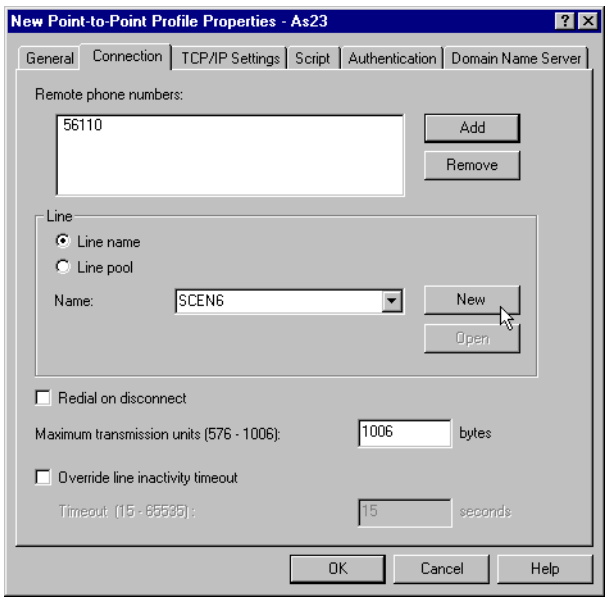


Figure 258. Specifying the remote number to call

5. Click **Add**. The input area for the phone number is made available for input. Type the phone number to dial to reach the remote system. Be sure to include any prefix that may be required. Select the **Line Name** field. Type the name of the new line you are creating or select an existing line to use. To create a new line, click **New**. The display shown in Figure 259 on page 176 appears.

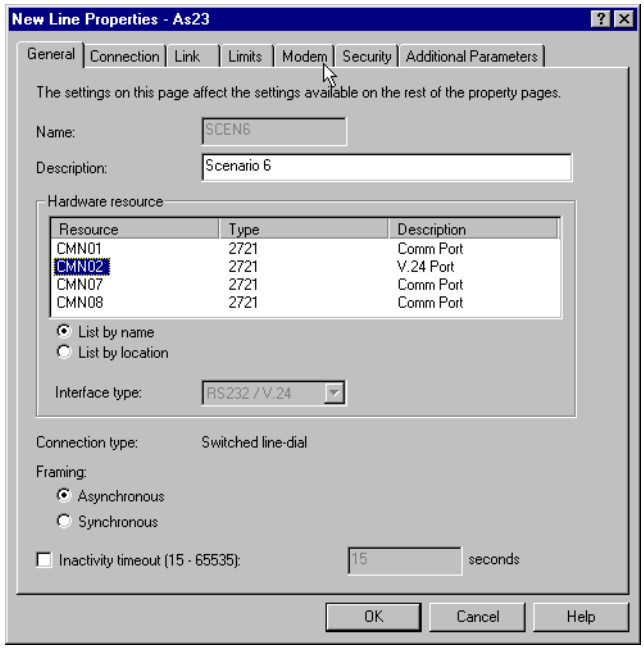


Figure 259. Selecting the hardware adapter to use

6. Select the appropriate hardware adapter. Click the **Modem** tab. The display shown in Figure 260 appears.

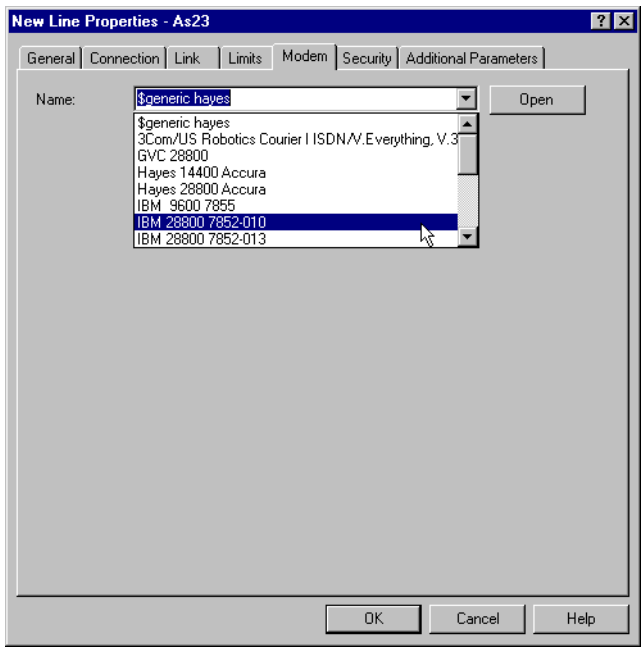


Figure 260. Selecting the correct modem type

7. Select the modem you are using from the list shown. Click **OK**. The display shown in Figure 258 on page 175 appears. Click **TCP/IP Settings**. The display shown in Figure 261 appears.

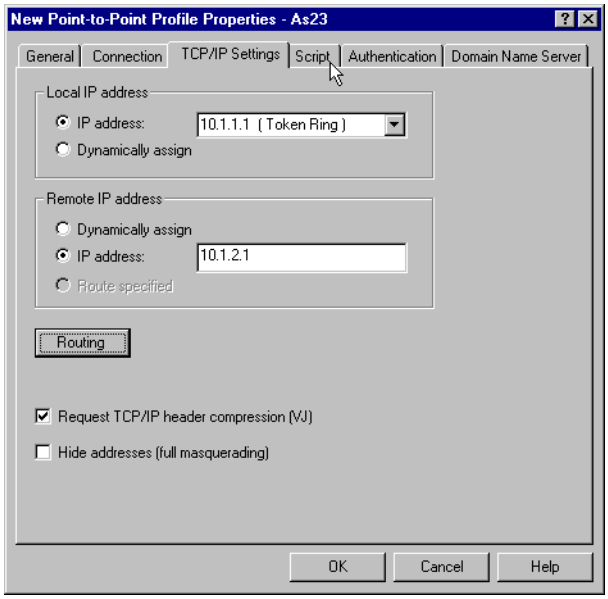


Figure 261. Configuring the TCP/IP settings for the SLIP connection

8. In this example, we set up an unnumbered network. For the local IP address, we use the address of the AS/400 LAN adapter (10.1.1.1). For the remote address, we use an address of 10.1.2.1. The IP addresses are flipped in relation to step 8 on page 170 . Here, the local IP address is the same as the remote IP address in step 8 on page 170 and the remote IP address is the same as the local IP address in step 8 on page 170. Select **Request TCP/IP header compression (VJ)** to speed up your connection. Click **Script**. The display shown in Figure 262 appears.

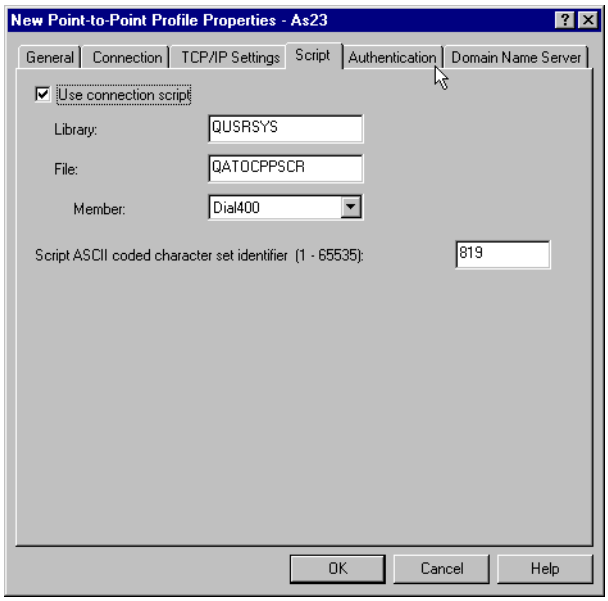


Figure 262. Selecting the script to use for the SLIP connection

9. This is one place where PPP is different from SLIP. With SLIP, you use a script to pass information to the remote system. This information is required to

establish the connection. The system is shipped with several standard default scripts. In this case, we select **Use connection script**, and specify the member for dialing a AS/400 connection (Dial400). Click the **Authentication** tab. The display shown in Figure 263 appears.

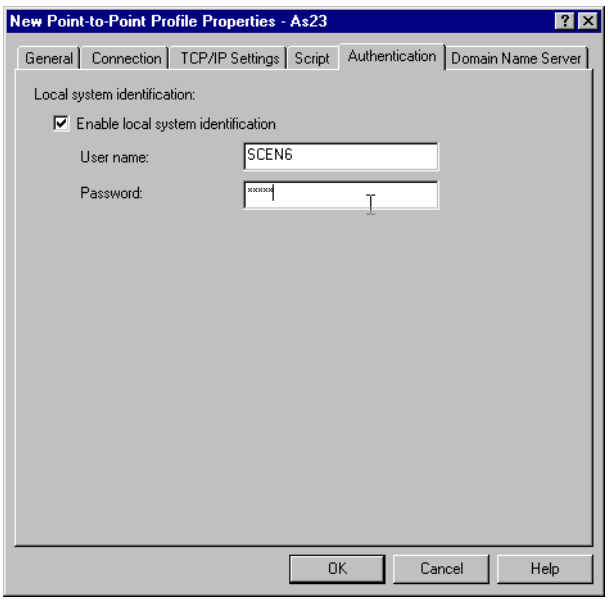


Figure 263. Identifying the user to be used on the SLIP connection

10. Check **Enable local system identification**. Type a user name and password. This value must match the value specified in step 12 on page 172. The display shown in Figure 264 appears.

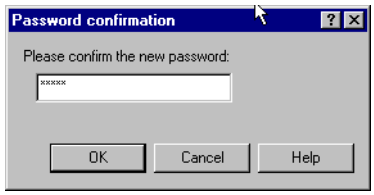


Figure 264. Confirming the password

11. Enter the password again to confirm the value entered. Click **OK**. The display shown in Figure 265 appears. Notice that your profile has been added in the right-hand window.

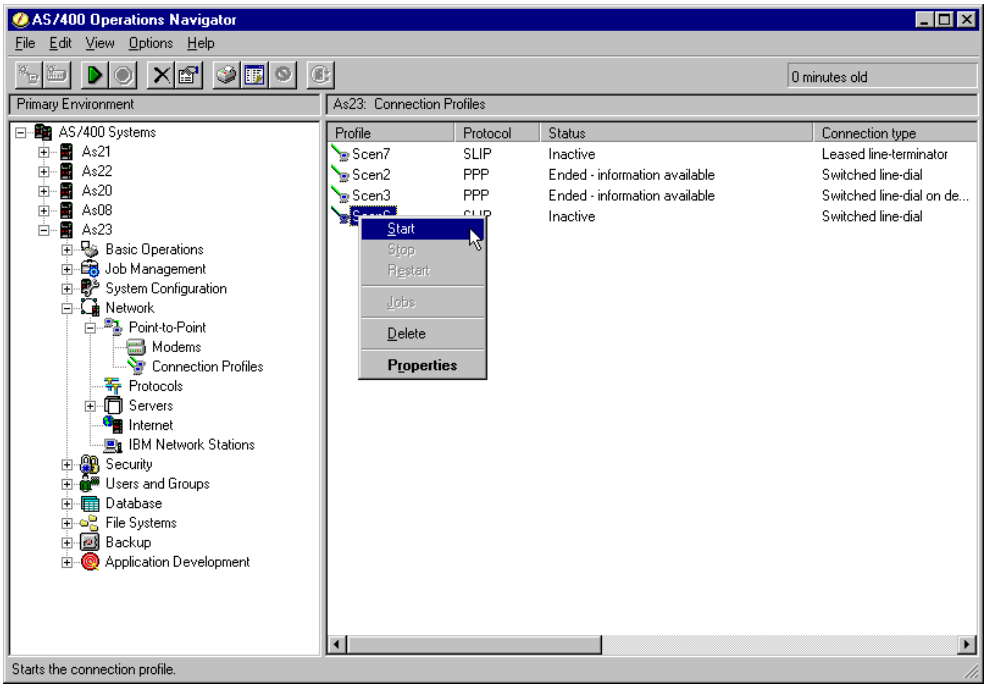


Figure 265. Starting the SLIP connection on the AS23 system

12. To start the connection profile, right-click on the profile name. Select **Start** from the pop-up menu. If you are near the modem, and the modem speaker is on, you may hear a dial tone. Monitor the call progress by using the F5 key to refresh the window contents. After some time, you should see that the status changes to `Active` as seen in Figure 266.

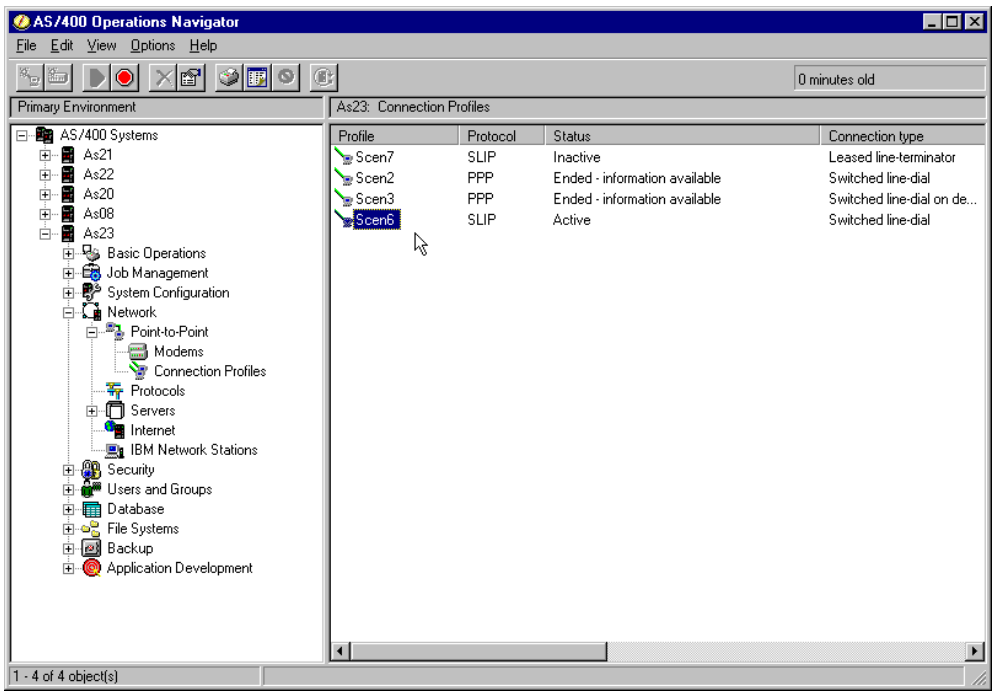


Figure 266. The ready-to-use SLIP connection on the AS23 system

The connection is now ready to use.

4.14.3 Testing the scenario

The test of the scenario is performed by using of the Start TCP/IP TELNET (TELNET) command. To perform the test, complete the following steps:

1. Sign on to the AS23 system.
2. On the AS/400 command line, issue the command:

   ```
   TELNET RMTSYS('10.1.2.1')
   ```

   In this example, 10.1.2.1 is the address of the remote system (AS08) that we want to access. Press Enter. If the connection is working, you should be presented with a sign on screen from the remote system (Figure 267).

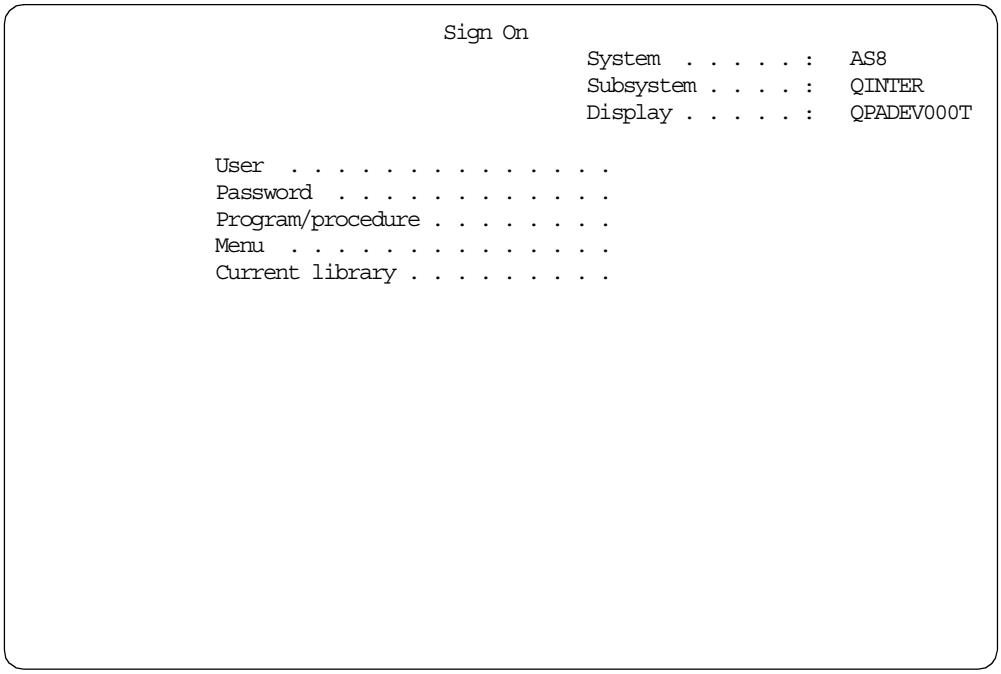


Figure 267. The connection to the AS08 system is successful

3. Sign on, and test the link. After you complete the test, sign off, and end the Telnet session.

4.15 Scenario 7: Windows 9x PC dial to AS/400 answer—SLIP

In this scenario (Figure 112 on page 102), we show the steps used to communicate between an AS/400 system and a Windows 9x PC using TCP/IP SLIP. The connection must be manually started before communications can occur. To make this work, you accomplish the following tasks:

- Configure a SLIP switched answer connection profile on AS23.
- Configure a SLIP dial-up-networking connection on the PC.

Refer to Chapter 2, “TCP/IP basic installation and configuration” on page 7, to create the basic setup of the IP interfaces on the AS/400 systems. We used the IP address AS23: 10.1.1.1/24, on the 10.1.1.0/24 network on the AS/400 system.

The scenario is shown using the AS/400 command line interface. This interface can be used if you cannot use Operations Navigator to configure the SLIP connection due to hardware limitations (see 4.3.2, “Hardware requirements” on page 90).

4.15.1 Configuring the SLIP connection at the AS23 system

The task is shown using the AS/400 command line interface. It can also be accomplished by using the Operations Navigator interface. The following objects are needed on the AS/400 system for a SLIP connection:

- An asynchronous line description
- An authorization list
- A point to point connection profile with a type of *SLIP
- A modem definition
- A connection script (a default script may work)

Perform the following procedure to create the required objects on the AS/400 system:

1. Use the Create Line Desc (Async) (CRTLINASC) command to create the SLIP line:

   ```
   CRTLINASC LIND(SCEN7) RSRCNAME(CMN01) CNN(*SWTPP)   LINESPEED(19200)
   SWTCNN(*DIAL) AUTOANS(*NO) AUTODIAL(*YES)  DIALCMD(*OTHER) INACTTMR(*NOMAX)
   MAXBUFFER(1500)
   ```

   If the Resource name is different on your system, use the Work with Hardware Resources (WRKHDWRSC) command to locate the correct resource name.

2. Create an equalization list. Only the user profiles specified in the authorization list are allowed to connect to this AS/400 system from a remote system. The authorization list is created by using option 1 of the Work with Authorization Lists (WRKAUTL) command (Figure 268 on page 182). Add one or more AS/400 user profile names that should be authorized to connect to the list using option 2 (Edit), and then press F6 (add new user). Remember to create the user profile before you edit the authorization list.

```
                        Work with Authorization Lists

Type options, press Enter.
  1=Create   2=Edit   4=Delete          5=Display   8=Display objects in list
  9=Display documents/folders in list   13=Change description

Opt  List        Text
1    SCEN7
     QIWSADM     Client Access/400 Administrators
     QOPTSEC     Default Optical Authorization List
     QPWFSERVER




                                                                        Bottom
Parameters for options 1, 5, 8, 9 and 13 or command
===>
F3=Exit      F4=Prompt   F5=Refresh   F9=Retrieve   F11=Display names only
F12=Cancel   F16=Repeat position to   F17=Position to
```

Figure 268. Creating an authorization list for the SLIP connection

3. To set up the AS/400 SLIP configuration, use the Work with Point-to-Point TCP/IP (`WRKTCPPTP`) command to define a point-to-point connection profile of type answer. Type the following command on any AS/400 command line:

   ```
   WRKTCPPTP
   ```

   Press Enter. The screen in Figure 269 appears.

```
                        Work with Point-to-Point TCP/IP

Type option, press Enter.
  1=Add     2=Change   3=Copy   4=Remove          5=Display details   6=Print
  9=Start   10=End     12=Work with line status   14=Work with job

                                                 Line      Line    Job
Opt  Name        Mode   Type   Status          Description  Type    Name
1    SCEN7       *ANS

  (No configuration profiles)









                                                                        Bottom
F8=Work with modems   F9=Command line   F10=Local interface status
F11=Display text      F12=Cancel   F14=Work with active jobs   F24=More keys
```

Figure 269. Creating the SLIP profile

4. Enter option `1` (Add), enter a name for the new configuration profile (in our example, it is SCEN7), and then type `*ANS` as mode. Press Enter. You see the prompt shown in Figure 270.

```
                  Add TCP/IP Point-to-Point *ANS Profile
                                                          System:   AS23
Name:   SCEN7
Text

Type choices, press Enter.

TCP/IP information:
  Protocol type . . . . . . . . . . :   *SLIP
  Local interface address . . . . . .   10.1.1.1          Address, F4 for list
  Remote IP address . . . . . . . . .   10.1.1.200        Address
  Maximum transmission unit . . . . .   1006              576-1006
  Allow proxy ARP . . . . . . . . . .   N                 Y=Yes, N=No
  Add default route . . . . . . . . .   N                 Y=Yes, N=No

Physical line information:
  Line description  . . . . . . . . .   SCEN7             Name
  Line type . . . . . . . . . . . . :   *ASYNC
  Autocreate controller and device      Y                 Y=Yes, N=No
    Remote location name  . . . . . .                     Name
                                                                    More...
F2=Change modem information   F3=Exit   F4=List   F9=Command line
F12=Cancel
```

Figure 270. Creating the SLIP answer profile at AS23

- The local interface address is the IP address of the AS/400 server you will contact with this SLIP connection. This value is the local IP address to use as the gateway address for the remote clients. It is used as a next hop value for a route or default route to the AS/400 system. Press F4 to get a list of defined local addresses to use, or enter a new address. If a defined local address is chosen, it can be used for Proxy ARP on behalf of the remote system dialing in. The Allow proxy ARP parameter has to be set to `Y` for this to occur.
- The remote IP address is the address that the remote client will use for this SLIP connection. This is the address that the remote client should use as its local interface address. It is the IP address used to allow the remote system and the local AS/400 system to communicate. If the local IP address chosen already exists and is being used for Proxy ARP, the remote IP address that you choose must be defined on the same subnet as the local IP address defined by its subnet mask.
- Maximum transmission unit should be set to 1006. The MAXBUFFER parameter of the Create Asynchronous Line Description (CRTLINASC) command can be used to specify the maximum size of the line's inbound and outbound data buffers. The default for the MAXBUFFER parameter is 896 bytes. The value specified for the SLIP MTU must be less than or equal to the value of MAXBUFFER. In this example, we use 1500 bytes for MAXBUFFER.
- The default for the Allow proxy ARP parameter is “N”. Change it to `Y` if proxy ARP is to be used. This field can only be set to “Y” if the Local interface address defined is a true local interface that is already defined

and the Remote IP address is defined to be on the same subnet as the local address.

- Line description is the name of the asynchronous line description created in step 1 on page 181. A valid line description needs to be entered here. Only lines of type *ASYNC are supported.
- Y is usually a desirable option for the Autocreate controller and device parameter. Then, TCP/IP creates the appropriate controller and device for the session. When the session is completed, the automatically created controller and device are deleted.
- Press Page Down (PgDn) to see the second panel (Figure 271) of the *ANS profile configuration.

```
                  Add TCP/IP Point-to-Point *ANS Profile
                                                          System:   AS23
Name:   SCEN7
Text

Type choices, press Enter.

Modem information:
  Use a modem . . . . . .   Y                Y=Yes, N=No
    Modem information name   IBM 28800 7852-010          F4 for list


Script source information:
  Use connection dialog script  . . .   Y                Y=Yes, N=No
    Member  . . . . . .   ANS400           Name
    File  . . . . . .   QATOCPPSCR         Name
      Library . . . . . .     QUSRSYS       Name
    ASCII character set identifier      00819            1-65533, *DFT


                                                                   More...
F2=Change modem information   F3=Exit   F4=List   F9=Command line
F12=Cancel
```

Figure 271. Entering the modem and script parameters

- For the Use a modem parameter, we selected “Y” (Yes).
- When you add a new profile, the field Modem information name parameter is initially blank. If you specify a “Y” in the field, use a modem. You must enter a valid value for Modem information name parameter. Press F4 to display a pop-up selection list of predefined modem strings.
- For the Use connection dialog script parameter, we specified “Y” (Yes). For this example, we used script because we wanted an exchange of user ID, password, and IP addresses.
- Member/file/library indicates the location of the connection script to use to allow remote systems to dial into the AS/400 system. This script is used by the server. The remote system that dials in must have a compatible script. See 4.15.2, “Creating and changing connection scripts” on page 186, for information on how to use customized scripts.
- For the ASCII character set identifier parameter, the ASCII CCSID is used to translate ASCII to EBCDIC and EBCDIC to ASCII connection script data.

- Press Page Down (PgDn) again to see the third configuration display (Figure 272).

```
                    Add TCP/IP Point-to-Point *ANS Profile
                                                              System:   AS23
Name:   SCEN7
Text

Type choices, press Enter.

Local system security:
  Allow IP datagram forwarding  . . .   N                Y=Yes, N=No
  System access authorization list      SCEN7            *NONE, Name









                                                                      Bottom
F2=Change modem information   F3=Exit   F4=List   F9=Command line
F12=Cancel
```

*Figure 272. Add TCP/IP Point-to-Point *ANS Profile*

- When a remote client connects to the AS/400 system, the value for Allow IP datagram forwarding determines whether the TCP/IP stack allows IP datagrams originating from the remote host to be forwarded on to IP addresses other than the local IP address defined for the AS/400 system for this connection. If IP datagram forwarding is set to OFF at the system level through the IPDTGFWD parameter on the CHGTCPA command, the value for IP datagram forwarding on any of the SLIP profiles has no effect since IP datagram forwarding is not allowed for any TCP/IP interface.
- If an authorization list is specified, only the user profiles specified in the authorization list are allowed to connect to this AS/400 system from a remote system. The option *NONE means that when a client is trying to connect to the AS/400 system, no user ID or password needs to be specified. If there is no authorization list (that is, you specified *NONE), it only means that there is no connection security. If the client passes a user ID and password, these values are ignored and the connection is allowed. However, there is still application level security, such as the Telnet user ID and password or, as in our case, a Notes ID that is needed. If the user wants to validate that the remote client is allowed to connect, a valid system authorization list name can be entered here. The AS/400 user profiles may already exist. If not, you can create them using the Create User Profile (CRTUSRPRF) command before you add their names to the authorization list. The password included with the connection dialog must match the password specified for the AS/400 user profile.

4.15.2 Creating and changing connection scripts

You cannot change the content of the default connection script file QATOCPPSCR in Library QUSRSYS. If you want to use a modified script, you must first create your own connection script file. Do this by copying the default file using the Copy File (CPYF) command as follows:

```
CPYF FROMFILE(QUSRSYS/QATOCPPSCR) TOFILE(lib/file)
FROMMBR(*ALL) TOMBR(*FROMMBR) MBROPT(*ADD) CRTFILE(*YES)
```

In this example, `lib/file` represents your own new file. Now you may modify the copy of the script file. Figure 273 shows an example of a script file member.

```
************************************************************************
*SERVER CONNECTION SCRIPT EXAMPLE WITH LOGIN AND PASSWORD / WIN95
(PROMPT)
& Userid:
(USERID)
& Password?
(PASSWORD)
& InternetLR/E>
(PROMPT)
& (IPGATE) IS AS/400 IP ADDRESS.
& (IPADDR) IS IP ADDRESS OF SYSTEM CALLING AS/400.
* END OF SERVER CONNECTION SCRIPT EXAMPLE
************************************************************************
```

Figure 273. Server connection script example

You should use the SEU editor to make your required changes to the file member. The changes needed depend on the requirements specified by the connecting system.

4.15.3 Adding SLIP support to the Windows 9x system

Additional software may need to be downloaded from the Microsoft home page and installed on your workstation. The following procedure details the installation instructions:

1. Create a folder called SLIP on your PC.
2. Using a Web browser, go to the site: `http://www.microsoft.com/`

   Use the search function, and find `dscrpt.exe`.
3. Download the file `dscrpt.exe` to the SLIP folder.
4. Run dscrpt.exe, which is a self-extracting file.
5. Double-click the **My Computer** icon.
6. Double-click the **Control Panel** icon .
7. Double-click the **Add/Remove Programs** icon .
8. Click the **Windows Setup** tab.
9. Click **Have Disk**.
10. Click **Browse**.
11. Select the SLIP folder from the directory tree. RNAPLUS.INF should be highlighted in the left side of the Open box.

12.Click **OK**.

13.Click **OK** again.

14.Check the **SLIP and Scripting for Dial-Up Networking** box.

15.Click **Install**, and click **OK**.

More details about this and optional connection scripts can be found in *AS/400 Client Access for Windows 95/NT - Setup V4R2*, SC41-3512, in section 3.3.1, “Setting up to Connect to AS/400.”

You can also visit the Microsoft home page at: `http://www.microsoft.com/`

4.15.3.1 Configuring a new Windows connection for PPP

After the Dial-Up Networking support is installed, you must configure a SLIP connection to communicate with the AS/400 system. Refer to Figure 274 as a starting point as you run this procedure to configure the connection.

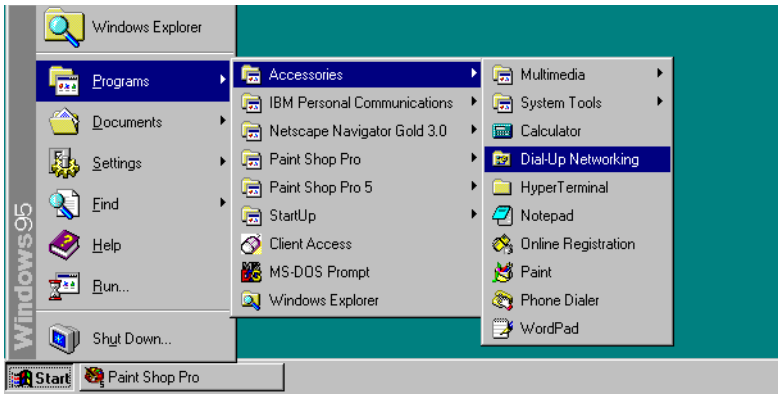


Figure 274. Starting Dial-Up Networking

1. On the Windows menu bar, click **Start->Programs->Accessories->Dial-Up Networking** to start the Dial-Up Networking configuration. If this is the first configuration, the display shown in Figure 275 appears. If there are existing configurations, the Dial-Up Networking window (Figure 279 on page 189) appears. If you receive the Dial-Up Networking window, double-click the **Make New Connection** icon. The display shown in Figure 276 appears. Skip to step 3 on page 188.

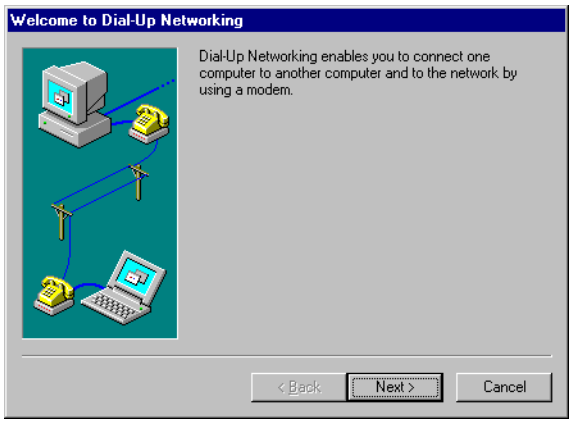


Figure 275. Starting a new Dial-Up configuration

2. Click **Next**. The display shown in Figure 141 appears.

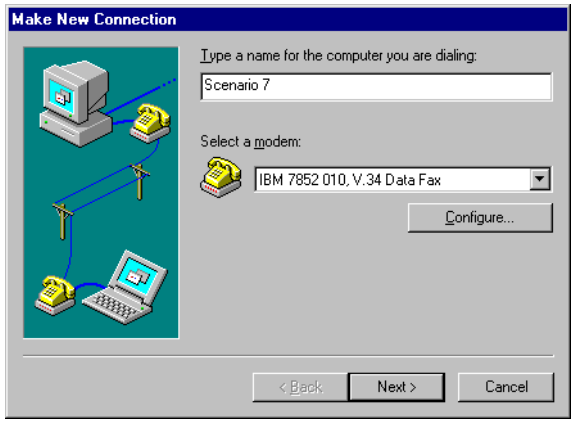


Figure 276. Specifying a new name and a modem to use

3. Enter the name of the new connection, and select the modem. Click **Next**. The display shown in Figure 277 appears.

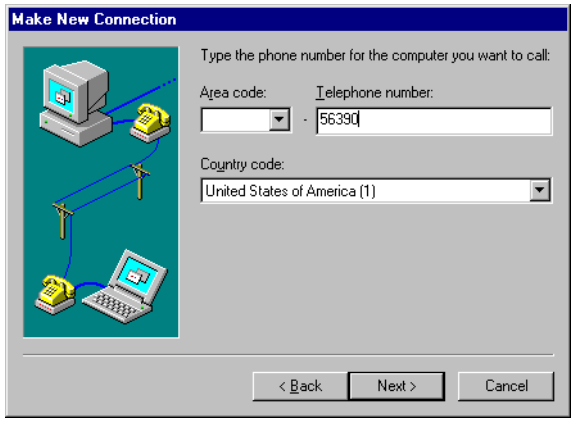


Figure 277. Entering the remote phone number

4. Enter the name of the new connection, and select the modem. Click **Next**. The display shown in Figure 278 appears.

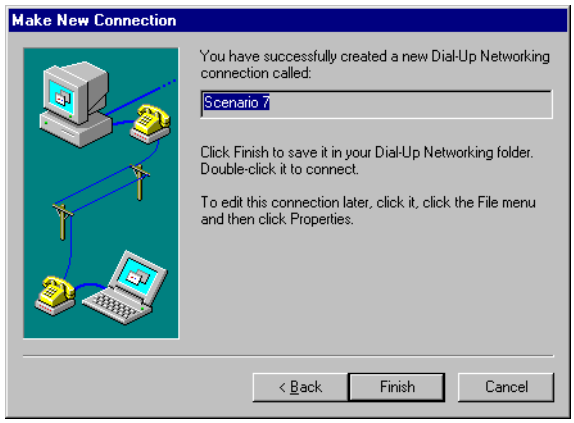


Figure 278. Basic configuration is completed

5. Click **Finish**. The display shown in Figure 279 appears.

Figure 279. Modifying the properties of the newly created connection

6. You now need to specify the server type to which to connect. Right-click the icon added for your connection. Select the **Properties** item from the pop-up menu. A window similar to the one shown in Figure 280 appears. The contents of the window may be different depending on your level of Windows.

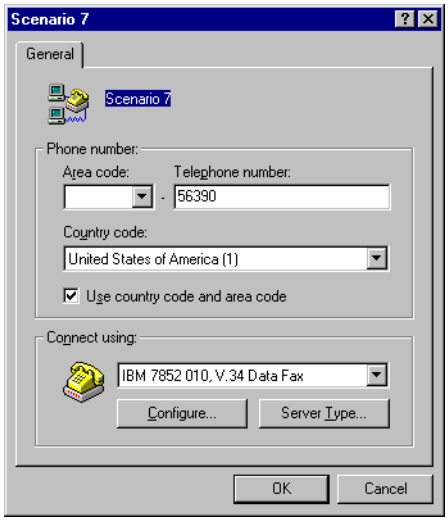


Figure 280. Adjusting the Server Type

7. Click the **Server Type** button or tab. The display shown in Figure 281 appears.

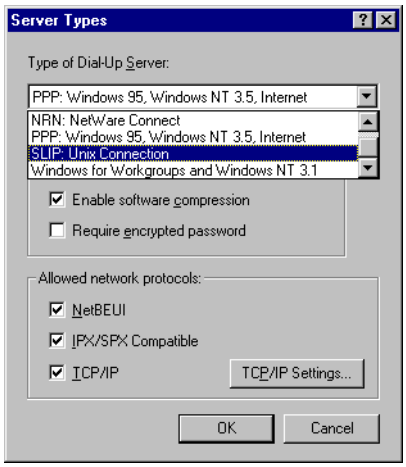


Figure 281. Selecting the SLIP server type

8. Select the **Slip: Unix Connection** item from the drop-down box. The display shown in Figure 282 appears.

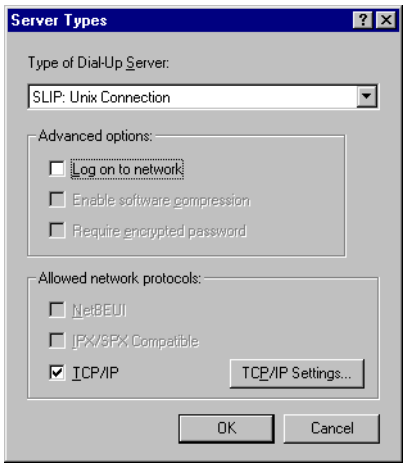


Figure 282. TCP/IP is the only protocol allowed on a SLIP connection

9. Remove the check from **Log on to network**, and place a check in **TCP/IP**. Notice that TCP/IP is the only supported protocol with SLIP. Click **TCP/IP Settings**. The display shown in Figure 283 appears.

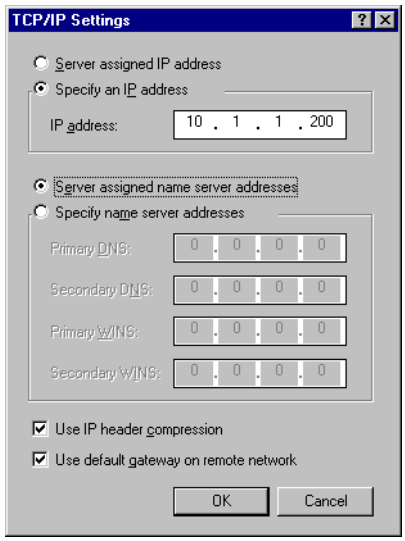


Figure 283. Specifying the IP address of this SLIP connection

10. Select **Specify an IP address**, and type the IP address to be assigned to the PC in the IP address field. This address is provided by the network administrator. In this example, we use 10.1.1.200. Select the other values based on the information provided to you by the network administrator. Click **OK**. You return to the Dial-Up Networking window (Figure 279 on page 189).

4.15.4 Testing the scenario

This section shows how to test the connection that you defined. It consists of three parts:

- Starting the point-to-point connection for the SLIP answer
- Starting the dial connection from Windows
- Starting a TCP/IP application such as TELNET

4.15.4.1 Starting the point-to-point connection for SLIP answer

The point-to-point connection profile must be started before the dial-in connection will be answered. To start the connection profile, follow these steps:

1. To display the Work with Point-to-Point TCP/IP Profiles (WRKTCPPTP) panel, type the following command on any AS/400 command line:

   ```
   WRKTCPPTP
   ```

   Press Enter.

2. From the Work with Point-to-Point TCP/IP (WRKTCPPTP) panel, type option `9` (Start) beside the profile name to start the profile. After the profile has successfully started, the status of the profile should be RINGW (ring waiting). It is necessary to press F5 (Refresh) to update the display. If the status stays in STRSSN, there may be a message in the QSYSOPR message queue.

You are now ready to start the Windows dial-up connection.

4.15.4.2 Starting a dial connection with Windows 9x

To complete the connection, you must prompt Windows to dial the phone number and make the connection. Use the following procedure to start the dial connection using Windows 9x:

1. On the Windows menu bar, click **Start->Programs->Accessories->Dial-Up Networking** to start Dial-Up Networking. The display shown in Figure 279 on page 189 appears.
2. From the **Dial-Up Networking** window, double-click the connection with which you want to connect (SCEN7 in our example). The display shown in Figure 284 appears.

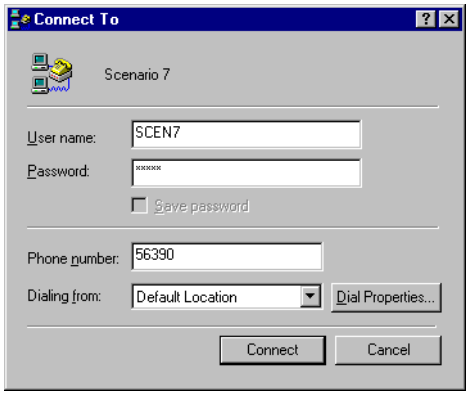


Figure 284. Starting the SLIP connection to the AS23 system.

3. Type the user ID and password values. You may also change the dial properties. Click **Connect**. The display shown in Figure 285 on page 192 appears.

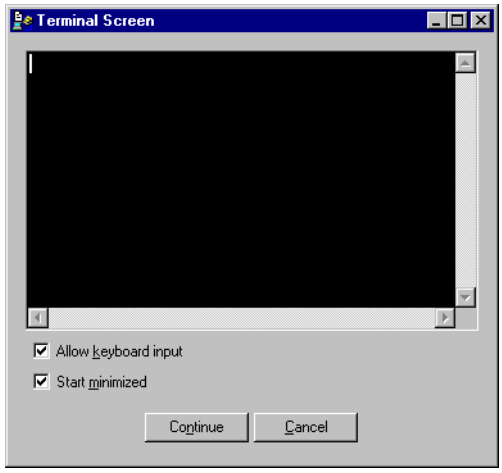


Figure 285. Getting ready to access the connection

4. The window shown in Figure 285 is presented to begin the interaction with the SLIP script. The cursor is located in the upper-right of the black area. Press Enter *once* to continue. The display shown in Figure 286 appears.

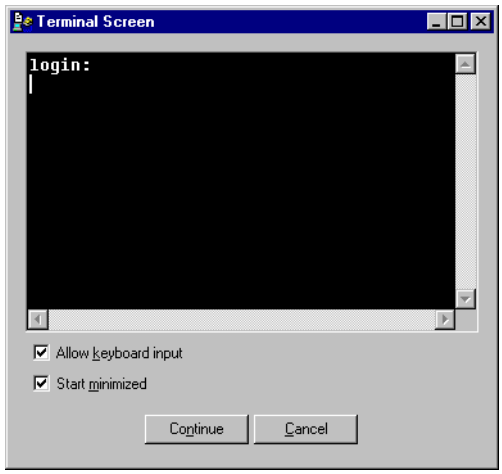


Figure 286. Entering the user ID

5. Type in your user ID as the login. The text entered is not echoed on the display. Press Enter. The display shown in Figure 287 appears.

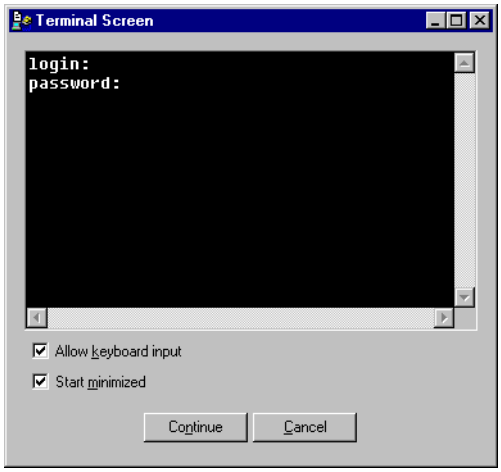


Figure 287. Entering the password

6. Type in the password, and press Enter. The text entered is not echoed on the display. Click **Continue**. The display shown in Figure 288 appears.

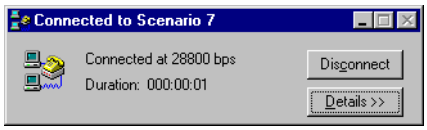


Figure 288. The SLIP connection is now ready

The connection is now ready to work.

4.15.4.3 Starting a TCP/IP application

Refer to 4.9.4.3, “Starting a TCP/IP application” on page 127, for instructions about starting the Telnet application to test the link.

4.16 Common errors

The following sections describe some common errors that are found when setting up point-to-point connections. While the examples refer to PPP connections, most of these techniques also work when used with SLIP connections.

Modem hardware configuration

The typical problem with modem hardware is the wrong configuration of dip-switches and other hardware settings. Make sure that the modem is configured for the correct framing type, either Async or Sync. Refer to the modem manual for instructions.

Modem AT commands

If the modem you are trying to use is not in the predefined list of modems supplied with OS/400, you have to create a new modem. This can be done by basing the new modem on an existing modem, for example, $generic hayes.

If you suspect problems with the Hayes AT commands, check the PPP jobs in the QSYSWRK subsystem (Figure 105 on page 96). You should examine either the job log of the PPP job or the spooled files that the job generates.

You may access the job logs from the QSYSWRK subsystem or from the WRKCFGSTS panel. A debugging example is shown in Figure 289 through Figure 292.

```
                      Work with Configuration Status              AS8
                                                          11/20/98  13:23:00
Position to  . . . . .              Starting characters

Type options, press Enter.
  1=Vary on   2=Vary off   5=Work with job   8=Work with description
  9=Display mode status    13=Work with APPN status...

Opt  Description       Status               -------------Job--------------
     SCEN2            ACTIVE
       SCEN2NET       ACTIVE
5        SCEN2TCP     ACTIVE               QTPPANS008  QTCP        013692









                                                                       Bottom
Parameters or command
===>
F3=Exit   F4=Prompt   F12=Cancel   F23=More options   F24=More keys
```

Figure 289. Work with the PPP job

```
                              Work with Job
                                                          System:   AS8
Job:   QTPPANS008     User:   QTCP          Number:   013692

Select one of the following:

     1. Display job status attributes
     2. Display job definition attributes
     3. Display job run attributes, if active
     4. Work with spooled files

    10. Display job log, if active or on job queue
    11. Display call stack, if active
    12. Work with locks, if active
    13. Display library list, if active
    14. Display open files, if active
    15. Display file overrides, if active
    16. Display commitment control status, if active
                                                                       More...
Selection or command
===> 4

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
```

Figure 290. Look at the spooled files or the job log

```
                          Work with Job Spooled Files

Job:   QTPPANS008     User:   QTCP           Number:   013692

Type options, press Enter.
  1=Send   2=Change   3=Hold   4=Delete   5=Display   6=Release   7=Messages
  8=Attributes        9=Work with printing status

                Device or                          Total   Current
Opt  File       Queue       User Data    Status    Pages     Page   Copies
 5   SCEN2      QPRINT      QTPPANS008   RDY         0                 1

                                                                      Bottom
Parameters for options 1, 2, 3 or command
===>
F3=Exit   F10=View 3   F11=View 2   F12=Cancel   F22=Printers   F24=More keys
```

Figure 291. Display the spooled file

```
                              Display Spooled File
File  . . . . . :   SCEN2                          Page/Line   1/6
Control . . . . .                                  Columns     1 - 78
Find  . . . . . .
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
16:13:22.849 === Attempting modem reset.
16:13:22.928 ==> AT&FS0=0
16:13:22.929 === Reading modem response.
16:13:23.502 <== AT&FS0=0
16:13:23.502 ... ERROR
16:13:25.391 === Attempting modem reset.
16:13:25.391 ==> AT&FS0=0
16:13:25.392 === Reading modem response.
16:13:25.920 <== AT&FS0=0
16:13:25.920 ... ERROR
16:13:27.793 === Attempting modem reset.
16:13:27.793 ==> AT&FS0=0
16:13:27.794 === Reading modem response.
16:13:28.321 <== AT&FS0=0
16:13:28.321 ... ERROR

                                                                      Bottom
F3=Exit   F12=Cancel   F19=Left   F20=Right   F24=More keys
```

Figure 292. The spooled file from the PPP job

As shown in Figure 292, the PPP job receives an error while initiating the connection. The spooled file is useful in debugging the wrong AT commands. A brief description of the symbols shown in the spooled file is provided here:

- ======== Regular information text
- <======= Outbound text (to the modem) follows
- =======> Inbound text (from the modem) follows

Note that the modem in this example echoes the AT commands received from the AS/400 system back to the AS/400 PPP job.

Figure 293 shows a connection with no errors.

```
                            Display Spooled File
File  . . . . . :   SCEN2                              Page/Line   1/6
Control . . . . .                                      Columns     1 - 78
Find  . . . . . .
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
13:06:53.764 === Attempting modem reset.
13:06:53.765 ==> AT&FS0=0
13:06:53.766 === Reading modem response.
13:06:54.430 <== AT&FS0=0
13:06:54.430 ... OK
13:06:54.436 === Attempting modem initialization.
13:06:54.436 ==> AT&D2&C1X4V1Q0S7=70W2\N3&K3&S1
13:06:54.439 === Reading modem response.
13:06:55.106 <== AT&D2&C1X4V1Q0S7=70W2\N3&K3&S1
13:06:55.106 ... OK
13:06:55.108 === Attempting modem dial/answer.




                                                                        More...
F3=Exit   F12=Cancel   F19=Left   F20=Right   F24=More keys
```

Figure 293. PPP connection profile debugging

For some reason, if you cannot find the spooled files from the PPP jobs, use the Work with Spooled Files (`WRKSPLF QTCP`) command. This can help you locate the spooled files. Look for job names starting with User Data `QTPPANSnnn` or `QTPPDIALnn`.

Normally, the spooled file is only generated in cases where errors occur. To force the generation of the spooled file, start the PPP connection from the green-screen interface with the Start Point-to-Point TCP/IP (STRTCPPTP) command (Figure 294).

```
                  Start Point-to-Point TCP/IP (STRTCPPTP)

Type choices, press Enter.

Configuration profile  . . . . . > SCEN2         Name
Restart  . . . . . .   *NO          *NO, *YES
Script dialog output . . . . . .   *PRINT       *ERROR, *NONE, *PRINT

                           Additional Parameters

Send inquiry message . . . . . .   *NO          *NO, *YES
Autodelete configuration . . . .   *NO          *NO, *YES

                                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 294. Start the PPP job with the STRTCPPTP command

Problems with PPP users and passwords

Make sure that the user ID and password are entered using the same case. For example, make sure that at system A, the user is “USER” and it is spelled the same way on the system B, “USER”, not “User”.

Furthermore, make sure that the authentication protocol used by the peers is the same. Do not use PAP at one peer, while the other peer is configured as CHAP.

Problems with PPP lines when starting the configuration profile

Remember to vary off other lines using the same hardware resource.

Problems with the PPP protocol

Investigating the lower-levels of the PPP protocol may be necessary in some situations where the peers are unable to communicate with each other due to a configuration error. If the PPP log or the job log of the PPP job does not show any indication of the problem, you can investigate the problem by using the communications trace function. The following screens show the use of a communications trace.

Use the Start Communications Trace (STRCMNTRC) command to start a communication trace. Figure 295 on page 198 shows an example.

```
                  Start Communications Trace (STRCMNTRC)

Type choices, press Enter.

Configuration object . . . . . > SCEN4         Name
Type . . . . . > *LIN          *LIN, *NWI, *NWS
Buffer size  . . . . . .  16M          *MIN, *MAX, 128K, 256K, 2M...
Data direction . . . . . .  *BOTH        *SND, *RCV, *BOTH
Trace full . . . . . .  *STOPTRC     *WRAP, *STOPTRC
Number of user bytes to trace:
  Beginning bytes  . . . . . .  *CALC        Number, *CALC
  Ending bytes . . . . . .                   Number, *CALC

                                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 295. Starting a communication trace on the SCEN4 PPP line

Select the appropriate buffer size for your system and your requirements.

After the communication trace has been started, you can initiate your PPP connection.

If the buffer runs full, the trace automatically stops. Otherwise, you have to stop the trace by issuing the End Communications Trace (ENDCMNTRC) command as shown in Figure 296.

```
                    End Communications Trace (ENDCMNTRC)

Type choices, press Enter.

Configuration object . . . . . > SCEN4         Name
Type . . . . . > *LIN          *LIN, *NWI, *NWS

                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 296. Stopping the communication trace on the SCEN4 PPP line

Now you can print the trace using the Print Communications Trace (PRTCMNTRC) command as shown in Figure 297.

```
                   Print Communications Trace (PRTCMNTRC)

Type choices, press Enter.

Configuration object . . . . . > SCEN4         Name
Type . . . . . > *LIN          *LIN, *NWI, *NWS
Output . . . . . *PRINT        *PRINT, *OUTFILE
Character code . . . . . > *ASCII        *EBCDIC, *ASCII, *CALC
Format TCP/IP data . . . . . > *YES          *LINTYPE, *YES, *NO
Format LCP data  . . . . . *YES          *YES, *NO
Format NCP data  . . . . . *YES          *YES, *NO
Format TCP/IP data by address:
  Source/destination IP address    *ALL
  Source/destination IP address    *ALL

                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 297. Printing the communication trace on the SCEN4 PPP line

Remember to specify `ASCII` in the Character Code(CODE) parameter.

A spooled file containing the trace is now generated. Figure 298 on page 200 shows an example of the trace on a PPP line. The PPP protocol uses options to

configure the PPP connection. Any misconfiguration or problems may be found if you search for the *Configure Nak* LCP code. Please refer to the relevant PPP RFCs for a complete description of the PPP protocol.

```
COMMUNICATIONS TRACE      Title:                              11/25/99  13:32:57                        Page:       3
 Record       Data    Record      Record          DLC Frame  Address  Control
 Number  S/R  Length  Status      Timer           Format     Field    Field    Protocol ID
 ------  ---  ------  ----------  --------------  ---------  -------  -------  ------------
      1   S       10   00000000   13:31:37.42368    ASYNC
        Data . . . . . :  4154445435363131 300D                                        *AIDT56110.                    *
      2   R       10   00000000   13:31:37.94419    ASYNC
        Data . . . . . :  4154445435363131 300D                                        *AIDT56110.                    *
      3   R       17   00000000   13:31:51.85288    ASYNC
        Data . . . . . :  0D0A434F4E4E4543 542032383830300D  0A                         *..CONNECT 28800..             *
      4   S       24   00000000   13:31:51.97144    PPP        FF      UI    C021 (LCP)
        LCP  . . . . . :  Code: 01 (Configure Request)       ID: 11           Length:    20
        Option . . . . :  Code: 01 (MRU)                     Length:   4      MRU:  2048
        Option . . . . :  Code: 02 (Async Map)               Length:   6      Async Map: 00000000
        Option . . . . :  Code: 05 (Magic Number)            Length:   6      Magic Number: 0F6766BC
        Data . . . . . :  FF03C02101110014 0104080002060000  000005060F6766BC                *...!.................gf.       *
      5   R       29   00000000   13:31:52.10729    PPP        FF      UI    C021 (LCP)
        LCP  . . . . . :  Code: 01 (Configure Request)       ID: 00           Length:    25
        Option . . . . :  Code: 02 (Async Map)               Length:   6      Async Map: 00000000
        Option . . . . :  Code: 03 (CHAP Protocol)           Length:   5      CHAP Protocol: C223
        Option Data  . :  80                                                           *.                             *
        Option . . . . :  Code: 05 (Magic Number)            Length:   6      Magic Number: 00004354
        Option . . . . :  Code: 07 (Protocol Comp)           Length:   2
        Option . . . . :  Code: 08 (Addr/Cntrl Comp)         Length:   2
        Data . . . . . :  FF03C02101000019 0206000000000305  C223800506000043 5407020802          *...!.............#.....CT....   *
      6   R       24   00000000   13:31:52.11412    PPP        FF      UI    C021 (LCP)
        LCP  . . . . . :  Code: 02 (Configure Ack)           ID: 11           Length:    20
        Option . . . . :  Code: 01 (MRU)                     Length:   4      MRU:  2048
        Option . . . . :  Code: 02 (Async Map)               Length:   6      Async Map: 00000000
        Option . . . . :  Code: 05 (Magic Number)            Length:   6      Magic Number: 0F6766BC
        Data . . . . . :  FF03C02102110014 0104080002060000  000005060F6766BC                *...!.................gf.       *
      7   S       12   00000000   13:31:52.11476    PPP        FF      UI    C021 (LCP)
        LCP  . . . . . :  Code: 03 (Configure Nak)           ID: 00           Length:     8
        Option . . . . :  Code: 03 (PAP Protocol)            Length:   4      PAP Protocol:  C023
        Data . . . . . :  FF03C02103000008 0304C023                                    *...!.......#                  *
      8   R       28   00000000   13:31:52.21280    PPP        FF      UI    C021 (LCP)
        LCP  . . . . . :  Code: 01 (Configure Request)       ID: 01           Length:    24
        Option . . . . :  Code: 02 (Async Map)               Length:   6      Async Map: 00000000
        Option . . . . :  Code: 03 (PAP Protocol)            Length:   4      PAP Protocol:  C023
        Option . . . . :  Code: 05 (Magic Number)            Length:   6      Magic Number: 00004354
        Option . . . . :  Code: 07 (Protocol Comp)           Length:   2
        Option . . . . :  Code: 08 (Addr/Cntrl Comp)         Length:   2
        Data . . . . . :  FF03C02101010018 0206000000000304  C023050600004354 07020802          *...!.............#....CT....   *
      9   S       28   00000000   13:31:52.22188    PPP        FF      UI    C021 (LCP)
        LCP  . . . . . :  Code: 02 (Configure Ack)           ID: 01           Length:    24
        Option . . . . :  Code: 02 (Async Map)               Length:   6      Async Map: 00000000
        Option . . . . :  Code: 03 (PAP Protocol)            Length:   4      PAP Protocol:  C023
        Option . . . . :  Code: 05 (Magic Number)            Length:   6      Magic Number: 00004354
        Option . . . . :  Code: 07 (Protocol Comp)           Length:   2
        Option . . . . :  Code: 08 (Addr/Cntrl Comp)         Length:   2
        Data . . . . . :  FF03C02102010018 0206000000000304  C023050600004354 07020802          *...!.............#....CT....   *
     10   S       10   00000000   13:31:52.22304    PPP                   C021 (LCP)
        LCP  . . . . . :  Code: 0B (Discard Request)         ID: 12           Length:     8
                          Magic Number: 0F6766BC
        Data . . . . . :  C0210B1200080F67 66BC                                        *.!.....gf.                    *
     11   S       18   00000000   13:31:52.22471    PPP                   C023 (PAP)
        PAP  . . . . . :  Code: 01 (Authenticate Request)    ID: 13           Length:    16
                          Peer-ID Length:   5                Peer-ID:  534345 4E34          *SCEN4                         *
                          Password Length:   5               Password: 736365 6E34          *scen4                         *
        Data . . . . . :  C023011300100553 43454E3405736365  6E34                           *.#.....SCEN4.scen4            *
     12   R       28   00000000   13:31:54.05323    PPP        FF      UI    C021 (LCP)
        LCP  . . . . . :  Code: 01 (Configure Request)       ID: 01           Length:    24
        Option . . . . :  Code: 02 (Async Map)               Length:   6      Async Map: 00000000
        Option . . . . :  Code: 03 (PAP Protocol)            Length:   4      PAP Protocol:  C023
```

Figure 298. Sample line trace data

After you end your problem determination, you should delete the collected trace using the Delete Communications Trace (DLTCMNTRC) command. This frees up the auxiliary storage space allocated by the trace (Figure 299).

```
                    Delete Communications Trace (DLTCMNTRC)

Type choices, press Enter.

Configuration object . . . . . > SCEN4         Name
Type . . . . . .   *LIN          *LIN, *NWI, *NWS

                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 299. Deleting the communication trace on the SCEN4 PPP line

Chapter 5. Telnet and the AS/400 system

This chapter discusses Telnet APIs and exit programs. It also explains the host on demand configuration in combination with SSL.

5.1 AS/400 Telnet server

The Telnet protocol provides a standardized interface, through which a program on one host (the Telnet client) may access the resources of another host (the Telnet server) as though the client were a local terminal connected to that server.

The AS/400 Telnet server support negotiates the transmission of data with the remote Telnet client application for the following operating modes:

- 5250 full-screen mode
- 3270 full-screen mode
- VT220 full-screen mode
- VT100 full-screen mode
- ASCII line mode
- Printer pass-through mode

These operating modes are negotiated by the Telnet server and the Telnet client application. The functions available to you depend on the terminal type that is negotiated.

5.1.1 Virtual device description

Every time a Telnet session is established to the AS/400 system, a virtual device is associated with the session (Figure 300 on page 204). It is used to form a connection between a user and a physical workstation attached to a remote system. That virtual device does not have hardware associated with it. It provides information about your physical device to the programs on the server.

To determine which type of emulation an autoselected Telnet client session negotiates, the type of virtual device that is created can be found by typing the following command on the server system:

```
WRKDEVD QPADEV*
```

For both VT220 and VT100, the virtual device type that is created is V100.

Telnet server can be configured or managed with the command line interface or with Operations Navigator (graphical user interface). You may need to use both interfaces to complete your tasks.

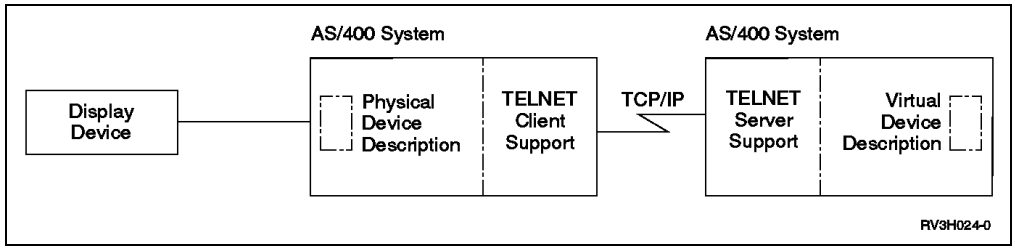


Figure 300. Virtual device description

When a Telnet connection is made to the AS/400 system, the Telnet server support attempts to match a virtual device description with a device type and model similar to the device on your local system. The server system automatically selects (and creates, if necessary) a virtual device description. All this happens if you do not specify a particular device name. If a specific device name is requested, one of the following events happens:

- If it does not exist and if it is allowed to create by QAUTOVRT, the system creates that device.
- If it does not exist and if it is not allowed to create by QAUTOVRT, the request fails.
- If it exists and is already in use, the request fails.

5.1.2 QAUTOVRT system value

The QAUTOVRT system value specifies the maximum number of devices that are automatically configured by the system. This system value has been changed for Version 4 Release 2 to support numeric values of 0 through 32,500. This implies that the “xxxx” values for device naming of the virtual devices (QPADEVxxxx) are no longer only numeric values, but are also alphanumeric characters from 0001 to ZZZZ (excluding A, E, I, O, U, and Y). This allows a maximum of 810,000 unique device names.

If you want to use more than 32,500 devices, which is the maximum value for the QAUTOVRT system value, you can set the QAUTOVRT system value to `*NOMAX` to allow additional devices to be created.

5.1.3 Telnet device naming convention

The AS/400 Telnet server uses the following conventions for naming automatically created virtual controllers and devices:

- Virtual controllers are named QPACTLnn
- Virtual devices are named QPADEVxxxx

This section discusses the naming convention for devices and its considerations.

5.1.3.1 Unpredictable system-assigned device names

The Telnet server reuses available existing virtual device descriptions that were auto-created by selecting virtual devices of the same device type and mode. When there are no more device type and model matches, there are still available virtual devices. The device type and model are changed or deleted and recreated if necessary to match the client device and model negotiated between the Telnet client and server.

Up until now, OS/400 Telnet server has been unable to take advantage of many AS/400 capabilities commonly used to manage interactive jobs and users. Because the device name QPADEVxxxx used for the interactive job was system assigned and unpredictable, it was not possible to set up an interactive subsystem so that users in the subsystem always use the same device description. It was difficult to take advantage of the AS/400 work management capabilities, specifically associating a specific device description with a particular user.

5.1.3.2 Specifically assigned Telnet session device names

There are two ways to select virtual devices for Telnet. A Telnet client or exit program can choose to supply their own name for the virtual device. In this case, the virtual controller is named QVIRCDnnnn. The name specified for the virtual device has to conform to the OS/400 object naming rules. You should establish naming conventions to easily manage your configuration afterwards.

> **Note**
>
> The QTCP user profile must be granted authority to the user-created virtual controller devices.

There are now a variety of ways to assign a specific device name to a Telnet session. This opens up all the options of the work management workstation configuration and makes it possible to better manage users on your AS/400 system.

The virtual device selection method can be done by the attaching client (using the Internet-Draft “5250 Telnet Enhancements” listed on the Web at: `http://www.ietf.org/ids.by.wg/tn3270e.html`), or through a registered user exit program where you specify the device description for the Telnet session.

To use the device selection capability with Client Access for Windows 95/NT, refer to the Informational APAR II10918 for more detailed information. This functional enhancement adds the ability to specify a device ID for Telnet. To take advantage of this capability, you can set the workstation ID on the client emulator session the same way you specified the workstation ID for emulation sessions that used APPC in the past.

Beginning with V4R1, the IBM thin client, the Network Station, lets you request a specific device name. Using the Network Station Manager, click **Startup**, followed by clicking **Programs**. You can specify that you want to configure the system defaults, group defaults, or user defaults. The next screen that appears lets you select the 5250 sessions to autostart. In Other Parameters, you can enter `DISPLAY_NAME` to specify the device description that should be used for the 5250 session.

All virtual devices that are created under the QPACTLnn controllers and the QVIRCDnnnn controllers count toward the QAUTOVRT limit. If you want to delete virtual devices to enforce a smaller QAUTOVRT value, you should begin by deleting the devices from the controller with the highest QPACTLnn value.

In case the device name is specifically requested, the client request or the user exit program will be honored, regardless of the attributes of the terminal type

negotiated. However, be aware that results can be unpredictable if the device attributes of the selected device do not adequately match the characteristics or attributes of the physical device itself.

Client Access Configure PC5250 screen

In Client Access for Windows 95/NT, you have to go to the Configure PC5250 window (Figure 301), where you can enter a device name in the Workstation ID field.

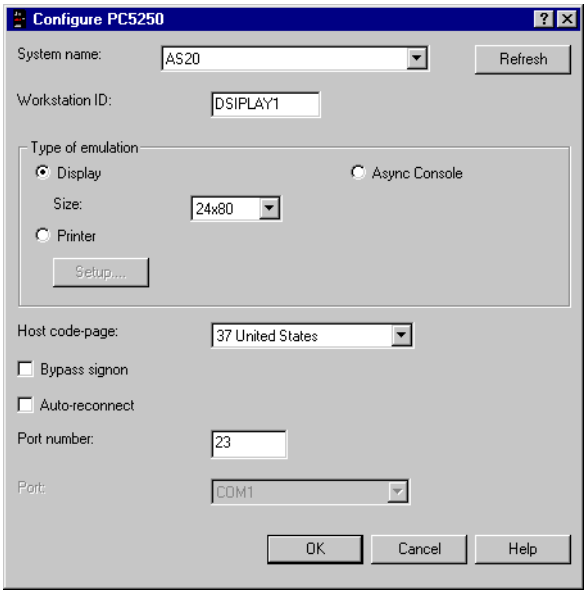


Figure 301. Configure PC5250 display

Graphical Access AS/400 Connection Properties screen

When using Graphical Access on your PC, you can also associate a unique workstation ID with a specific physical end-user device.

You can do this by clicking **Start->Programs->IBM AS/400 Client Access->AS/400 Connections**. You have to select the appropriate system's AS/400 Connection. Right-click **Properties**. Click the **Default View** tab (Figure 302). On this page, select **Graphical Access**. At the Graphical Access command line options field, enter:

```
/G:DevN=name
```

In this line, the name is a device name with up to eight characters.

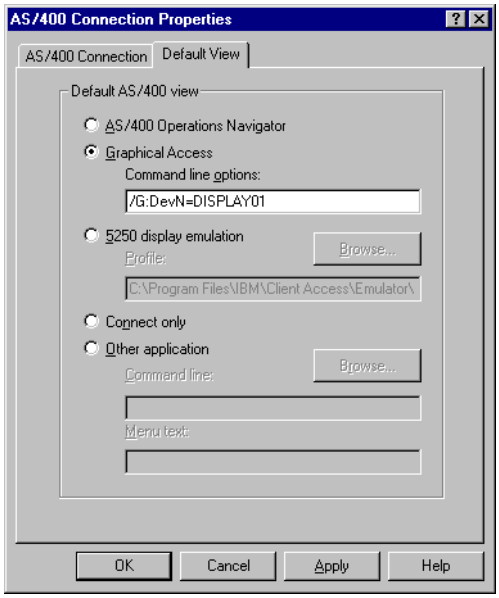


Figure 302. Graphical access Connection Properties display

5.1.3.3 Subsystem routing and device name selection

In addition to the device name, you can also set the code page, character set, and keyboard attributes for each Telnet session. This gives you a greater flexibility in national language support. The requested code page and character set are used in the CHRID parameter, and the keyboard attributes are used in the KBDTYPE parameter.

Using the functions mentioned above, you can enter the device names in the workstation entries of interactive subsystems to specify in which subsystems Telnet users run. *Communications Management Guide*, SC41-5406, strongly recommends that no more than 300 users be serviced by any given subsystem to maintain good performance of the subsystem's device recovery processing.

Beginning with V4R2, users can configure their interactive subsystems to subdivide the work if necessary. This can be accomplished by using the Add Work Station Entry (ADDWSE) command to specify which devices a subsystem should allocate a particular name of virtual terminal devices.

The following command uses QINTER to allocate all QPADEV* workstations, which means that all such devices are routed to the QINTER subsystem:

```
ADDWSE SBSD(QINTER) WRKSTN(QPADEV*) AT(*SIGNON)
```

The following command uses QINTER to not allocate all QPADEV* workstations, which means that all such devices can be allocated to a different subsystem:

```
ADDWSE SBSD(QINTER) WRKSTN(QPADEV*) AT(*ENTER)
```

Note

The subsystem for the QPADEV* devices must be running, or the default Telnet will not work.

Users can now also develop their own device naming conventions to subdivide the work. For example, one kind of subdivision is to route certain devices to national language support related subsystems in two different locations. This is explained in the following example.

For the purpose of this example, the two users are located in Endicott and Rochester. The users are assigned to AS/400 subsystems ENDICOTT and ROCHESTER, respectively, according to their geographic location. The characteristics of this example include:

- The IP addresses for Endicott start with 1.2.3.*
- The IP addresses for Rochester start with 2.3.4.*

For all of the Endicott TELNET sessions to run in the ENDICOTT subsystem and the Rochester TELNET sessions to run in the ROCHESTER subsystem, the user exit program is employed to create a virtual device name that starts with “ENDI” for all TELNET connections from 1.2.3.* and a virtual device name that starts with “ROCH” for all connections from 2.3.4.*

The user exit program assigns the virtual device name “ENDI0001” for an IP address of 1.2.3.01 and a virtual device name of “ROCH0001” for an IP address from 2.3.4.01.

To ensure that virtual devices ENDI0001 and ROCH0001 go into subsystems ENDICOTT and ROCHESTER, respectively, the workstation entries are set up as follows:

```
ADDWSE SBSD(QINTER) WRKSTN(ENDI*) AT(*ENTER)
ADDWSE SBSD(QINTER) WRKSTN(ROCH*) AT(*ENTER)
ADDWSE SBSD(ENDICOTT)  WRKSTN(ENDI*) AT(*SIGNON)
ADDWSE SBSD(ROCHESTER) WRKSTN(ROCH*) AT(*SIGNON)
```

5.1.4 DSCJOB support

There are two basic ways on the AS/400 system to return display devices to a sign-on screen. You can end the interactive job, and the job will be terminated. Otherwise, you can disconnect the job, in which case the job is suspended rather than ended.

In case the user gets signed on again from the same device, the disconnected job is reconnected. This allows them to continue working where they left off. Disconnecting jobs reduces the overhead related to ending and starting interactive jobs, which may be an important consideration in case of a network failure that results in many users suddenly losing their sessions.

Prior to V4R4, being unable to disconnect a Telnet interactive job was a shortcoming. Those jobs could not be reconnected because, to do so, the user had to sign on again using the same display device description. It was unlikely that a user could disconnect and then sign on again using the same device description.

Now, with V4R2 support for specified device names, you can use the Disconnect Job (DSCJOB) command for Telnet sessions assigned to a specific device. Telnet sessions that use the system-assigned device descriptions still cannot be disconnected.

V4R2 also supports the *DSCMSG and the *DSCENDRQS options of the job's DEVRCYACN attribute and the QDEVRCYACN system value. This device recovery action attribute and system value specify the action to be taken when an I/O error occurs for an interactive job's workstation. The *DSCMSG and the *DSENDRQS values will both disconnect the Telnet interactive job, saving system resources and offering improved error recovery performance in the event of a network failure.

The system value QDSCJOBITV determines when disconnected jobs are cleaned up. Any Telnet sessions that remain disconnected for the time specified in the system value are ended, just like any other interactive disconnected jobs. Jobs that exist, but are not used any more, needlessly increase the size of the work control block table. Having too many such jobs in the system can slow down performance. It's a good idea to use the system value QDSCJOBITV to ensure that the disconnected jobs are ended if user's don't reconnect in a reasonable amount of time.

5.1.5 QINACTITV support

The OS/400 Telnet server also had another limitation in that it was excluded from the system's inactive-job processing. Since V4R2, Telnet and Virtual Terminal APIs connected display sessions are now subject to the settings of QINACTITV, which specifies when the system takes action on inactive interactive jobs. When the QINACTITV expires, the Telnet jobs are included in the processing as defined by QINACTMSGQ. If QINACTMSGQ specifies the timed-out job is to be disconnected, the Telnet session must support the disconnect job function. Otherwise, the job will be ended rather than disconnected.

Note

This function is included in V4R2 and is available by PTF for V4R1 (SF47141) and V3R7 (SF51321).

To activate the QINACTITV support for TELNET server jobs, the following data area must be created and set to a value of '1':

```
CRTDTAARA DTAARA(QSYS/QWTINACTIV) TYPE(*CHAR) LEN(1) VALUE('1')
```

The QINACTITV support for TELNET server jobs can be turned off by either deleting the QSYS/QWTINACTIV data area:

```
DLTDTAARA DTAARA(QSYS/QWTINACTIV)
```

Or, you can change the data area to a value '0':

```
CHGDTAARA DTAARA(QSYS/QWTINACTIV) VALUE('0')
```

You can use the "inactivity timeout" INACTTIMO parameter on the Telnet configuration to reduce the exposure when a user leaves a Telnet session unattended. Before V4R2, QINACTITV processing did not include inactive Telnet sessions, which could be timed out only using the INACTTIMO parameter on the Change Telnet Attributes (CHGTELNA) command. A problem with this solution was that for a device to be considered active, it must be doing input/output (I/O). Long-running applications or queries could be timed out because no I/O was taking place, even though the device remained in use.

Note

We recommend that you set the INACTTIMO parameter in the CHGTELNA to 0 and use the setting of the QINACTITV system value. IBM may drop support for the TCP/IP Telnet INACTTIMO parameter in a future release.

If you specify both QINACTITV and the Telnet INACTTIMO attribute, the time-out that occurs first is the one to take effect. For example, if you set QINACTITV to 180 minutes, and INACTTIMO to 10 minutes, INACTIMMO will always expire first, and jobs will never be active long enough for QINACTITV to affect them.

5.1.6 Support for QRMTSIGN

The QRMTSIGN system value controls how the system is handling remote sign-on requests (for example, whether users are allowed to sign on to the system automatically). In the past, this system value was used to control access to the AS/400 system through 5250 Display Station Passthrough. Beginning with V4R2, Telnet supports the capability for a Client Access user to bypass the sign-on display by sending a user profile name and password with the Telnet session request. The system uses the setting for the QRMTSIGN system value (remote sign-on) to determine how to handle requests for automatic sign-on.

Client Access V3R1M3 or later is required for these secure auto-signon capabilities. Refer to Informational APAR II10918 for more specific requirement information.

The password validation runs before the Telnet exit program runs. The exit program will receive an indication whether the validation was successful. The exit program can still allow or deny the session, regardless of the indicator. In other words, if a Telnet device initialization exit point is used to allow automatic sign-on, the user exit point has the option of overriding the QRMTSIGN setting. The indication has the following possible values:

- Client password not validated (or no password received)
- Client clear-text password validated
- Client encrypted password validated

Although secure automatic sign-on is supported (using DES encryption algorithms to protect the password) for Telnet display sessions which have specific settings for the QRMTSIGN system value, clients do not yet capitalize on that support. Until V4R2, the QRMTSIGN system value historically has had no significance for Telnet.

The Telnet server will begin to honor the *VERIFY and *SAMEPRF settings of the QRMTSIGN system value when a connecting Telnet client requests an automatic sign-on session by passing valid “user profile” and “password” values at session initialization time, rather than waiting for a sign-on panel to be transmitted. As a result, the user is signed on automatically without the sign-on screen being presented to the user.

The *REJECT value is not used. It only applies to 5250 Display Station Passthrough. To reject all Telnet requests to the system, you simply have to stop the Telnet server itself. However, in a future release, changes may be made to cause the Telnet server to begin to honor the *REJECT value of the QRMTSIGN

system value. If this is not desirable for all your remote Telnet sessions, you should consider registering a Telnet initialization user exit program, which can be used to override the QRMTSIGN system value on a connection by connection basis.

The *FRCSIGNON value will result in the sign-on screen to be presented to the user. When you specify a remote session program value for the QRMTSIGN system value, Telnet sessions will behave as if QRMTSIGN were set to *FRCSIGNON.

5.1.7 Telnet device initialization and termination exit points

One of the enhancements that V4R2 brought was the addition of an exit point interface, into which you can hook exit programs that control access to the OS/400 Telnet server. No changes are needed for the connecting Telnet client emulator, so existing clients would be able to immediately benefit from this new feature

An *exit point* is a specific point in the TCP/IP application program where control may be passed to an exit program for customized processing. The *exit program* is a program to which the exit point passes control.

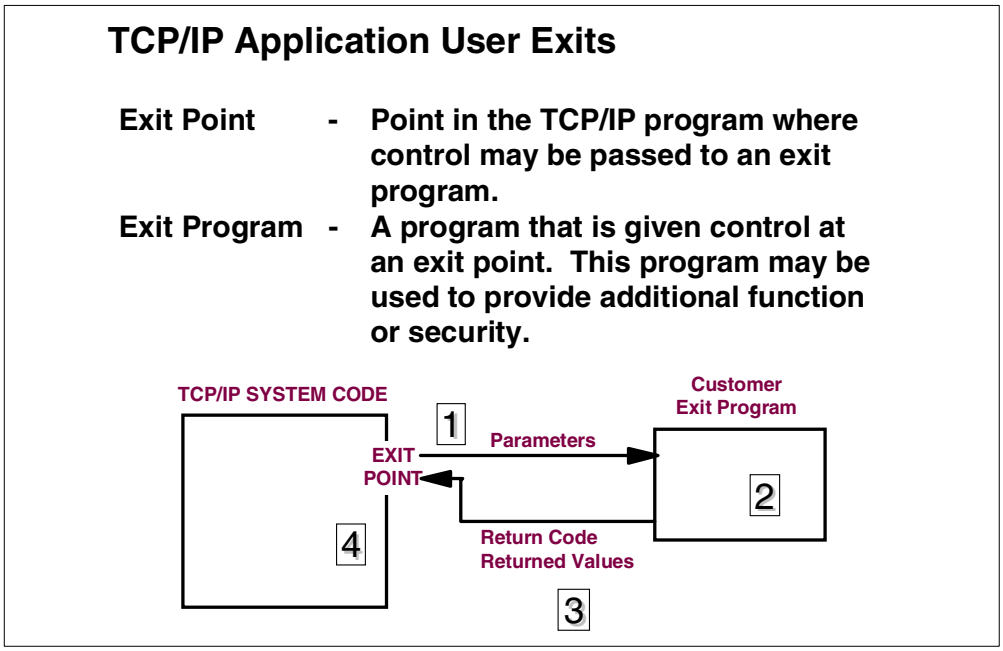


Figure 303. Telnet exit point processing

The Telnet Exit Point Processing flow shown in Figure 303 is described here:

1. TCP/IP application passes request parameters to the exit program.
2. The exit program processes request parameters.
3. The exit program returns information to the TCP/IP application.
4. The TCP/IP application performs an operation based on the exit program response.

The AS/400 Telnet server application now has exit points in its session initialization and termination logic. They are respectively:

- QIBM_QTG_DEVINIT (Exit Point Format is INIT0100)
- QIBM_QTG_DEVTERM (Exit Point Format is TERM0100)

When a Telnet client attempts to connect to the Telnet server, any exit program registered to the QIBM_QTG_DEVINIT exit point is triggered.

The QIBM_QTG_DEVTERM exit point occurs when a Telnet client ends the Telnet session. This gives customers the opportunity to log session termination information and to perform device reset or cleanup operations.

When the OS/400 Telnet server executes the exit program, control is transferred through the exit point to the exit program. After the exit program has performed its tasks, it transfers control back to the OS/400 Telnet server. The return parameters are used by the exit program to direct the Telnet server to perform some action (for example, the exit program may tell the OS/400 Telnet server which virtual device has to be assigned to a specific session).

The exit points give the exit programs full control over connections. The exit points allow them to determine whether a specific user is allowed to connect, which virtual device is allocated, whether the user can bypass the OS/400 sign-on panel, and so on.

5.1.7.1 QIBM_QTG_DEVINIT exit point format INIT0100

For each exit point, there is an associated programming interface, called an *exit point interface*. This is a list of input and output parameters that the exit point will use to pass information between the Telnet server and the exit program.

Table 7 shows the parameters passed between the QIBM_QTG_DEVINIT exit point and the associated exit program. Note that some parameters are input only, some are output only, and some are both input and output.

Table 7. QIBM_QTG_DEVINIT parameters

Required parameters	Direction	Data type
User description information	I/O	Char(*)
Device description information	I/O	Char(*)
Connection description information	Input	Char(*)
Environment options	Input	Char(*)
Length of environment options	Input	Bin(4)
Allow connection	I/O	Char(1)
Allow auto-signon	I/O	Char(1)

This information can also be found in member ETGDEVEX by typing the following command:

```
WRMBRPDM FILE(QSYSINC/H) MBR(ETGDEVEX)
```

A full description of all the parameters in Exit Point Format INIT0100 is presented in the following list:

- **User description information**: Information about the user that the system will use as part of the auto-signon process.
- **Device description information**: Information that the system will use to create or change the device that it uses for this Telnet session.
- **Connection description information**: Information about the client connection that the exit program can use.
- **Environment options**: An array containing all the Internet-Draft “5250 Telnet Enhancements” environment options negotiated by the client. These will be in the exact format that they were in when received from the client and specified by the Internet-Draft “5250 Telnet Enhancements”. The array will, in general, consist of one or more pairs of environment variable names and associated values. The draft specifies that each variable name will always be preceded by either an X'01' or X'03', depending on whether it is an Internet-Draft “5250 Telnet Enhancements” defined VAR or an application specific defined USERVAR. If a value is to be associated with a VAR (or USERVAR), that value will appear next in the array preceded by the Internet-Draft “5250 Telnet Enhancement” defined VALUE character X'01'. This sequence of VAR/VALUE pairs will be repeated up to a maximum of 1024 total bytes of negotiation data.

  The Internet-Draft “5250 Telnet Enhancements” and the more general TELNET negotiation RFCs also allow for control characters to appear within the VAR/USERVAR variable names or their associated values. This is allowed through the use of the ESC character X'02' and rules that apply when the ESC character itself or TELNET IAC control characters must appear in the negotiation sequence. Refer to the Internet-Draft “5250 Telnet Enhancements” for a more complete description of control character escaping rules.
- **Length of environment options**: Actual length of the Environment options parameter.
- **Allow connection**: Applies to all devices and indicates to the TELNET server whether it should allow the client to connect. If the device type is DISPLAY and you have enabled auto-signon, this client may also bypass the sign-on panel on the AS/400 system. The valid values are:
  - “0” rejects the request from the client
  - “1” accepts the request from the client
- **Allow auto-signon**: Applies to DISPLAY device types, and indicates to the TELNET server whether the auto-signon operation should be allowed to proceed for this particular client. If auto-signon is allowed, this client can bypass the sign-on panel on the AS/400 system. The valid values are:
  - “0” rejects the application request from the client. The system ignores the User profile, Current library, Program to call, Initial menu, and Device name output parameters.
  - “1” accepts the application request from the client. The system may consider the User profile, Current library, Program to call, Initial menu, and Device Name output parameters valid if the exit program returns them.

  The “Allow connection” setting will always override the Allow-autosignon, if it is set to '0' (reset the connection).

5.1.7.2 QIBM_QTG_DEVTERM exit point format TERM0100

Table 8 shows the parameter passed between the QIBM_QTG_DEVTERM exit point and the associated exit program.

Table 8. QIBM_QTG_DEVTERM parameter

Required parameters	Direction	Data type
Device name	Input	Char(10)

The only parameter in the Exit Point Format TERM0100 is the device name. It is the specific virtual device to be associated with the Telnet session.

5.1.8 Registering an exit program

Once you have created your exit program, you must tell the Telnet server the name of the program and the library in which it is located. You do this by using the Work with Registration Information (WRKREGINF) or Add Exit Program (ADDEXITPGM) commands (the parameters are similar).

For example, to register an exit program for exit point QIBM_QTG_DEVINIT, follow these steps:

1. Issue the following command:

   ```
   WRKREGINF EXITPNT(QIBM_QTG_DEV*) FORMAT(*ALL)
   ```

   You reach a screen (Figure 304) where you find the Telnet Device Initialization and the Telnet Device Termination Exit Points.

```
                    Work with Registration Information

Type options, press Enter.
  5=Display exit point   8=Work with exit programs

                           Exit
     Exit                  Point
Opt  Point                 Format    Registered  Text
 8   QIBM_QTG_DEVINIT      INIT0100     *YES     Telnet Device Initialization
     QIBM_QTG_DEVTERM      TERM0100     *YES     Telnet Device Termination




                                                                        Bottom
Command
===>
F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
```

Figure 304. Work with Registration Information display

2. Enter option `8` at QIBM_QTG_DEVINIT to display the Work with Exit Programs panel for that exit point.
3. Enter option `1` to add your exit program. Figure 305 shows the resulting Add Exit Program panel where you can enter the program name and library.

```
                        Add Exit Program (ADDEXITPGM)

Type choices, press Enter.

Exit point . . . . . > QIBM_QTG_DEVINIT
Exit point format  . . . . . > INIT0100      Name
Program number . . . . . > 1             1-2147483647, *LOW, *HIGH
Program  . . . . . .                     Name
  Library  . . . . . .                   Name, *CURLIB
Text 'description' . . . . . .   *BLANK




                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display        F24=More keys
```

Figure 305. Add Exit Program display

4. Press F5 to return to the Work with Exit Programs panel. It now lists your exit program as shown in Figure 306.

```
                           Work with Exit Programs

Exit point:   QIBM_QTG_DEVINIT         Format:   INIT0100

Type options, press Enter.
  1=Add   4=Remove   5=Display   10=Replace

              Exit
            Program     Exit
Opt          Number     Program        Library

                  1     YOURPGM        YOURLIB




                                                                      Bottom
Command
===>
F3=Exit   F4=Prompt   F5=Refresh   F9=Retrieve   F12=Cancel
```

Figure 306. Work with Exit Program display

Once this is complete, your exit program is active. It is *not* necessary to end and restart the Telnet server to pick up this exit program.

The previous steps should be repeated for both the initialization and termination exit programs.

5.1.9 Exit point exception handling

User-written exit programs should be well tested prior to installation on a production system, ensuring it can handle most exceptions. There are many possible causes for exception errors.

There are two basic types of exceptions that an exit program may encounter:

- Exceptions it can handle and from which it can recover
- Exceptions it cannot handle and from which it can recover

However, it is not always possible to anticipate every kind of exception that can occur. Therefore, Telnet has established some server safeguards to handle these cases. They are designed to allow the server to continue functioning.

Exceptions occurring when user-written exit programs are run, which are not handled by the exit program itself, are handled by the Telnet server. The exceptions are passed from the exit program to the Telnet server. The server converts any Escape messages posted by the exit program to a Diagnostic message through a call to the Change Exception Message (QMHCHGEM) API. This diagnostic message is posted to the Telnet server job log. This allows the Telnet server to resume operation and continue without aborting. It is the responsibility of the exit program programmer to analyze any Telnet server job logs to isolate and correct the error in the exit program.

Furthermore, the Telnet server interrupts any long running exit program to allow the Telnet server to regain control. A timeout period of 60 seconds has been established as the threshold at which the exit program is interrupted to allow the Telnet server to regain control. This interruption mechanism is critical to preclude the exit program from locking up the Telnet server (and all Telnet clients being serviced by that server job). So, if the Telnet server does not receive a response from the exit program within 60 seconds, it posts a diagnostic message to the Telnet server job log and the operation is resumed. The Telnet client receives an AS/400 sign-on display, just as if no exit program was running. You should analyze any Telnet server job logs frequently for exception errors and resolve them.

5.1.10 Exit point performance

Before you become concerned of what kind of impact the exit programs will have to your Telnet servers and users, remember that the exit programs are only called when you first try to connect (QIBM_QTG_DEVINIT), or when you disconnect (QIBM_QTG_DEVTERM).

The Telnet server response time for your initial session request includes any time it takes for the QIBM_QTG_DEVINIT exit program to be called, executed, and returned. If your exit program is doing significant processing, such as searching a large number of existing devices for a free device name, this performance impact may result in a longer wait before your session is actually established. Well-written programs should not have a noticeable impact from a user perspective.

Once the Telnet session is established, meaning once you have your sign-on panel or another AS/400 panel, there is no performance impact since the exit program is no longer in the Telnet call path. Therefore, established or steady-state Telnet sessions experience no delays whatsoever due to the QIBM_QTG_DEVINIT exit program.

There is no user-visible performance impact associated with disconnecting the session. This should not be confused with signing off the Telnet session. Disconnecting means that you actually end your terminal emulation session, not

just sign-off and return to the sign-on panel. If you disconnect, the QIBM_QTG_DEVTERM exit program is invoked, which performs termination processing for your session. This is not seen by users since it occurs after the connection is broken.

5.1.11 Exit program security

To ensure that your exit program does not expose your system to hackers, you should take steps to secure both the program and the source code for the program itself.

The Telnet server runs under user QTCP. But, if QTCP is not authorized to your exit program, the Telnet server adopts authority sufficient to call the program. If your exit program is not secured, it could be replaced by users to provide unauthorized access to your system.

How do you protect your Telnet exit program?

First, create a secure library with PUBLIC(*EXCLUDE) authority. The only private authority you should grant is *USE authority to QTCP.

Create all source physical files (for source code) in this library with PUBLIC(*EXCLUDE) authority.

Create your exit programs in this library with PUBLIC(*EXCLUDE) authority. If your exit program requires other objects to run (for example, log files, mapping tables, etc.), your exit program must either adopt sufficient owner authority to the object, or you must grant user profile QTCP private authority to the object. The specific private authority you grant depends on the function you require, such as *ADD to insert a log file record or *READ to read a mapping table.

To protect the source code for your exit programs from unauthorized access and tampering, remove all observable information with the `CHGPGM RMVOBS(*YES)` command. For CL programs, turn CL logging off, and disallow the Retrieve CL Source (RTVCLSRC) command.

You should be aware that anyone on your system with *SECOFR authority has the ability to modify the registration information for an exit program, such as changing the program that is registered. The Telnet exit programs can be modified to allow anyone to sign-on, even without knowing the password for a user profile. If you lose control of *SECOFR authority on your system, you lose control of Telnet security as well.

A related concern involves auto-signon and TN5250E clients. When these Telnet clients connect and send a user profile and password, the Telnet server checks the password for validity. If the QRMTSIGN value allows, the server initializes the profile and auto-signon fields of the exit program interface before the exit program is called. Without an exit program, only clients supplying valid passwords have their Allow auto-signon parameter set to 1. An exit program can override this Telnet server password checking by changing the initialized value for the Allow auto-signon parameter. To be secure, exit programs should not change the auto-signon flag, except in cases where they explicitly set a user profile to be used for auto-signon or when they want to deny auto-signon even for clients sending valid passwords. Otherwise, an exit program may inadvertently

open the door for TN5250E clients to submit a profile without a password (or with an invalid password) and still bypass the signon panel.

For a final security note, you should be aware that anyone on the Internet can intercept and read the AS/400 Telnet data packets because they're unencrypted. Hackers can acquire user profile and clear-text password information flowing across the Internet and use it to gain access to your system. If you want to let users access your system via Telnet over the Internet, you should implement some form of secure Virtual Private Network, such as Microsoft's Point-to-Point Tunneling protocol.

Note

Telnet exit programs can greatly enhance the security of your Telnet server.

Each TCP/IP server application is responsible for its own security. If your system is connected to the Internet, there are larger security issues of which you must be aware. Your system must be protected from unauthorized access through your TCP/IP connection to the Internet.

For more information about securing your AS/400 system on the Internet, refer to these sources:

- *AS/400 Internet Security: IBM Firewall for AS/400,* SG24-2162
- *The Domino Defense: Security in Lotus Notes and the Internet*, SG24-4848
- *AS/400 Internet Security: Protecting Your AS/400 from HARM in the Internet*, SG24-4929

5.1.12 Creating exit programs

We have explained the benefits of the Telnet exit programs and we discussed the exit point parameters and walked through the exit program registration. Now, all you need to do is write an exit program to provide the functions you require. You must have OS/400 V4R2 or higher installed on your AS/400 to use Telnet exit points and programs.

Several steps are involved in designing and writing exit programs:

1. Review the purpose of the exit point and the format of its interface.
2. Define the scope and operation of your exit program.
3. Design the exit program.
4. Code the exit program.
5. Add the exit program to the appropriate exit point in the registration facility.

Note

Only users with both *SECADM and *ALLOBJ authority are allowed to add and remove TCP/IP application exit programs.

6. Test your exit program.
7. Test for each user ID.
8. Test for each operation.

The most important step in establishing security exit programs is verifying that the exit program works. You must ensure that the security wall works and does not have any weaknesses.

Note

- If the exit program fails or returns an incorrect output parameter, the operation is not allowed by the TCP/IP application.
- To ensure the highest level of security, create the exit program in a library that has *PUBLIC authority of *EXCLUDE and give the exit program itself a *PUBLIC authority of *EXCLUDE. The TCP/IP application adopts authority when it is necessary to resolve and call the exit program.

5.1.13 Sample Telnet exit programs

You can see some sample exit programs (written in C) on the following Web address: `http://www.as400.ibm.com/tstudio/tech_ref/tcp/indexfr.htm`

Two “versions” of the same program exist. The basic Telnet exit program is a simple exit program that exploits only a subset of the interface. The advanced program is more sophisticated and uses the full features of the interface.

A small amount of modification to these example exit programs is required before they will be of use to you. Instructions on which parts of the programs to modify and how to modify them are included. A CL utility program, TELCRT, is also provided which automates the compilation, installation, and registration process.

To use these programs, you need to have the ILE/C compiler product (5769CX2 - ILE C for AS/400) and also the System Openness Includes (OS/400 Installation Option 13 for QSYSINC library).

The Telnet exit program information is packaged into easily downloadable files. The ZIP and SAVF files contain the same files. We recommend that you use the second method to avoid a CCSID conversion error that may occur with FTP in the first method.

- The files in telnet42.zip are in a format that is compatible with PCs. Choose telnet42.zip to download the program and information files to your PC, unzip them, then transfer them to your AS/400 system. You'll need to rename most of the files once you get them to your AS/400 system.
- Telnet42.savf is an AS/400 save file. Download it to your PC, and then transfer it to your AS/400 system. We recommend that you create a temporary library on your AS/400 system and transfer the save file to that. You can unpack the save file in the temporary library and follow the instructions in the READ.ME file.

5.1.14 Printer emulation support

The enhanced OS/400 Telnet server provides printer emulation support. The Telnet Printer Pass-Through mode (TPPT) allows the AS/400 user with a Telnet client that supports printer emulation to attach printer devices on the AS/400 system over the network and to work over native TCP/IP. This is described in Figure 307 on page 220.

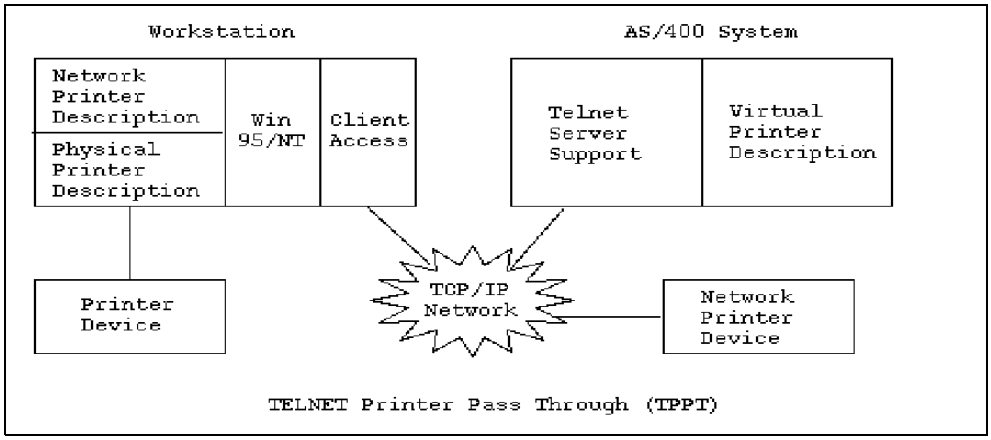


Figure 307. Telnet Printer Pass-Through

If you intend to use the TCP/IP Telnet Printer Pass-Through, check with the client vendor or with third parties that are known to provide 5250 clients for the availability of the printer pass-through function. Clients that support this function are:

- IBM Client Access for Windows 95
- Personal Communications Version 4 Release 2

This support is accomplished by negotiating one of the following generic EBCDIC printer devices:

- IBM 3812-1 for Single Byte Character Set
- IBM 5553-B01 for Double Byte Character Set

Telnet Printer Pass-Through (TPPT) delivers the printer data stream between the two systems as either EBCDIC or ASCII, depending on the preferences of the requesting client.

Additionally, either of these two generic device types can be more completely specified by requesting the AS/400 Host Print Transform (HPT) function and selecting the specific manufacturer type. The printer data stream is sent in ASCII.

5.1.14.1 TPPT mode with Client Access Windows 95/NT Telnet client

The IBM Client Access for Windows 95 client provides both display emulation, 5250 full-screen Telnet client, and printer emulation. You need OS/400 V4R3, V4R2, V4R1, V3R7, or V3R2; Client Access for Windows 95/NT V3R1M3 or V3R2 or Client Access for Windows 3.1 V3R1; and the appropriate PTFs or Service Packs.

What is needed on the AS/400 side?

The QAUTOVRT system value also applies to printer emulation sessions, so be sure to set it large enough to allow for necessary printer sessions, as well as display sessions. The default value QAUTOVRT is set to zero to prevent automatic virtual device creation. You need to change this value. To allow for up to 50 virtual devices, issue the following command:

```
CHGSYSVAL SYSVAL(QAUTOVRT) VALUE(50)
```

The OS/400 Telnet server needs to be started before the printer emulation can be used. This can be done by issuing the command:

```
STRTCPSVR SERVER(*TELNET)
```

Alternatively, the Operations Navigator interface can be used. At the Operations Navigator screen, select **Network->Servers->TCP/IP**. If you want the Telnet server to start automatically, right-click **Telnet** and select **Properties**. Click the **General** tab, and check the **Start this server automatically when TCP/IP starts** box.

You also have to make sure that an interactive subsystem is active to run interactive jobs for Telnet sessions. The QSPL subsystem needs to be started to run printer passthrough sessions.

What is needed on the PC side

If you have installed Windows 95 or Windows NT 4.0, the Client Access for Windows 95/NT V3R1M3 client and Service Pack SF46891 or later should be installed for this to work. Client Access V3R2M0 supports printer emulation without any Service Pack. You can check Informational APAR II10918 for additional related information.

If you have Windows 3.1 installed, you have to make sure Client Access Enhanced for Windows 3.1 is installed. You can check Informational APAR II11226 to find the latest PTF numbers available.

Configuring a printer emulation session for Windows 95/NT

First, make sure your printer is configured and recognized by Windows. Most likely it will be configured and setup on LPT1. From the Windows desktop, select **My Computer**, and then select **Printers**. If there is no printer defined, select **Add Printer** and follow the instructions given by the wizard.

To get started with this new Client Access feature, you can configure and start a native TCP/IP printer session by following these steps:

1. Select **Start->Programs->IBM AS/400 Client Access->Accessories->Start or Configure Session**.

   Figure 308 on page 222 shows the window that appears.

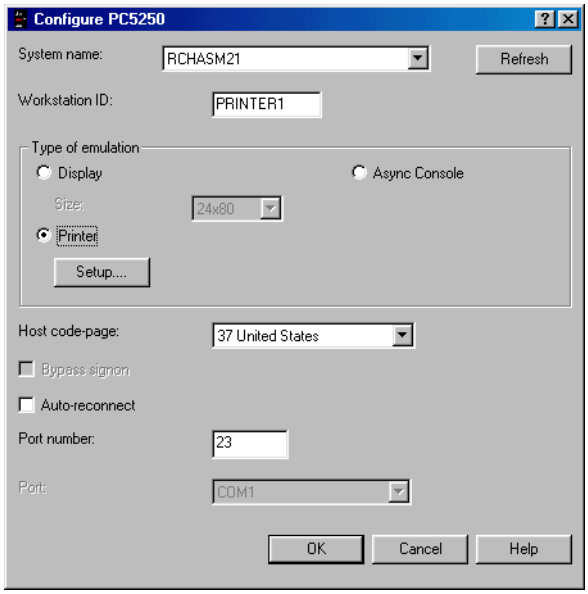


Figure 308. Configure PC5250 display

2. Select the name of an AS/400 system to connect to from the System name pull-down list.
3. Choose **Printer** for the type of emulation. You can use the Workstation ID field to specifically request an AS/400 virtual device name for the printer device or leave it blank and the Telnet server will then auto-select a compatible virtual device (QPADEVxxxx) and return the name on the printer control panel. This definition is the same as for a 5250 SNA printer emulation session and is saved in the 5250 section of the user's workstation profile, as are the SNA connections.
4. Click **Setup....** to bring up the PC5250 printer emulation set-up window as shown in Figure 309.

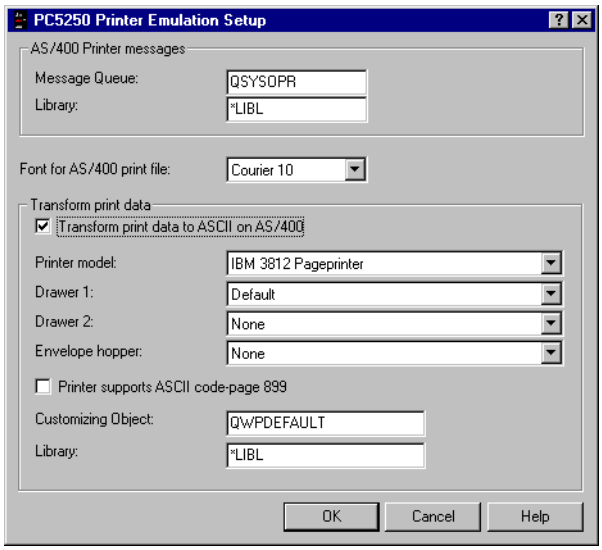


Figure 309. PC5250 Printer Emulation Setup

5. From the set-up screen, you may configure such items as font, the AS/400 message queue where printer messages are to be sent (the default will be QSYSOPR), and the “transform print data to ASCII on AS/400” (HPT) host function. If HPT is selected, this enables other configuration items, such as printer model and media tray selection options. There is also an auto-reconnect option, and an option to override the default AS/400 Telnet port number (23).
6. Click **OK**, and then click **OK** again at the Configure PC5250 panel. A host session is started, and the Session panel is started with an overlying printer status panel whose title bar contains the AS/400 system name and the printer ID (based on the ID you specified before). Otherwise, a system assigned device name will be displayed. This status panel is shown in Figure 310. The corresponding printer device is created on the AS/400 system.

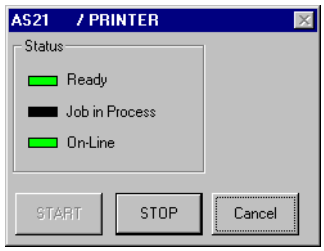


Figure 310. Printer status display

7. Select **Printer Setup** from the Session panel’s File menu as shown in Figure 311.

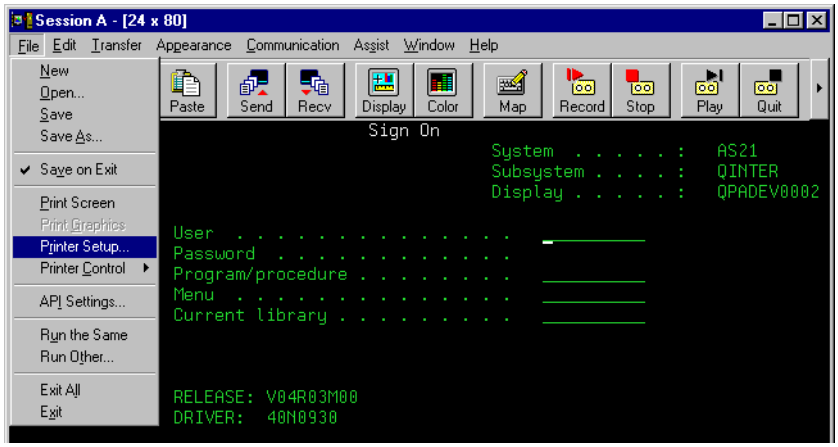


Figure 311. Printer Setup in the session’s panel file menu

8. At the Printer Setup screen shown in Figure 312 on page 224, check that the correct PC-attached printer is being highlighted as the default printer. Ignore the Setup button. Click **OK**.

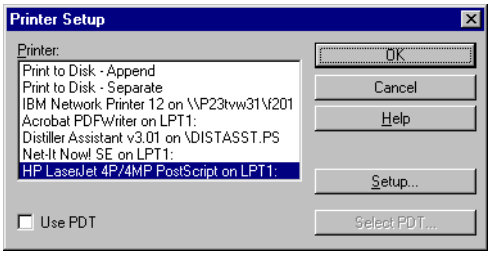


Figure 312. Printer Setup display

9. Back at the session panel, select **File->Save As**. At the Save Workstation Profile as window, shown in Figure 313, enter the printer ID in the File Name field. Select the folder in which you want to save the printer emulation profile.

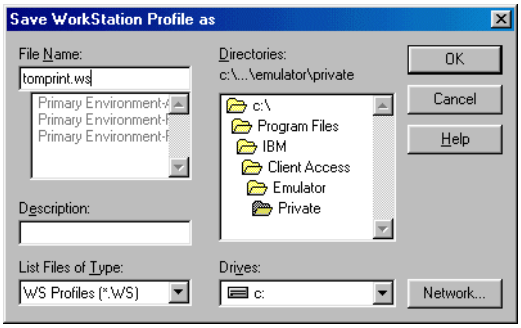


Figure 313. Save Workstation Profile as display

10. This filename should have a .WS extension. Click **OK**. You are then asked whether you want an icon and in what folder it should be placed.
11. The session is ended by selecting **Communication->Disconnect** from the menu bar.

Client Access printer emulation supports more than 200 PC printers, letting remote users obtain OS/400 spooled reports on the local printer that they have available.

5.2 AS/400 Telnet SSL Proxy

The AS/400 Telnet SSL Proxy is an application program that runs on OS/400 V4R2 or V4R3 in conjunction with the AS/400 Telnet server to provide secure Telnet connections between SSL-enabled Telnet clients and the AS/400 system.

AS/400 Telnet-based applications can now be run across networks with a higher degree of security.

5.2.1 AS/400 Telnet SSL Proxy basic principle

The Telnet SSL Proxy runs as one or more AS/400 jobs and listens on TCP/IP port 992 for connection requests from SSL-enabled clients.

Once the connection between the AS/400 system and client is established, the data flowing between the Telnet SSL Proxy and client is encrypted across the TCP/IP network:

- The Telnet SSL Proxy decrypts data from an SSL-enabled Telnet client and passes the data to the native Telnet server.
- The Proxy also encrypts data from the Telnet server and passes the data to the SSL enabled Telnet client.

This is illustrated in Figure 314.

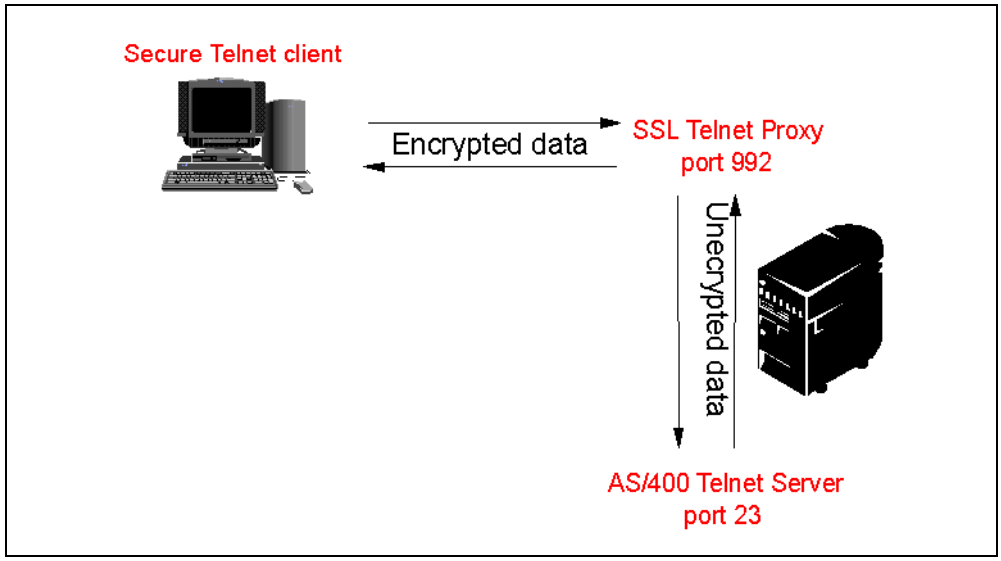


Figure 314. AS/400 Telnet SSL proxy data flow

Note

Be aware that the SSL Telnet Proxy server only interacts with the AS/400 Telnet server on the same AS/400 system.

5.2.2 Limitations and security considerations

The Telnet SSL Proxy server only runs on V4R2 and V4R3 systems. It refuses to run on earlier OS/400 releases. The Telnet SSL Proxy server must be deployed in a way that is consistent with the security policies and objectives of the customer.

The Telnet session data between the Telnet SSL Proxy server and the external Telnet clients is being encrypted. However, the Telnet session data between the Telnet SSL Proxy and the actual Native Telnet server on the AS/400 system is exchanged in the clear. This data is transmitted across a “loopback” TCP/IP connection.

Because the Telnet SSL Proxy works in conjunction with the AS/400 Native Telnet server on the same AS/400 system, all incoming traffic from both the loopback address and the Internet can reach that AS/400 system through the Native Telnet Server.

If the system administrator wants to block non-secure (unencrypted) Telnet sessions from being made, they can perform one of the following actions:

- Create and register a Telnet Initialization user exit program that will block all connections except those coming from the “loopback” address.
- Filter out all external Telnet requests to the AS/400 system via an external firewall, the AS/400 Firewall for AS/400, or use the IP Filtering and Network Address Translation capabilities of TCP/IP within V4R3 (see 10.8.3, “Scenario 2: Masquerading NAT and IP Packet Filtering” on page 415).

5.2.3 Distribution and packaging

The SSL Telnet Proxy is distributed as a save file that is available for download from the AS/400 Technical Studio Web page only (two versions of the save file are provided (one for V4R2 and one for V4R3)):
`http://www.as400.ibm.com/tstudio/tech_ref/tcp/sslproxy/index.htm`

You can find the following information on this Web page:

- Functional description of the Proxy intended to allow potential users to determine its use and make decisions on the applicability to their application environment
- Terms and conditions for its download and use
- Instructions on how to download and enable the Proxy
- Security guidelines
- Dependencies (the Cryptographic Access Provider support and 5769-TC1 must be installed)
- Feedback button

All support is packaged in an AS/400 save file that includes all necessary executable, as well as the documentation for use.

5.2.4 SSL Telnet Proxy server support

The SSL Telnet Proxy is distributed “as is” with the appropriate terms and conditions spelled out on the Web page listed in the previous section. IBM *does not* provide service and support for the SSL Telnet Proxy server. For detailed instructions, such as downloading the files and Proxy server setup, refer to the Web page at: `http://www.as400.ibm.com/tstudio/tech_ref/tcp/sslproxy/index.htm`

5.2.5 Client certificate

If you want to use SSL for secure communications, you must set up your AS/400 system to use digital certificates. You can use Digital Certificate Manager (DCM) with OS/400 V4R3 to create your own intranet certificate authority (CA) and use this also to issue certificates to servers and clients. DCM comes with several popular CA certificates in the server-default key ring.

In this section, we show you how to use DCM to certify your Telnet client running on a PC. For more information about DCM, refer to Chapter 3, “SSL security on the AS/400 system” on page 41.

Once a server has a digital certificate, SSL-enabled browsers can communicate securely with the server using SSL. A digital certificate is issued by a certificate

authority (also known as a CA). The DCM uses key-ring files to store digital certificates. The key-ring file also stores the certificate's private key.

1. To access the Digital Certificate Manager menu, go to the AS/400 Tasks panel (shown in Figure 315) by typing the following URL:
   `http://your.server.name:2001`

   Here, `your.server.name` is the fully qualified name of your host.

2. Since this is a controlled systems administration function, you are prompted for a user ID and password.

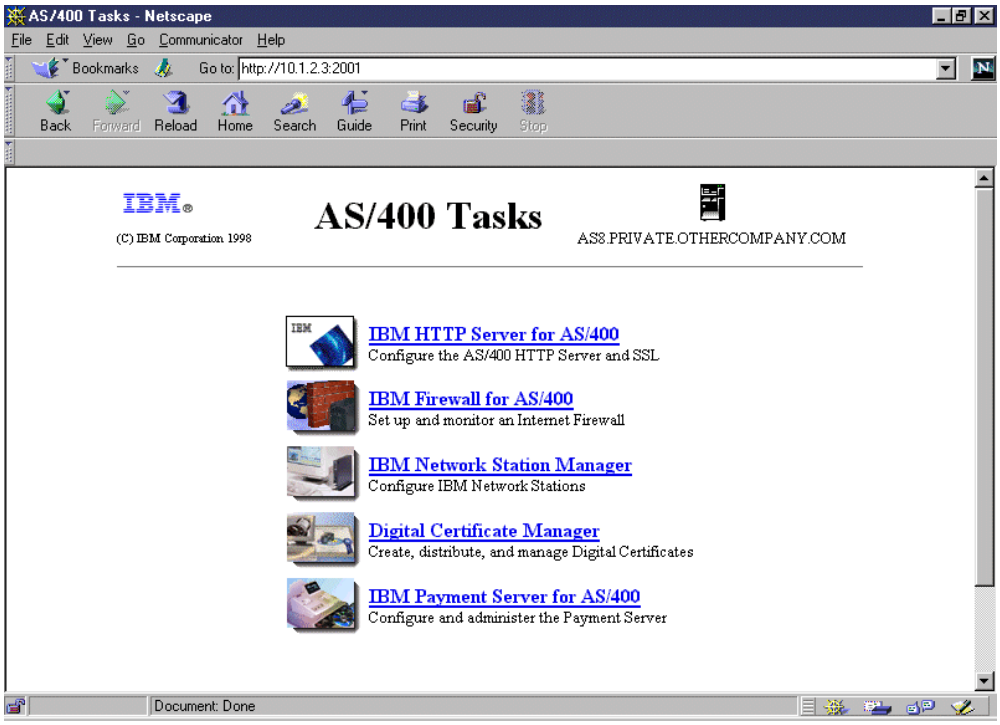


Figure 315. AS/400 Tasks panel

3. To access the Digital Certificate manager, click on the hyperlink for **Digital Certificate Manager** from the AS/400 Tasks page. You get the DCM page displayed in Figure 316 on page 228.

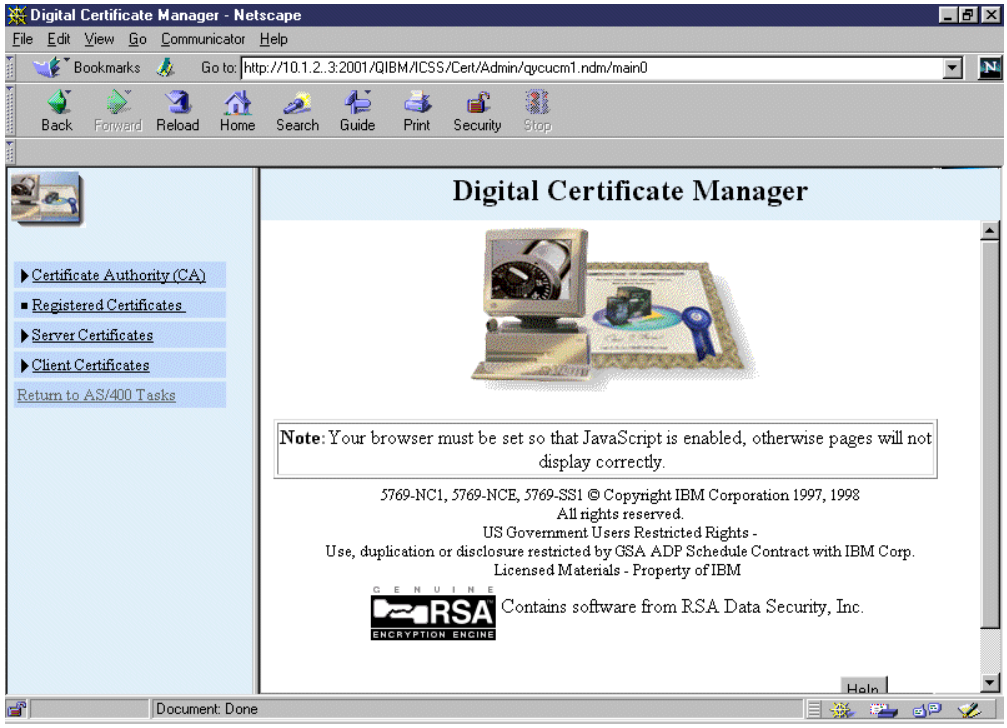


Figure 316. DCM page

4. Click **Client Certificate->Install CA certificate on your PC**.

   The window shown in Figure 317 appears.

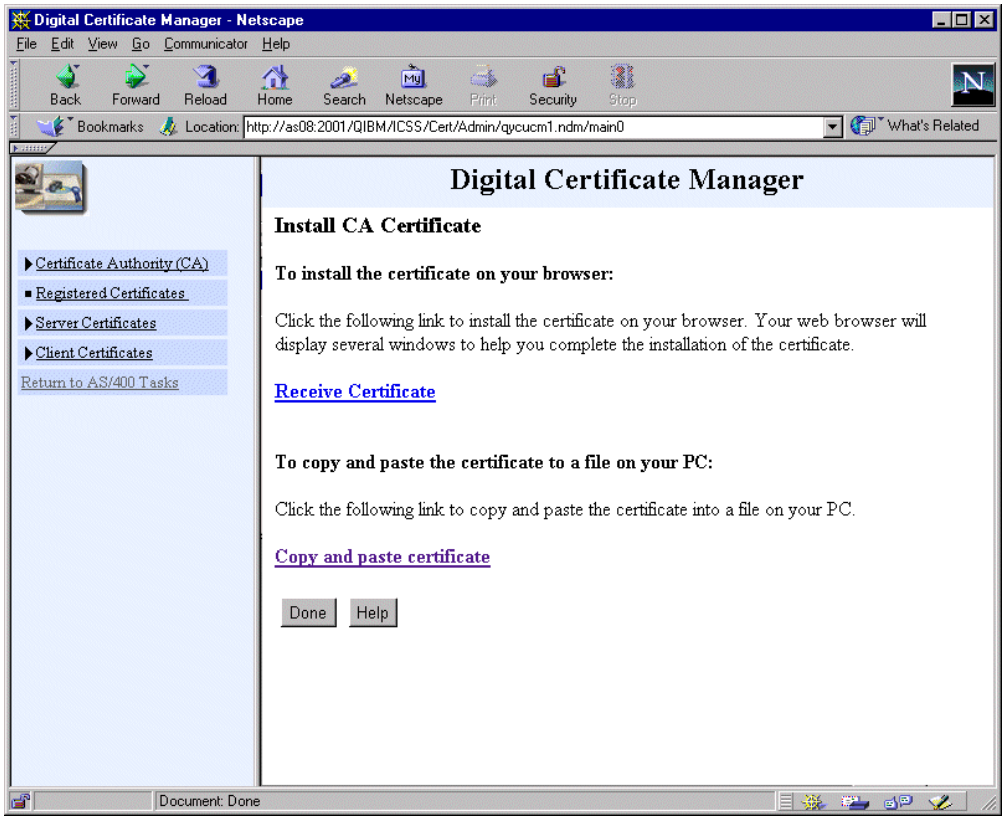


Figure 317. CA certificate install option display

5. Click **Copy and paste certificate**.
6. Select the lines shown in Figure 318 on page 230.

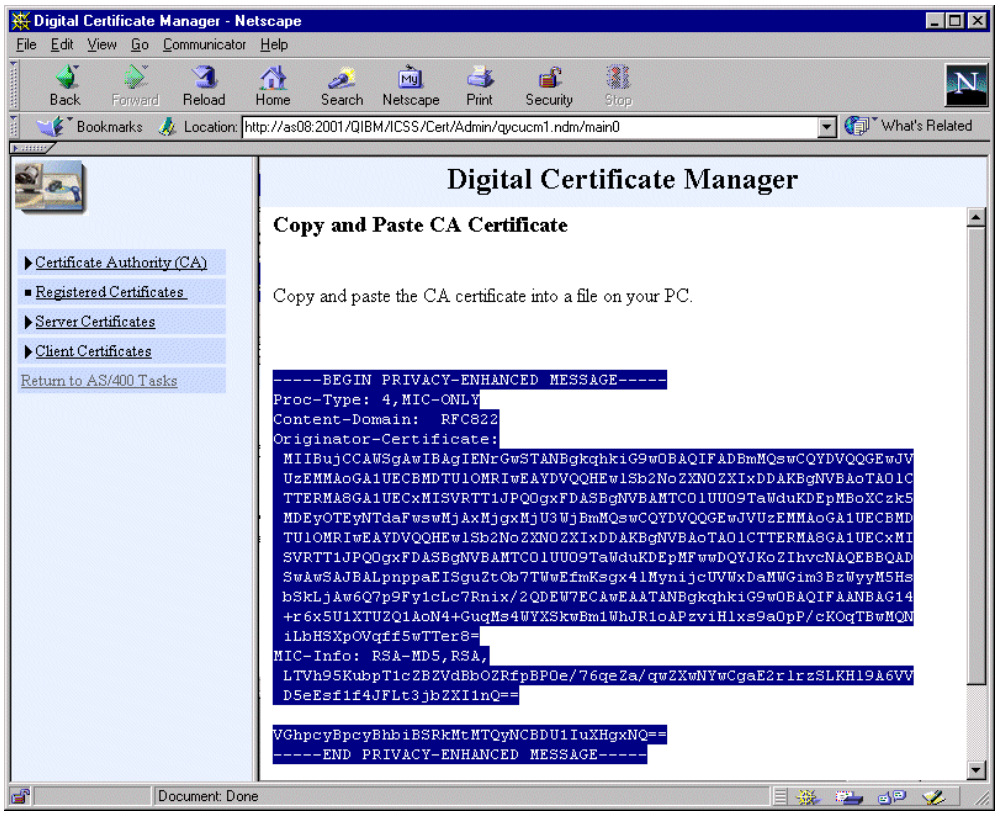


Figure 318. CA certificate

7. Press Ctrl+C at the same time to copy the lines to the clipboard.
8. Start the Notepad application on your PC to paste the data that was copied in the previous step. See Figure 319.

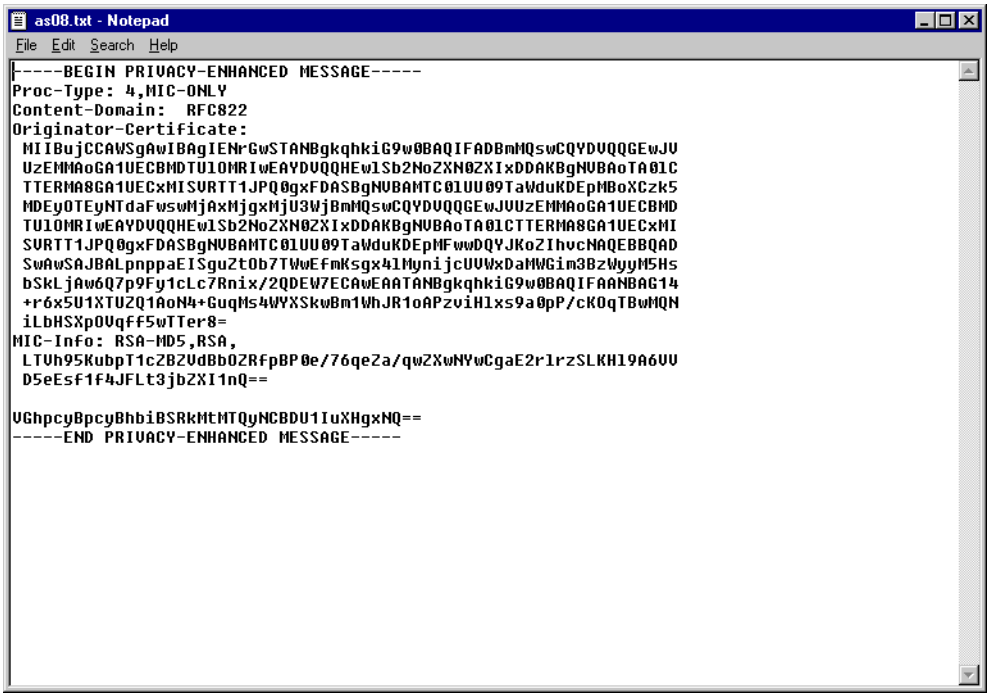


Figure 319. CA certificate copied in a text file

9. Save it as a text file. In this sample, it is as08.txt.

5.2.6 Telnet Proxy server setup

To setup the Telnet Proxy server, follow these steps:

1. Create a save file SSLTELNET in an AS/400 library with the command:

   ```
   CRTSAVF FILE(QGPL/SSLTELNET)
   ```

   Important

   The save file and the restore library are *not* the same. The library has the letter “Q” prefixed, but the save file does not.

2. Download the SSL Telnet Proxy server to the AS/400 from the Web page in the Technical Studio.
3. Restore the Telnet Proxy save file to QSSLTELNET library. Since the objects are owned by a QTCP user profile, and must be restored with these authorities intact, you may need *ALLOBJ and *SECADM authorities. Use the command:

   ```
   RSTLIB SAVLIB(QSSLTELNET) DEV(*SAVF) SAVF(QGPL/SSLTELNET)
   ```

   The restored objects are shown in Table 9 on page 232.
4. Give *USE authority to the user that administers the SSL Telnet Proxy Server. The user will need *USE authority to the following objects. In addition, the user running the Start SSL Telnet (STRSSLTELN) and End SSL Telnet

(ENDSSLTELN) commands requires *IOSYSCFG and *JOBCTL special authorities to start and end server jobs.

Table 9. QSSLTELNET library contents

Object name	Object type	Description
QSSLTELNET	*LIB	Telnet Proxy Server Library
STRSSLTELN	*CMD	Start SSL-Telnet Proxy Command
ENDSSLTELN	*CMD	End SSL-Telnet Proxy Command
TRCSSLTELN	*CMD	Set SSL-Telnet Proxy Trace Option Command
QZRDSTSTRW	*PGM	SSL-Telnet Proxy Start-up Program
QZRDSTSTRP	*PGM	SSL-Telnet Proxy Start-up POP Program
QZRDSTENDW	*PGM	SSL-Telnet Proxy Shut-down Program
QZRDSTTRCW	*PGM	SSL-Telnet Proxy Set Trace Option Program
QZRDSSLTN	*PF	SSL-Telnet Proxy Previous Start Parm File
QZRDSSLTN	*PNLGRP	SSL-Telnet Proxy Help

5. Give user profile QTCP *R authority to the server key-ring and stashed password files in the IFS file system. To assign QTCP user profile access to the SSL server certificates, use the commands:

```
CHGAUT OBJ('/ifs_directory/Server.kyr') USER(QTCP) DTAAUT(*R)

CHGAUT OBJ('/ifs_directory/Server.sth') USER(QTCP) DTAAUT(*R)
```

6. Check that the 127.0.0.1 *LOOPBACK interface is ACTIVE by using the command:

```
NETSTAT *IFC
```

   If 127.0.0.1 is INACTIVE, start it by using option 9.

   If 127.0.0.1 *LOOPBACK interface does not exist, add and start a 127.0.0.1 *LOOPBACK interface with the following commands:

```
ADDTCPIFC INTNETADR('127.0.0.1') LIND(*LOOPBACK) SUBNETMASK('255.0.0.0')
MTU(576)

STRTCPIFC INTNETADR('127.0.0.1')
```

7. Add QSSLTELNET to library list with the command:

```
ADDLIBLE QSSLTELNET
```

8. Start the Secure Telnet Proxy server with the command:

```
STRSSLTELN KEYRING('/ifs_directory/Server_certificate.kyr')
```

5.2.7 Work management related information

The library QSSLTELNET contains all the information related to the SSL Telnet Proxy Server.

The Proxy jobs are running in the QZRDSSLTN subsystem using user profile QTCP. The job description that the proxy jobs are using is QZRDSSLTN.

When the STRSSLTELN command is issued, the following actions occur:

- The QZRDSSLTN subsystem is started, if it is not already.
- A LISTEN job is started in the QZRDSSLTN subsystem.
- Initially, this results in one CHILD job being started.

This is illustrated in Figure 320.

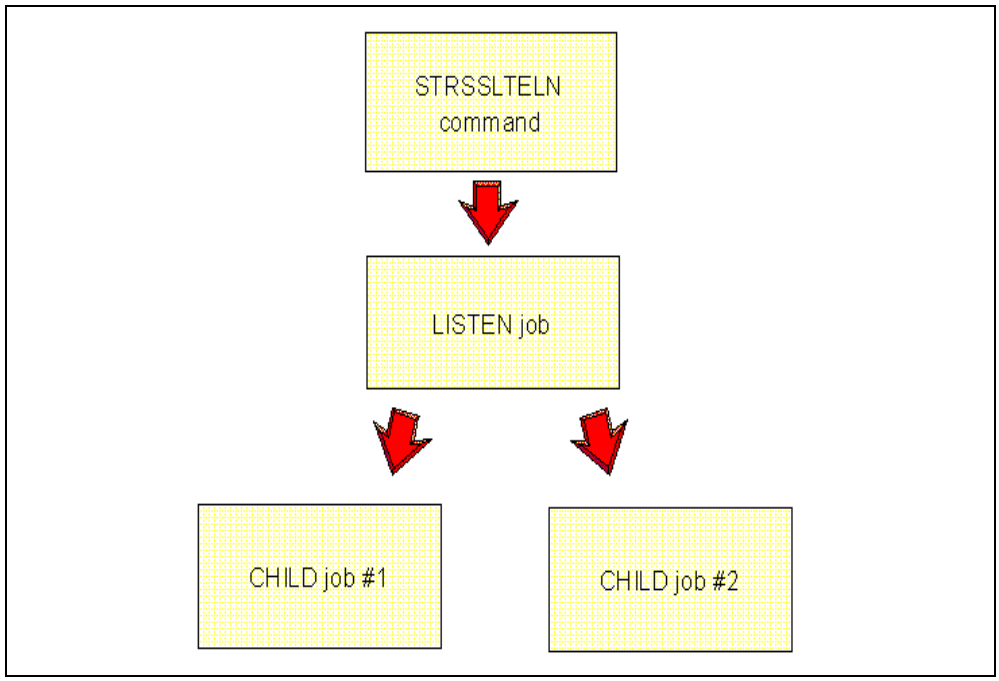


Figure 320. STRSSLTELN job flow

The LISTEN job accepts client connections on port 992. As each CHILD job is handling a maximum of 20 client connections, additional CHILD jobs are started by the LISTEN job, if necessary.

The CHILD job opens up one connection to the Telnet server for each client connection. This job encrypts and decrypts the data and passes between the client and the real Telnet server.

5.2.8 Available SSL clients

Some of the secure SSL Telnet clients available are:

- **IBM eNetwork HostOnDemand 3.0 or 4.0**: This is used in conjunction with a Web browser. This client does not support device naming or printers.
- **PCOMM 4.3**: This should be used when 3270 emulation is required.
- **Client Access Express**: This should be used when only 5250 emulation is required. It does not require a browser to run. Device naming and printer support is also available with this client.

5.2.9 Starting the SSL Telnet Proxy server

The SSL Telnet proxy server can be started by issuing the following command:

```
STRSSLTELN
```

The command related parameters are:

- **KEYRING**: The key-ring file specifies the path and file name of the key-ring file from which the certificate and private key are obtained for all SSL sessions uses by the SSL Telnet Proxy.
- **PASSWORD**: This parameter specifies the password to the SSL key-ring file. The possible values for this parameter are:
  - **STASHED* means that the password should be extracted from a stashed key-ring password file. This file is created by user selection during the key-ring configuration. This special value is the default and its use is strongly recommended.
  - *Password* is a mixed case password to the key-ring file.
- **TRACE**: This parameter specifies the SSL Telnet Proxy trace option setting.
  - **OFF* means the trace option setting is turned off. The data handled by the SSL Telnet Proxy will not be traced. The current trace information remains in the trace file.
  - **ON* means the trace option setting is turned on. The data handled by the SSL Telnet Proxy will be traced and retained in trace source file members. The information may also include SSL Proxy messages for problem determination purposes. TRACE is the name of the Proxy trace file that can be found in the QSSLTELNET library. If the trace has been active, the file TRACE will exist and contain multiple file members. The file member LISTEN is being used by the SSL Telnet Proxy listening job. This member may contain references to other trace file members. The file members named CHILD00000, CHILD00001, CHILD00002, ... are used by the corresponding proxy run jobs. All the trace file records are time stamped.

> **Note**
>
> Only a single level of trace information is being retained. When the STRSSLTELN command is used with the TRACE(*ON) option, the current trace information will then be discarded. The user may retain trace information by copying the trace file to another source file before setting on the trace option.

5.2.10 Ending the SSL Telnet Proxy server

The SSL Telnet proxy server can be ended by issuing the following command:

```
ENDSSLTELN
```

There are no parameters related to this command.

5.2.11 WRKACTJOB SBS(QZRDSSLTN)

When the SSL Telnet Proxy server is started, you should find the following jobs running within the QZRDSSLTN subsystem (Figure 321).

```
                          Work with Subsystem Jobs                  AS8
                                                              11/23/98  09:30:03
Subsystem  . . . . . . . . . . :   QZRDSSLTN

Type options, press Enter.
  2=Change   3=Hold   4=End   5=Work with   6=Release   7=Display message
  8=Work with spooled files   13=Disconnect


Opt  Job         User        Type     -----Status-----  Function
     LISTEN      QTCP        BATCH    ACTIVE            PGM-QZRDSTLIS
     QZRDSTRUN   QTCP        BATCHI   ACTIVE

                                                                      Bottom
Parameters or command
===>
F3=Exit   F4=Prompt   F5=Refresh   F9=Retrieve   F11=Display schedule data
F12=Cancel
```

Figure 321. Jobs running in the QZRDSSLTN subsystem

Figure 322 shows what you should have within the job log of the LISTEN job running in the QZRDSSLTN subsystem.

```
                             Display Job Log
                                                          System:   AS8
Job . . :   LISTEN       User . . :   QTCP          Number . . . :   013826

 >> CALL PGM(QSSLTELNET/QZRDSTLIS)

                                                                      Bottom
Press Enter to continue.

F3=Exit   F5=Refresh   F10=Display detailed messages   F12=Cancel
F16=Job menu           F24=More keys
```

Figure 322. Listen job log

The job log of QZRDSTRUN should appear as shown in Figure 323 after using the STRSSLTELN command by specifying TRACE (*ON).

```
                             Display Job Log
                                                          System:   AS8
Job . . :   QZRDSTRUN    User . . :   QTCP          Number . . . :   013827

    Job 013827/QTCP/QZRDSTRUN started on 11/23/98 at 09:29:52 in subsystem
      QZRDSSLTN in QSSLTELNET. Job entered system on 11/23/98 at 09:29:52.
    Member CHILD00000 added to file TRACE in QSSLTELNET.


                                                                      Bottom
Press Enter to continue.

F3=Exit   F5=Refresh   F10=Display detailed messages   F12=Cancel
F16=Job menu           F24=More keys
```

Figure 323. QZRDSTRUN job log

```
Columns . . . :    1  71          Browse                      QSSLTELNET/TRACE
 SEU==>                                                          LISTEN
 FMT **  ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
       *************** Beginning of data ************************************
0001.00 11/23/98 09:29:53   Listen job 013826/QTCP/LISTEN Started
0002.00 11/23/98 09:29:53   Child job 0 Started
0003.00                     Job : 013827/QTCP/QZRDSTRUN
0004.00                     Log File/Member : TRACE/CHILD00000
0005.00 11/23/98 10:53:51   Connection received from 10.1.2.3  at port 1033
0006.00                     Connection being processed by child job 0
0007.00 11/23/98 10:54:04   Connection received from 10.2.3.43 at port 1034
0008.00                     Connection being processed by child job 0



0041.00 11/23/98 10:57:48   Connection received from 10.4.5.63 at port 1051
0042.00                     Connection being processed by child job 0
0043.00 11/23/98 10:57:57   Connection received from 10.5.6.7 at port 1052
0044.00                     Connection being processed by child job 0
0045.00 11/23/98 10:57:57   All existing child jobs are full.
0046.00 11/23/98 10:58:01   Child job 1 Started
0047.00                     Job : 013842/QTCP/QZRDSTRUN
0048.00                     Log File/Member : TRACE/CHILD00001
       ****************** End of data ***************************************
 F3=Exit   F5=Refresh   F9=Retrieve   F10=Cursor   F11=Toggle   F12=Cancel
 F16=Repeat find        F24=More keys
```

Figure 324. TRACE file output

In Figure 324, a message is sent telling that all existing child jobs are full. We get a message telling that new child job (child job 1) got started. This was explained earlier. Each CHILD job is handling a maximum of 20 client connections, additional CHILD jobs are started by the LISTEN job if necessary.

The output of the `netstat *cnn` command should appear similar to the example in Figure 325.

```
                    Work with TCP/IP Connection Status
                                                                System:   AS8
Local internet address  . . . . . . . . . . . :   *ALL

Type options, press Enter.
  4=End   5=Display details

     Remote           Remote     Local
Opt  Address          Port       Port       Idle Time  State
     *                *          992        000:51:43  Listen
     *                *          1196       117:44:58  *UDP
     *                *          1453       000:13:35  *UDP
     *                *          as-admi >  000:37:57  Listen
     *                *          as-cent >  000:38:33  Listen
     *                *          as-data >  069:14:57  Listen
     *                *          as-dtaq    092:07:02  Listen
     *                *          as-file    087:24:52  Listen
     *                *          as-netprt  089:46:38  Listen
     *                *          as-rmtcmd  000:38:33  Listen
     *                *          as-signon  000:38:40  Listen
                                                                       More...
F5=Refresh   F11=Display byte counts   F13=Sort by column
F14=Display port numbers   F22=Display entire field   F24=More keys
```

Figure 325. Work with TCP/IP Connection Status screen: Port 992

Local port 992 shows up in a *listen* state. This is the port on which the SSL Telnet Proxy is listening.

```
                    Work with TCP/IP Connection Status
                                                                System:   AS8
Local internet address  . . . . . . . . . . . :   *ALL

Type options, press Enter.
  4=End   5=Display details

     Remote           Remote     Local
Opt  Address          Port       Port       Idle Time  State
     10.1.2.3         1074       telnet     000:00:00  Established
     10.1.1.4         1345       telnet     000:06:42  Established
     127.0.0.1        telnet     1806       000:00:10  Established
     127.0.0.1        1806       telnet     000:00:10  Established


Bottom
 F5=Refresh   F11=Display byte counts   F13=Sort by column
 F14=Display port numbers   F22=Display entire field   F24=More keys
```

Figure 326. Work with TCP/IP Connection Status screen: Loopback interface 127.0.0.1

Figure 326 shows the 127.0.0.1 loopback interface in an *established* state. This is the interface that is used to exchange the data between the Telnet SSL Proxy and the actual Native Telnet server on the AS/400 system.

5.2.12 Installing Netscape 4.06 Client with IBM HostOnDemand

Install the Netscape Version 4.06 or later, or Microsoft Internet Explorer 4.01 or later on your workstation. If you installed the Netscape Version 4.06, you can skip this step. Otherwise, ensure you have installed the Java plug-in 1.1. Complete this process:

1. Install IBM HostOnDemand on your workstation as instructed at the following Web site: `http://www.software.ibm.com/enetwork/hostondemand`
2. Start IBM HostOnDemand by selecting **Start->Programs->IBM eNetwork OnDemand->Administration->Key Management**.
3. Select **Key Database File->New** as shown in Figure 327.

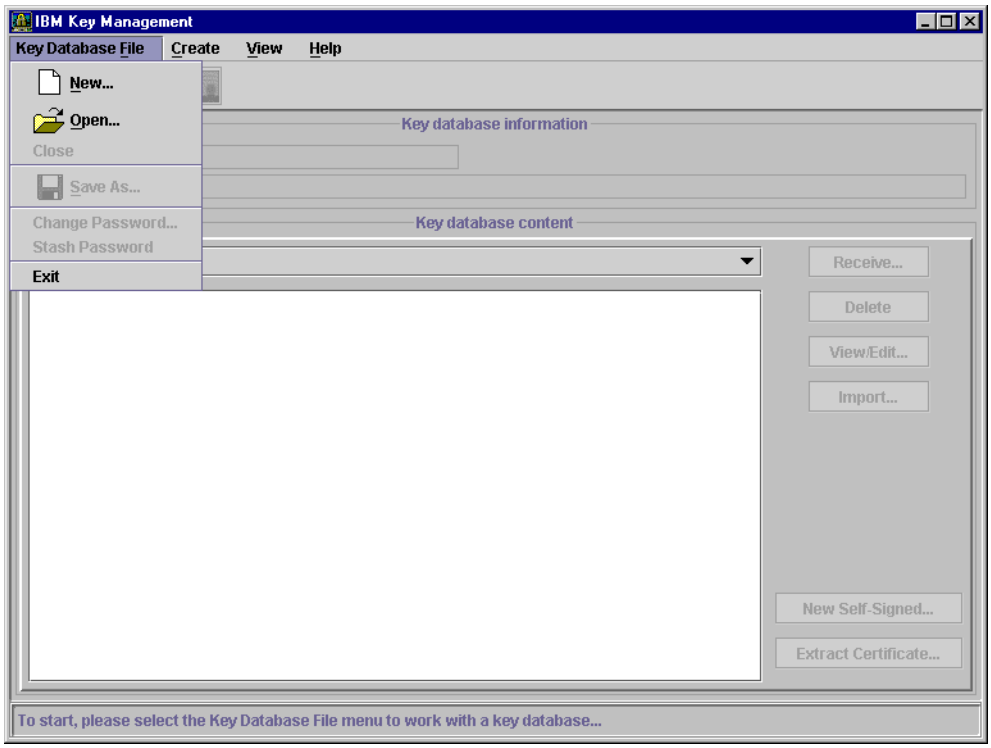


Figure 327. IBM Key Management screen

4. Type `temp.kdb` for File Name, and click **OK** (Figure 328). The temp.kdb is created in the c:\ondemand\bin directory.

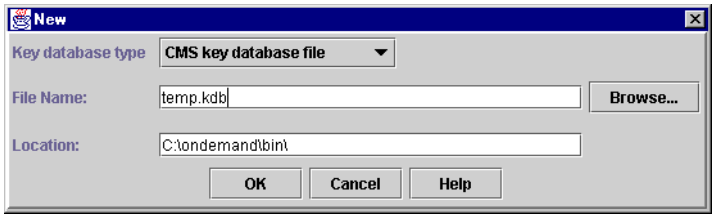


Figure 328. New key database type display

5. Type a password, and click **OK** (Figure 329).

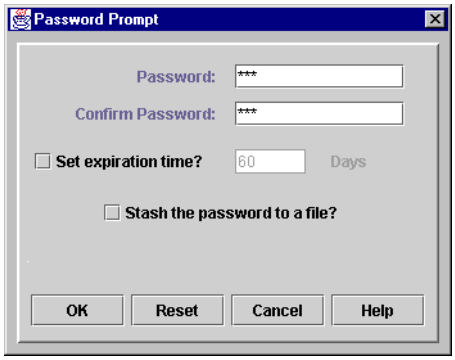


Figure 329. Password Prompt

The following display appears (Figure 330).

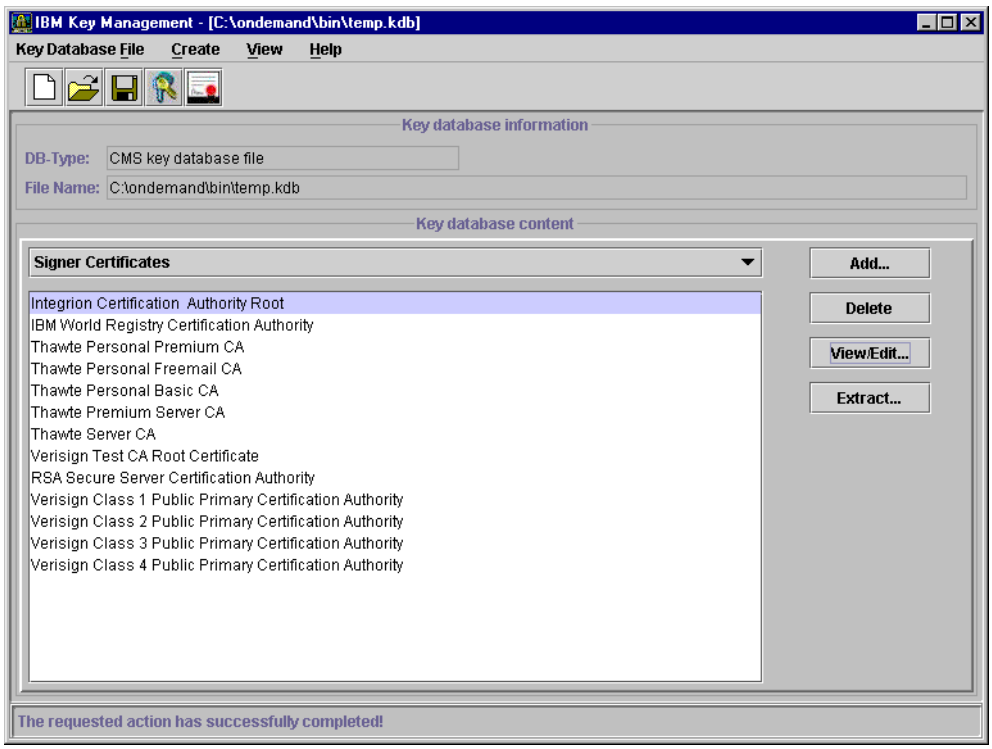


Figure 330. IBM Key Management display

6. Click **Add**.
7. In the dialog box that appears, as shown in Figure 331, type `*.*` for Certificate File Name, and click **Browse**.

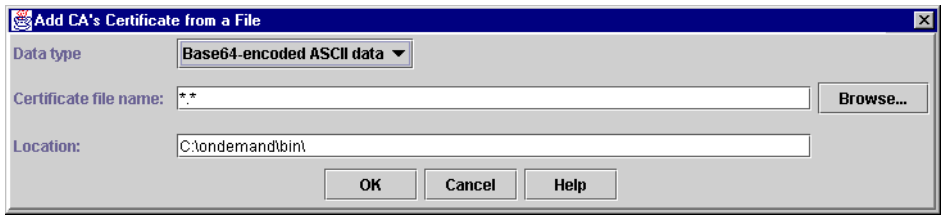


Figure 331. Add CA's Certificate from a File display

8. Locate the CA certificate file you created in 5.2.5, “Client certificate” on page 226. Click the file name of the certificate file, which was as08.txt in the previous example. Click **OK**.
9. Figure 332 shows the window that appears. Type the label name to identify the certificate. A choice is the host name of he AS/400 server. Click **OK**.

Figure 332. Certificate label display

10. Check that the certificate is highlighted in the list. Click **Extract** (Figure 333).

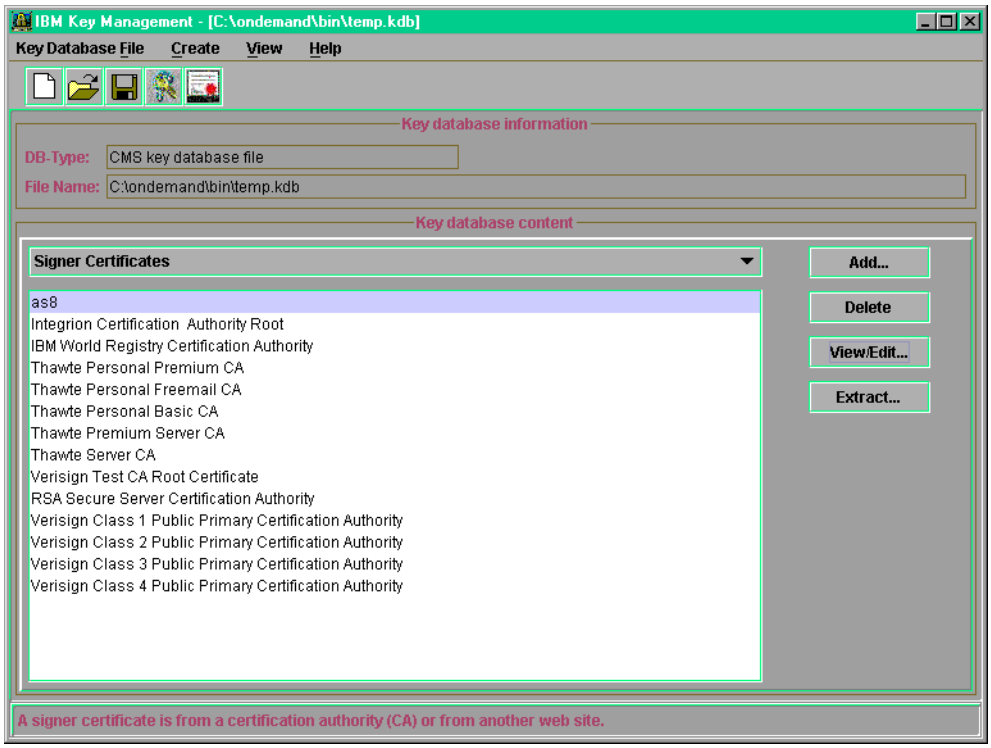


Figure 333. IBM Key Management display

Figure 334 shows the screen that appears.

11. Select **Binary DER data** for Data type.
12. For Certificate file name, type: `temp.der`
13. For Location, enter: `c:\ondemand\bin`

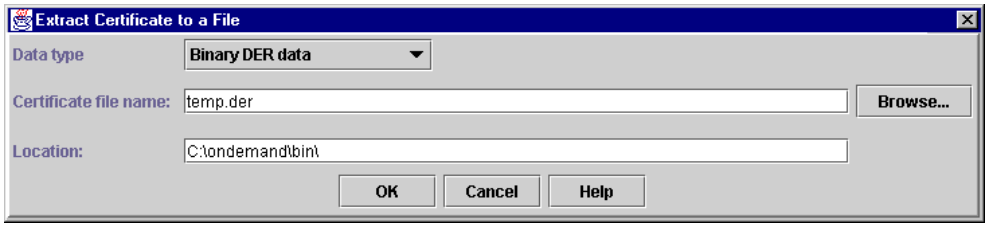


Figure 334. Extract Certificate display

14.Click **OK**. The temp.der is created in the directory c:\ondemand\bin.

15.Select **Key Database File->Exit** (Figure 335).

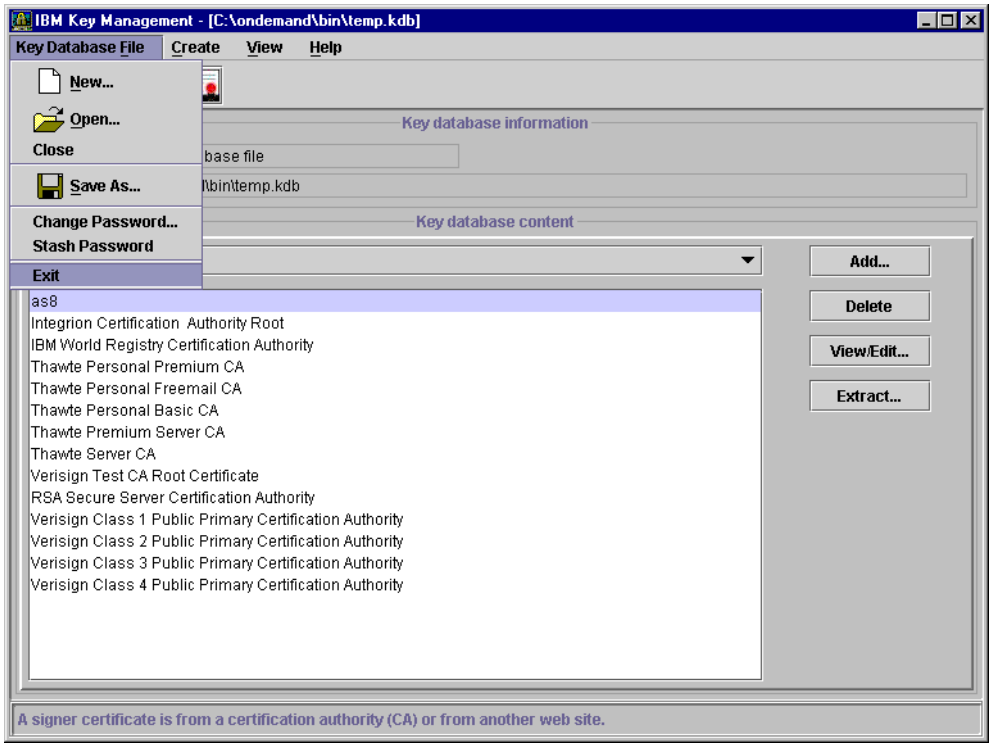


Figure 335. IBM Key Management display

Now, start an MS-DOS session.

16.Type: `cd c:\ondemand\lib`

17.Type: `keyrng CustomizedCAs add --ca c:\ondemand\bin\temp.der`

The certificate information is added to the CustomizedCAs file. The file is created, if it does not already exist.

5.2.13 Starting IBM HostOnDemand in Netscape 4.06

Complete these steps:

1. Click **Start->Programs->IBM eNetwork On-Demand->Host On-Demand 3.0->Host On-Demand**. This starts the Netscape browser and loads the file HOD_en.html (Figure 336 on page 242).

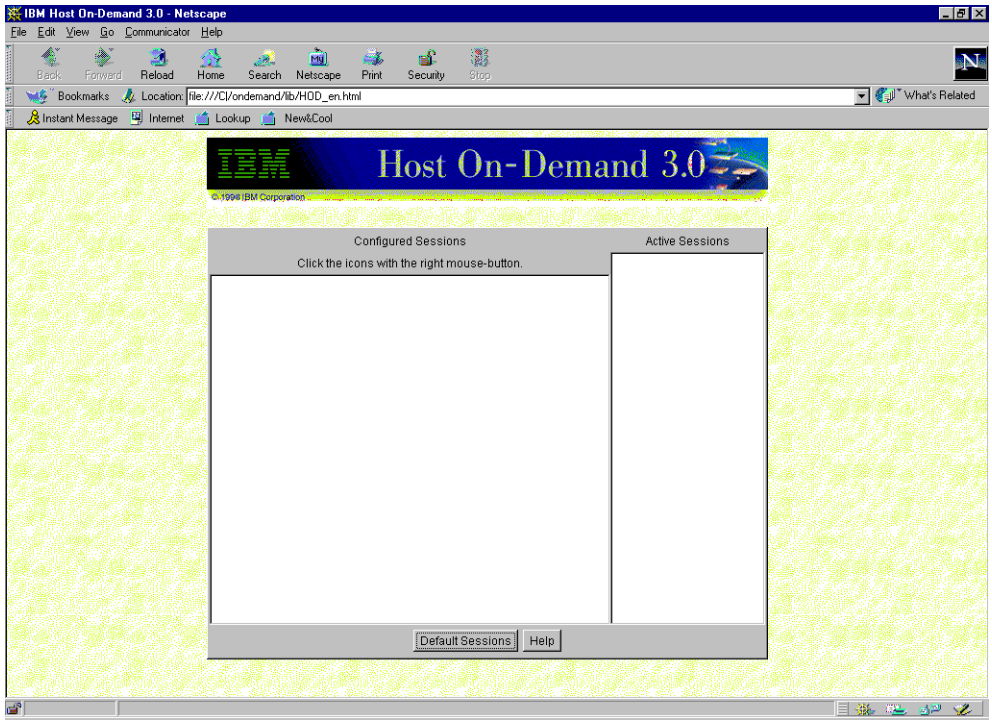


Figure 336. Host On-Demand 3.0 display

2. Click **Default sessions**. When the window shown in Figure 337 appears, right-click on the **5250** icon, and click **Copy**.

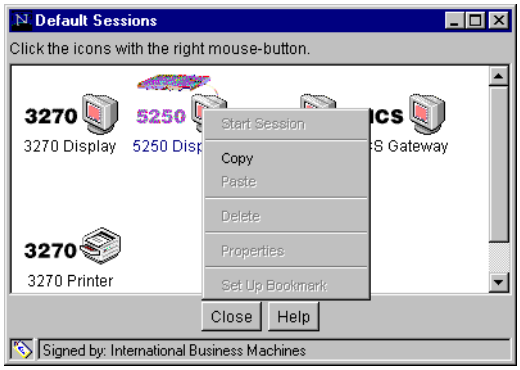


Figure 337. Default Sessions

3. A window appears for you to enter the host name and port (Figure 338)
   a. Type the TCP/IP host name of the AS/400 server for Destination Address.
   b. Type `992` for Destination Port.

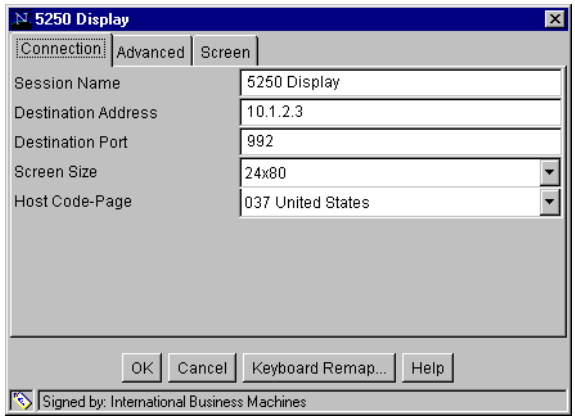


Figure 338. 5250 Display screen

4. Click the **Advanced** tab (Figure 339):

   a. Select **ON** for Enable Security (SSL).
   b. Optionally, select **ON** for server authentication.

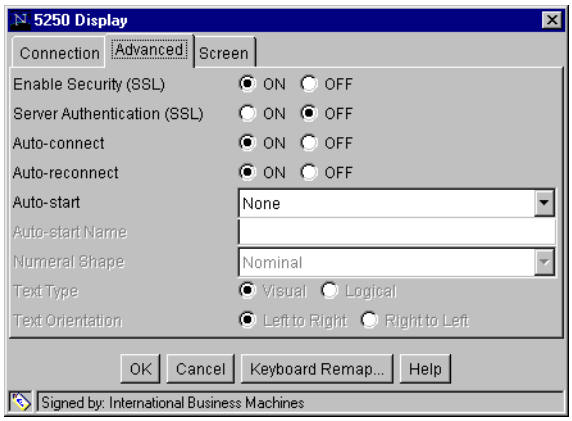


Figure 339. 5250 Display: Advanced

5. Click **OK**. The session icon is displayed in your browser session now (Figure 340 on page 244).

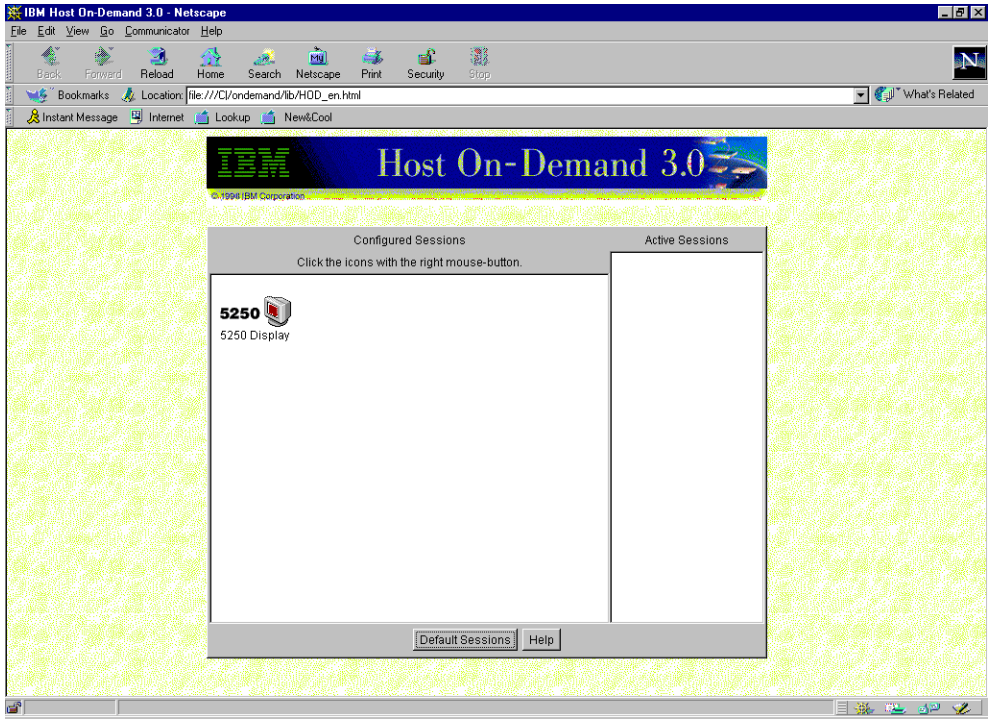


Figure 340. Host On-Demand 3.0

6. Double-click the **5250** session icon (or right-click, and select **Start Session**).
7. A 5250 session is established, and you get a secured 5250 session being established as shown in Figure 341.

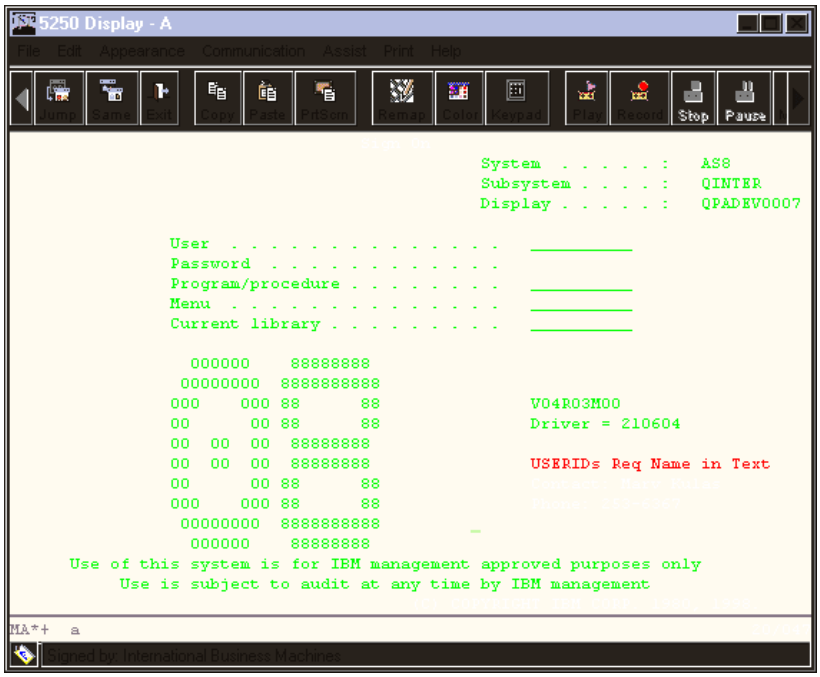


Figure 341. 5250 session display

The 5250 display session is started, and an active 5250 session shows up as shown on the right-side of the page shown in Figure 342.

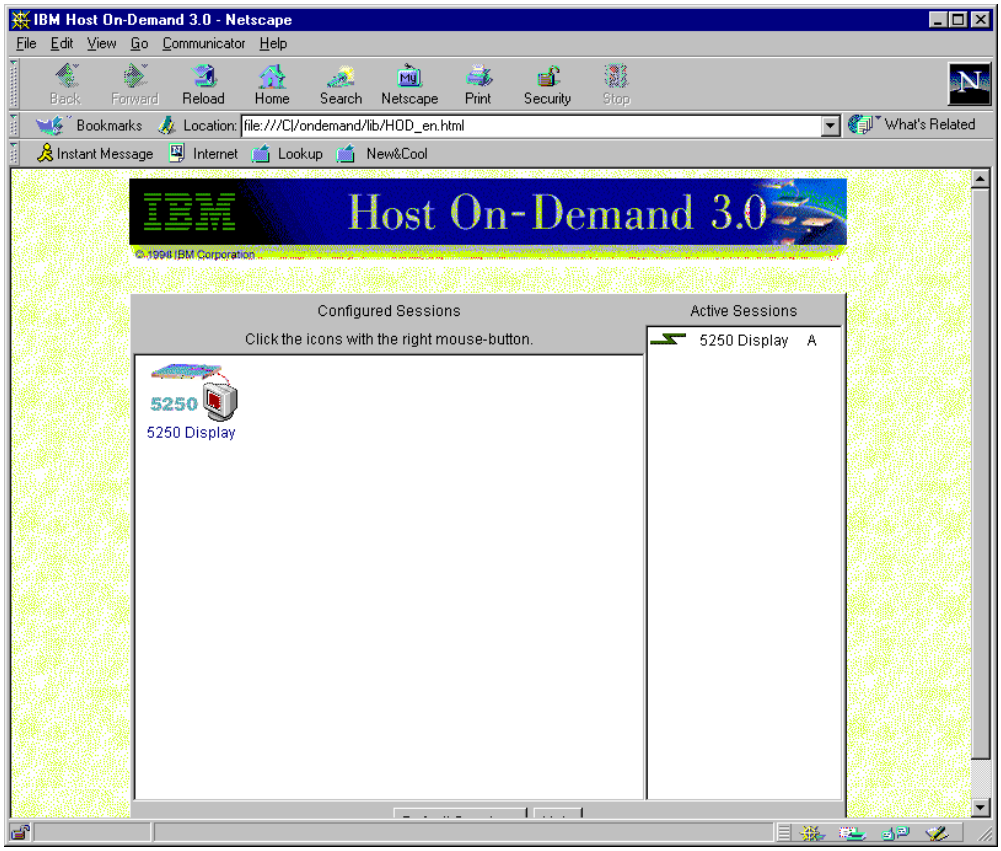


Figure 342. Host On Demand 3.0 screen: Active session

5.2.14 Setting up IBM Personal Communications 4.3

Complete this series of steps:

1. Start the IBM Personal Communications. Click **Start->Programs->IBM Personal Communications->Utilities->Certificate Management**.
2. Select **Open** in the Key Database File pull-down menu (Figure 343 on page 246).

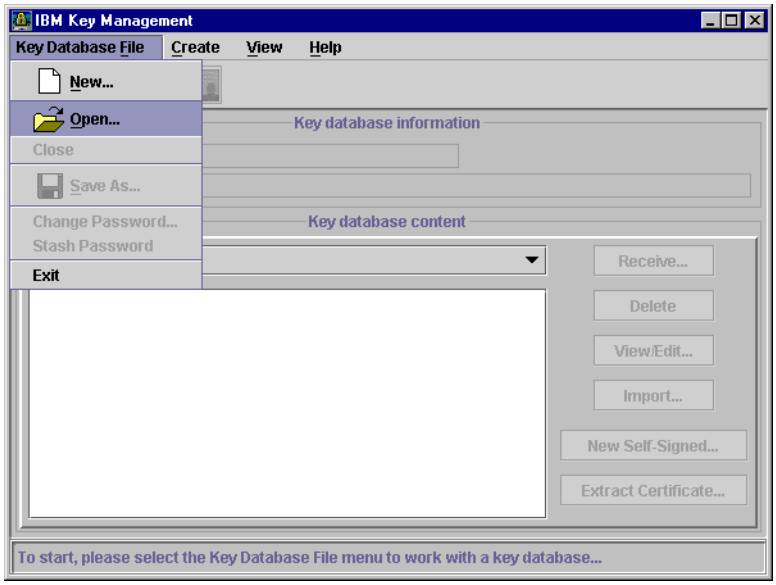


Figure 343. IBM Key Management display

3. Select **PComClientKeyDb.kdb** (Figure 344).

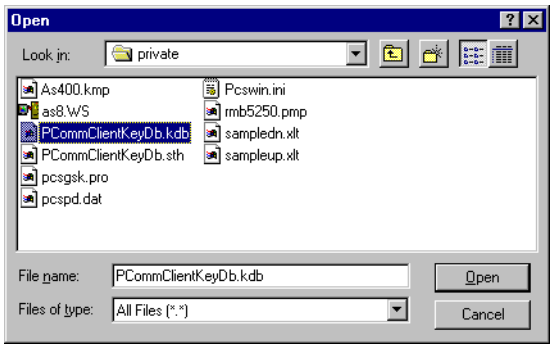


Figure 344. PComClientKeyDb.kdb

4. Type pcomm for password, if the password has not already been changed (Figure 345).

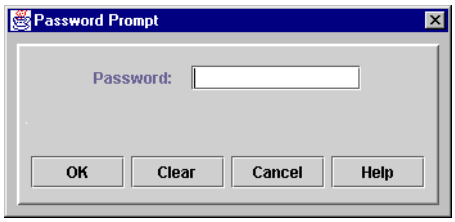


Figure 345. Password Prompt

5. In the list with the title Key Database Content shown in Figure 346, select **Signer Certificates**.
6. Click **Add**.

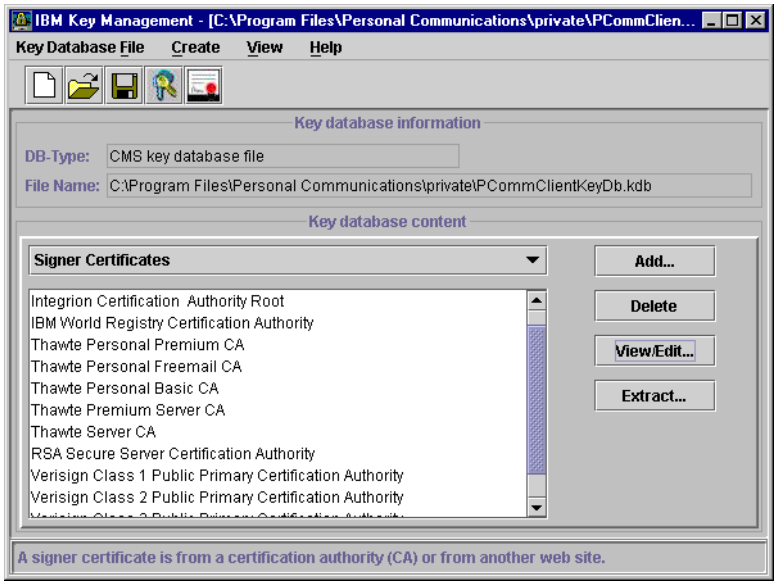


Figure 346. Signer Certificates

7. In the pull-down list, select **Binary DER data** (Figure 347).

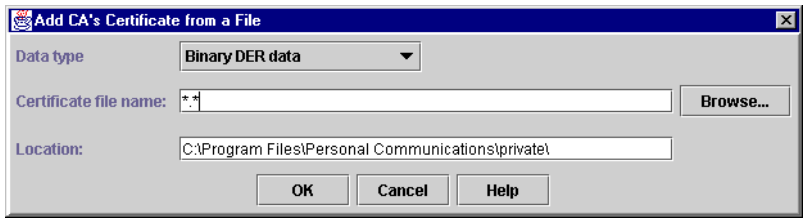


Figure 347. Binary DER data display

8. Click **Browse**, and find location of the as08.txt file that was created in 5.2.5, “Client certificate” on page 226.
9. Click the file name.
10. Click **Open**. The display shown in Figure 348 appears.
11. Type a label for the certificate (usually the system name), and click **OK**.

Figure 348. Label display

12. Click **OK**.

    The label is in the list of certificates.

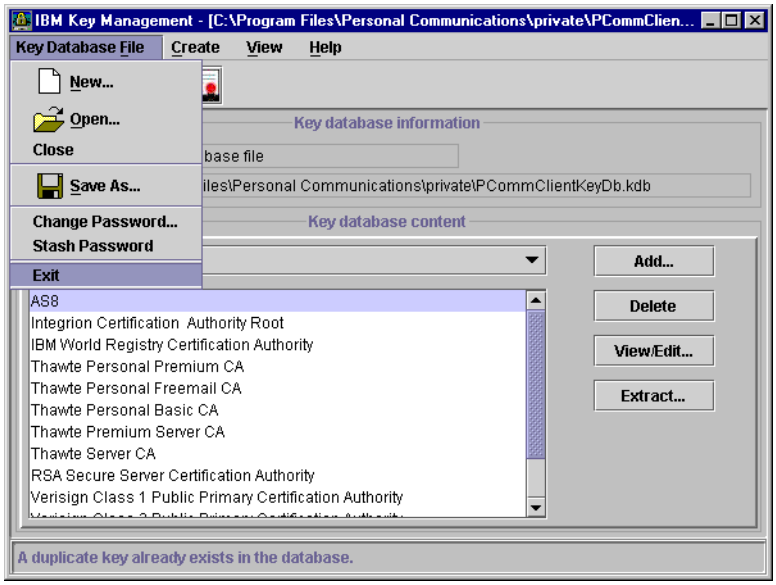


Figure 349. Exit window

13.Select **File->Exit**.

5.2.15 Configuring IBM Personal Communications 4.3

To configure IBM Personal Communications 4.3 to use SSL, perform the following steps:

1. Click **Start->Programs->IBM Personal Communications->Start or Configure session**. Click **OK** at the Welcome window (Figure 350).

Figure 350. Welcome window

The configuration screen appears as shown in Figure 351.

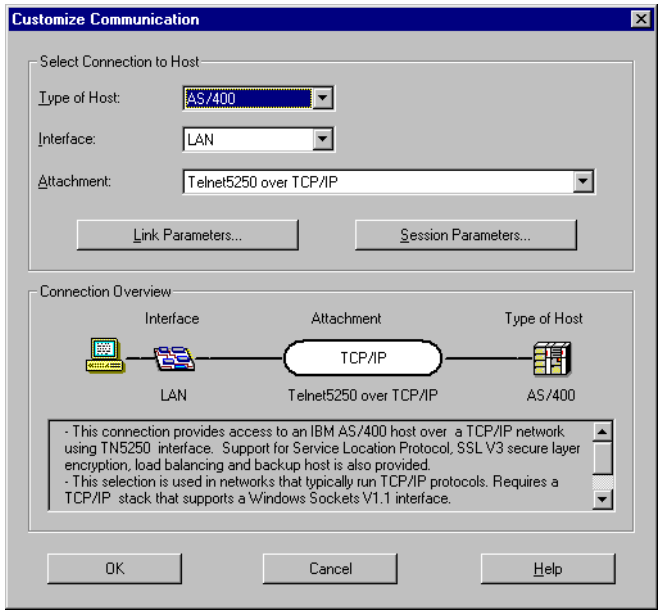


Figure 351. Customize Communication display

2. Select **AS/400** for Type of Host.
3. Select **LAN** for Interface.
4. Select **Telnet 5250 over TCP/IP** for Attachment.
5. Click **Link Parameters**. A window appears as shown in Figure 352.
6. Type in the host name or IP address for Primary host.
7. Type `992` for Port Number.
8. Select **Enable Security**.

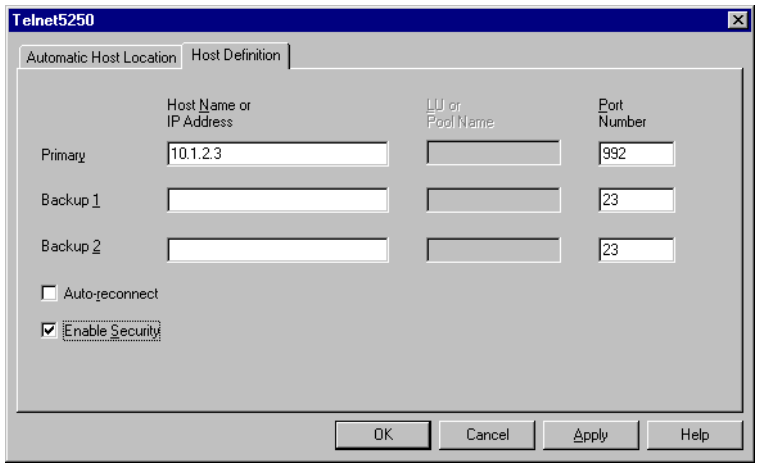


Figure 352. Link Parameters display

9. Click **OK**.
10. Click **OK** again.

    A secure AS/400 logon screen should be displayed now as shown in Figure 353 on page 250.

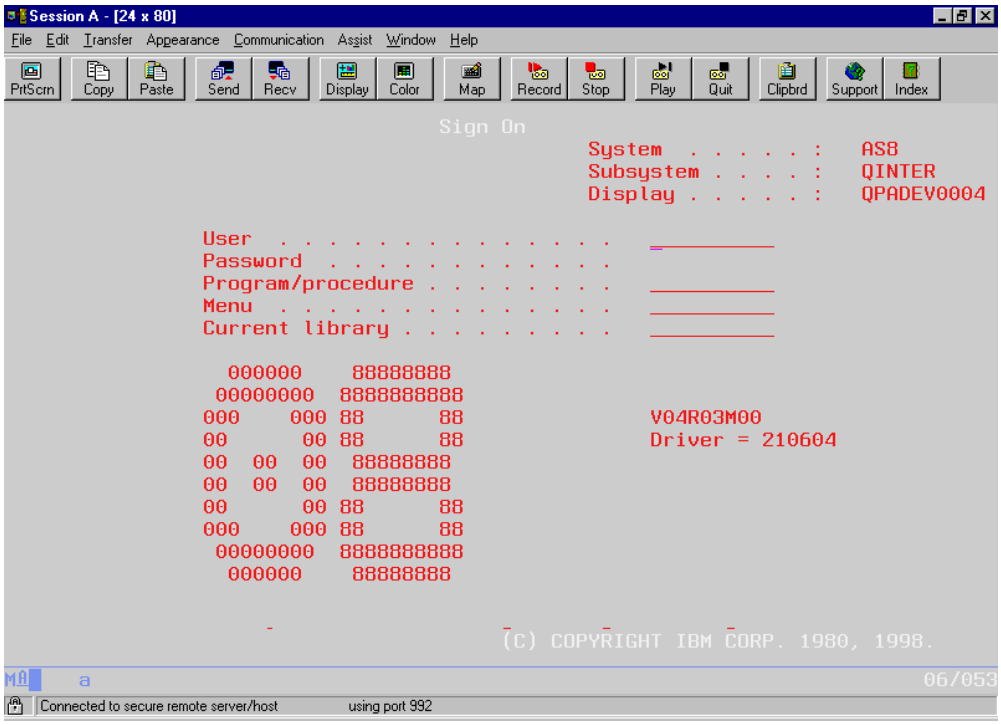


Figure 353. Secure 5250 Session display

The message `Connected to secure remote server/host, using port 992` is displayed at the bottom of your screen.

5.3 Changing the Telnet server port on the AS/400 system

As shipped, the Telnet server uses the default port 23 rather than the port number specified in the Service Database Table. The Telnet server may be made to use the port number from the Services Database Table. To enable this feature of the Telnet server, the system administrator must change a switch setting. To activate the function, issue the following command:

```
CALL PGM(QTCP/QTGSRV) PARM(*PORT)
```

To de-activate the function (the default setting), issue this command:

```
CALL PGM(QTCP/QTGSRV) PARM(*NOPORT)
```

5.3.1 Ending the Telnet server

You should first end the Telnet server that is running on the AS/400 system by typing the following command:

```
ENDTCPSVR SERVER(*TELNET)
```

5.3.2 The WRKSRVTBLE command

Note

This support is available for V3R2, V3R7, V4R1, V4R2, and V4R3 with the PTFs. For detailed PTF information, refer to Informational APAR SA79069.

The service table is used to manage the mapping of network services to ports and to record the protocols the services use. You can use the Work with Service Table Entries command to add new service table entries or to remove them. The service table contains a list of well-known network services and the port and protocol each service uses. Figure 354 shows the screen you get after you enter the WRKSRVTBLE command. There are two entries in this table for Telnet: one for UDP and one for TCP. You can remove those entries for Telnet by entering option 4 in front of both of them.

```
                    Work with Service Table Entries
                                                          System:   AS1
Type options, press Enter.
  1=Add   4=Remove   5=Display

Opt  Service                                Port  Protocol

     src                                     200  udp
     sunrpc                                  111  tcp
     sunrpc                                  111  udp
     systat                                   11  tcp
     systat                                   11  udp
     telnet                                   23  tcp
     telnet                                   23  udp
     telnet-ssl                              992  tcp
     telnet-x                              32477  spx
     tftp                                     69  tcp
     tftp                                     69  udp
                                                                  More...
Parameters for options 1 and 4 or command
===>
F3=Exit   F4=Prompt   F5=Refresh   F6=Print list   F9=Retrieve   F12=Cancel
F17=Top   F18=Bottom
```

Figure 354. WRKSRVTBLE display

Then, you can add two Telnet entries to this table by typing option 1.

Note

You must have *IOSYSCFG special authority to add or remove entries in this list.

A screen appears as shown in Figure 355 on page 252.

```
                    Add Service Table Entry (ADDSRVTBLE)

Type choices, press Enter.

Service  . . . . . . . . . . . . > ' '
Port . . . . . . . . . . . . . .                1-65535
Protocol . . . . . . . . . . . . > ' '
Text 'description' . . . . . . .   *BLANK



                                                                        Bottom
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display        F24=More keys
```

Figure 355. ADDSRVTBLE display

You can specify the following parameters:

- **Service**: Enter `telnet`.
- **Port**: Enter the port number (different from the well-known port 23).
- **Protocol**: Specify the name of the protocol that the service uses. You should enter `TCP` here.

Note that when you add a port for Telnet so that there are duplicate port numbers, the Telnet server uses the first entry.

5.3.3 Starting the Telnet server

You can now restart the Telnet server on the AS/400 system by typing the following command:

```
STRTCPSVR SERVER(*TELNET)
```

The Telnet server can now listen on the port you specified.

Chapter 6. Using FTP on the AS/400 system

File Transfer Protocol (FTP) enables you to send and receive files across a TCP/IP network to and from different servers and clients with different operating systems such as OS/400, OS/390, OS/2, Windows 9x/NT, AIX and many more.

FTP on the AS/400 system consists of two parts: the client and the server function. The client initiates FTP subcommands that are sent to the FTP server. The results of these subcommands are then displayed. On the AS/400 system, you can enter the subcommands interactively (FTP is designed to be an interactive protocol and interface), or write a simple script for an unattended batch mode operation. In this case, the commands are read from a file and the responses to the subcommands can be written to a file.

To transfer the files between the client and the server, two connections are used. The *control connection* is used to send the subcommands from the client to the server and back again. The second connection is called the *data connection* and is used for transferring lists of file names and the actual file data.

Both the client and the server have a data transfer function that interfaces to the AS/400 file system. These functions read and write to the local file system and pass the file data to the data connection. The AS/400 FTP client and server can use most of the different file systems in the Integrated File System (IFS). This basically opens up the entire system to FTP, except non-file type objects in QSYS.LIB such as programs or user profiles. See 6.2, “Access to the Integrated File System (IFS)” on page 254, for more information.

Normally, to transfer files, you need a user ID on both systems or a special configuration set up by the system administrator. A good way around this restriction is the anonymous FTP. It essentially allows anyone in the world to have access to a certain area of disk space in a non-threatening way. See 6.5, “Anonymous FTP support” on page 261, for more information.

A typical file transfer works similar to the process explained here:

1. The client starts an FTP session.
2. The client requests a file transfer by typing in FTP subcommands.
3. The user interface function of the client reads this request and passes it to the client protocol interpreter.
4. The client protocol interpreter determines what the client requested and translates this into the appropriate FTP server subcommands.
5. The server protocol interpreter receives the subcommands from the control connection and processes it.
6. The results of each server subcommand are transmitted back to the client in the form of an FTP server reply.

6.1 FTP limitations and abilities

FTP transfers Data as a Datastream, not on a record basis. Because the AS/400 system stores many files in EBCDIC and PCs, while UNIX stores files in ASCII format, translation may be needed. There is no support for translating special numerical formats like packed decimal or zoned decimal. Therefore, you need to

convert those fields to alphanumeric characters before you can send or receive the file to a non-AS/400 system.

In a QDLS file system, FTP supports only object types of *FILE, so you cannot directly FTP objects like user profiles or programs to another system. But, you can save them to a save file and FTP them to any other computer system. Once back on an AS/400 system, you can restore the objects from that save file without any special additional tasks.

As you can see, FTP is not the best TCP/IP application available, but it can help you get your work done when you need to transfer files between an AS/400 system and any other computer capable of running FTP.

6.2 Access to the Integrated File System (IFS)

The introduction of the IFS brought several different file systems to the AS/400 system. These different file systems all have different naming schemes and structures. In the following sections, we discuss the different file systems and offer short hints on how they can be used.

Root (/) file system
The root file system supports a directory structure and commands that access information in stream files. It is similar to the QDLS file system but has the ability to use longer file names (up to 255 characters) and has removed some constraints that the QDLS had. For the FTP server, the access to the root file system means mainly to get access to the stream files.

QSYS.LIB file system
This file system (normally known as the AS/400 database) allows you to have direct access from FTP clients to the database of the AS/400 system. This means that you can access physical files (PF), logical files (LF), source physical files (SRCPF), and save files (SAVF). Note that you may need to restructure some physical files due to the fact that only alphanumeric fields are allowed with FTP (that is, no packed numeric data).

AS/400 FTP does not allow you to access objects other than the ones listed previously. For example, you cannot FTP a spooled file or an output queue object. However, you can save those objects in a save file and then FTP them to the remote system. Hopefully, that remote system is another AS/400 system that knows what to do with a save file.

QDLS file system
With QDLS, you have access to “virtual hard disks”. Here, you can store and archive the PC files you may need to distribute to your PCs. The QDLS is also used with OfficeVision (OV/400). OV/400 is the electronic mailing, archiving, and distribution system on the AS/400 system. Access to QDLS means that you also have access to the mail and the documents you archived on the AS/400 system.

QOPT file system
With the support of the QOPT file system, you have access to the optical disks that you can attach to the AS/400 system. Here, the AS/400 system supports the CD-ROMs and CD-WORMs.

QOpenSys file system

The QOpenSys file system opens up the AS/400 system for the UNIX world, because this is the file system that is usually used on UNIX machines. This allows us to store files in a native UNIX-like format on the AS/400 system. Note that this allows the AS/400 system to act as an FTP server for the UNIX world. You do not need to convert the files from one format to the other only to allow UNIX clients to transfer files to and from an FTP server.

QLANSrv file system

The QLANSrv file system was introduced to allow the Integrated PC Server (formerly known as File Server Input Output Processor (FSIOP)), which is a "built-in PC" within the AS/400 system that offers fast access to AS/400 disks. This gives you the fastest access available on the AS/400 system to PC-like files. It is in many ways similar to the QDLS file system, but it is many times faster when you access QLANSrv through the Integrated PC Server.

Note: You need additional software (the LAN Server/400) to support this file system.

Naming formats on the AS/400 system

File names must be specified in particular formats. These formats vary depending on the file system in which the file resides. In this book, we cannot possibly list all of the different file formats that you may encounter, but we can make sure that you understand the AS/400 system format.

Two naming formats are supported by the AS/400 FTP. They can be categorized as:

- **NAMEFMT 0**: This is the *traditional* (before we knew better) way of naming the physical files, logical files, and source file members that FTP addresses on the AS/400 system. This is a naming format only for the *QSYS.LIB file system* database files.

  The syntax is `libname/filename.membername` that you can read as "library name slash file name dot (or period) member name". This naming format worked fine for the kinds of database objects that FTP originally accessed, but does not work for some of the new objects that we find in the QDLS or QOPT file systems.

  Since this name format was first, it is called `NAMEFMT 0`, and is the default whenever you use either the FTP client or server on the AS/400 system.

- **NAMEFMT 1**: This is the naming format for the *IFS*. This format must be used to work with the IFS file systems such as QDLS or QOPT and save files found in the traditional file system QSYS.LIB.

  This naming format was created when AS/400 FTP was improved to access (in addition to the database file system) QDLS and QOPT. You also can now FTP any AS/400 object that can find its way into a save file. Save files, to be clear, can be sent or received using either the old NAMEFMT 0 or the new NAMEFMT 1.

  A path name (also called a pathname on some systems) tells FTP how to locate an object. The path name is expressed as a sequence of directory names followed by the name of the object. Individual directories and the object name are separated by a slash (/) character. Here is an example:

directory1/directory2/file that is read as directory one slash directory two slash file. Some more real-to-life examples can sometimes help:

```
/QDLS/KRISP/MONDO/BIGFILE.TXT
/QSYS.LIB/ITSOIC400.LIB/QRPGLESRC.FILE/FTPLOGON.MBR
/QSYS.LIB/SAVLIB.LIB/SAVEJUN.SAVF
/QOpenSys/Direct/file12 [1]
/newclass/l05d040.htm
```

Note

The QOpenSys path names are case sensitive. Other path names can usually be written as upper, lower, or mixed case.

Note: Because NAMEFMT 1 does everything that the old NAMEFMT 0 does, plus a whole lot more, we strongly advise that you only use the new IFS naming format. Unfortunately, the old naming format is the default. Therefore, you must first issue the FTP subcommand NAMEFMT 1 when using the AS/400 FTP client. Conversely, when you are using the AS/400 FTP server, issue the SITE NAMEFMT 1 (or, with some FTP clients that do not support SITE subcommand, use QUOTE SITE NAMEFMT 1) subcommand from the non-AS/400 client. If you are using FTP between two AS/400 systems, you only need to enter NAMEFMT 1 on the client side since the naming format request is automatically passed to the server by your AS/400 FTP client. In V4R4, you may change the FTP attributes to default to NAMEFMT 1. Refer to 6.3.3, “AS/400 FTP server configuration” on page 257, for details.

6.3 AS/400 FTP server

The AS/400 system can be used as an FTP server and client. This section explains the AS/400 server. The distinction between the FTP server and FTP client is from the viewpoint of where the commands are entered, not where the data resides.

6.3.1 Why use the AS/400 FTP server

Here are some arguments that differentiate the AS/400 FTP server implementation from other FTP servers on the market:

- Support of several different file systems on *one* system (IFS)
- Built in high security standards
- AS/400 programs and CL commands can be called through an RCMD subcommand
- Transfer of folders and documents in the QDLS file system
- Sending or receiving physical files, source files, logical files, and save files (SAVF)
- Creating and deleting libraries, files, and members using the AS/400 FTP server subcommands
- Creating and deleting folders using the AS/400 FTP server subcommands
- Distribute most AS/400 objects from one AS/400 system to another via save files

- High availability of the FTP server and data through mirroring, RAID-5, and other AS/400 system availability options
- Exit programs for different FTP events

6.3.2 FTP server checklist

There are several points you should consider as you implement FTP on your AS/400 system. Some of the points apply only if you are going to connect your system to the Internet. You may use FTP on your intranet without being connected to the Internet. Some points to consider are:

- Do you have enough DASD space?
- Is your AS/400 system secured against hackers or crackers? Every server running on your AS/400 system increases the chance of system penetration.
- Is your AS/400 system on the Internet separated from your network?
- Does your company have a lot of public available information to distribute?
- Are your customers interested in having access to your FTP server?
- How can you make it attractive to access your FTP server?
- Is it of value for your company to know who contacted you?
- Is your data processing staff educated in TCP/IP, Internet, WWW, and so on?
- How do you load the data on your FTP server?
- Do your customers already have access to the Internet?
- How many users are going to use FTP service?
- Are you on a LAN or should you buy a LAN File Server instead?
- Did your customers already request access through the Internet?
- Do you have customers all over the world?
- Do you need to be online 24 hours a day?
- Do you know what it costs to be online?
- Can your afford *not* to be on the Internet?

6.3.3 AS/400 FTP server configuration

Before you can use the FTP Server on the AS/400 system, TCP/IP must be configured. To configure the AS/400 FTP Server, use the Change FTP Attributes (CHGFTPA) command. Figure 356 on page 258 shows the display for Change FTP Attributes. This screen was captured on a V4R4 AS/400 system. The portions in bold are V4R4-only parameters.

```
                    Change FTP Attributes (CHGFTPA)

Type choices, press Enter.

Autostart servers  . . . . . .   *YES          *YES, *NO, *SAME
Number of initial servers  . . .   3            1-20, *SAME, *DFT
Inactivity timeout . . . . . .   300            0-2147483647, *SAME, *DFT
Coded character set identifier     00819        1-65533, *SAME, *DFT
Server mapping tables:
  Outgoing EBCDIC/ASCII table  .   *CCSID       Name, *SAME, *CCSID, *DFT
    Library  . . . . . .                       Name, *LIBL, *CURLIB

  Incoming ASCII/EBCDIC table  .   *CCSID       Name, *SAME, *CCSID, *DFT
    Library  . . . . . .                       Name, *LIBL, *CURLIB
Initial name format  . . . . .   *LIB          *LIB, *SAME, *PATH
Initial directory  . . . . . .   *CURLIB       *CURLIB, *SAME, *HOMEDIR
Initial list format  . . . . .   *DFT          *DFT, *SAME, *UNIX
New file CCSID . . . . . .       *CALC         1-65533, *SAME, *CALC...


                                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 356. Change FTP Attributes

The parameters shown in the screen in Figure 356 are explained here:

- **Autostart Servers**: Use *YES to automatically start the FTP servers on the AS/400 system whenever TCP/IP starts.
- **Number of Initial servers**: Depending on the FTP traffic you expect, set the value for initial servers. The default 3 will be enough for most cases.
- **Inactivity timeout**: You can specify after how many seconds of inactivity the server will drop the connection. For most cases, 300 seconds (equal to 5 minutes) will fit.
- **Coded character set identifier**: The default value (00819) causes all incoming ASCII characters to be translated to EBCDIC 500 code page. You can also specify your own translation table. See the online help for more information.
- **Initial name format (V4R4)**: The default value (*LIB) causes all the servers to process file names in the format library/filename.member (Namefmt 0). If you change the value to *path, the IFS naming is used (Namefmt 1).
- **Initial directory (V4R4)**: The default value (*CURLIB) sets the current directory in the FTP session to the current library specified in their user profile when the user logs into FTP. If you change the value to *HOMEDIR, the current directory is set to the value specified in the homedir parameter of their user profile.
- **New file CCSID (V4R4)**: This parameter determines the CCSID of any files that are created using FTP.

To manually start the FTP server, use the Start TCP Server (`STRTCPSVR *FTP`) command. To end all FTP servers, use the End TCP/IP Server (`ENDTCPSVR *FTP`) command. You cannot end one specific FTP server using ENDTCPSVR *FTP. To

do this, use the End Job (ENDJOB) command for the FTP server job that you want to end.

6.4 AS/400 FTP client

With the FTP client, you can connect to any FTP server available to you. The FTP client allows you to use it interactively or in *batch* mode, where the commands for the FTP session are taken from an input file. An example program to easily transfer files with FTP is explained in 6.7, “Batch FTP” on page 266.

6.4.1 FTP passive transfer

FTP passive transfer is needed in some cases to connect to a FTP servers if you are behind a firewall. Since OS/400 V4R2, the AS/400 FTP client, by default, tries to connect to a FTP server in passive mode. If the FTP server does not support passive mode, the AS/400 FTP client tries to connect without passive transfer enabled as shown in Figure 357. To change the default behavior back to port rather than passive, a data area must be created. When the FTP client starts, it checks for the existence of the data area. If the data area is found, port mode is used. If the data area is not found, passive mode is attempted. To enable this switch, a PTF is needed for V4R2 and V4R3. The base code in V4R4 provides the switch function.

Note

A PTF is needed to implement the switch function in V4R2 and V4R3. Select the correct PTF for 5769-TC1 from the following list:

- V4R2 SF51972
- V4R3 SF52486

```
                          File Transfer Protocol

Previous FTP subcommands and messages:
  Connecting to remote host 10.8.62.168 using port 21.
  220 IIFTP Personal Server v1.1.0 Ready at Wed Nov 11 1998 - 17:24:17
> anonymous
  331 Anonymous Ok - Send E-Mail Address as Password
  230 Password Accepted - Anonymous User Logged In!
  MSDOS Type: L8 Version: IIFTP Personal Server v1.1.0
> ls
 500 PASV Command Unavailable
  Unable to setup for an active data connection to the server, reason code 1.
  Use of PASV disabled for this server session.
  200 Port Command Successful 10.8.69.232:1056
  150 Opening Data Connection to 10.8.69.232:1056 for /bin/nls (146 bytes)
```

Figure 357. Passive mode disabled

To set the default to SENDPASV OFF (passive mode is not used), create a data area QTMFTPPASV in library QUSRSYS using the following command:

```
CRTDTAARA DTAARA(QUSRSYS/QTMFTPPASV) TYPE(*LGL) AUT(*USE)
```

To revert to SENDPASV ON (try a passive connection first) as the default setting for the FTP client, delete the data area using the following command:

```
DLTDTAARA DTAARA(QUSRSYS/QTMFTPPASV)
```

6.4.2 Useful AS/400 FTP client subcommands

A selection of useful FTP subcommands is available for you in Table 10.

Table 10. Useful FTP subcommands

Subcommand	Description
AScii	Switch to ASCII mode. This is the default mode and is used for transferring character text. The FTP client translates the incoming ASCII to EBCDIC and outgoing EBCDIC to ASCII. The remote FTP server thinks it is talking to a native ASCII client.
Binary	Switch to binary mode (or sometimes called *image transfer*). This is used for transferring binary files such as ZIP files, SAVF, or files with packed decimal data.
CD	Change the current directory on the *remote* computer.
LCd	Changes the current directory on your *local* computer.
CDUp	Change to the parent directory on the *remote* computer.
Dir	List the files in the *remote* computer (same as in PC).
LS	Same as DIR but lists only the file names (same as in UNIX).
Get	Copies a file from the *remote* computer to the *local* computer.
MGet	Copies multiple files from the *remote* computer to the *local* computer.
PUt	Copies a file from the *local* computer to the *remote* computer.
MPUt	Copies multiple files from the *local* computer to the *remote* computer.
LPWd	Shows the present working directory (pwd) on your *local* computer.
PWd	Shows the present working directory (pwd) on the *remote* computer.
DEBug	To control the display of server subcommands sent to the server.
	Note: Notice all of the highlighted *local* and *remote* words in the text above and below this point. The reason they are highlighted is that it is sometimes difficult to learn just what each FTP subcommand sends to the remote system and what it gets back. By using the DEBUG subcommand, you can see the English-based communication traffic between the client and the server. This, for example, can help you remember if `CD` changes the directory on the remote or the local system. With `DEBUG 100`, you set an FTP client trace on. This results in trace data being dumped and formatted in a file named QPSRVTRC.
LOCstat	Shows the local status information.
NAmefmt	Changes the naming format from the default of 0 (which is the traditional way of naming the database objects in AS/400 libraries to 1, the new IFS naming format.
NOop	To find out if the remote FTP server is responding.
SYSCMD	To run a local AS/400 command. This is very useful.

Subcommand	Description
SITE	To send information to the server that it needs for services. Generally, you do not use this command from the AS/400 client that much.
QUOTE	To send a server subcommand directly to the server. It is sometimes useful to issue the command `QUOTE HELP PUT`, for example, to get help information for the PUT subcommand from the remote server.
SENDPAsv	Toggle the PASV status. See 6.4.1, “FTP passive transfer” on page 259, for details.
HELP	To get online help from the local AS/400 host.
?	To get general help from the local AS/400 host.

6.5 Anonymous FTP support

As the Internet started to grow to thousands of servers, there was an urgent need to have an easy access to this huge amount of data. Because you cannot administer to the millions of potential users in all parts of the world, you have to find an easy and simple way to allow access to this “public” data. One of these is the *anonymous FTP* method.

Most computer systems on the Internet offer this support. Normally, if you log on to a computer, you are asked for a user ID and a password. On the Internet, you probably log on to a hundred different computers distributed all over the world. For your daily work, you now need a common user ID. This common user ID is “anonymous”.

When the server asks you for a user ID, type `anonymous`. Custom and etiquette dictate that you use your e-mail address (for example, `myname@ibm.net`) as your password. Some servers may accept any character string such as `asdfasdf` as your e-mail address, but do not use that because some servers may insist on a formal correct e-mail address. Others may prohibit further access to your IP-address. If the server responds with the message `not a valid password`, this usually means that the password is scanned for the @ character. You have to respond with a formal valid e-mail address.

The anonymous FTP servers usually contain software, documents, graphic images, or other kinds of information. All of this information is considered to be publicly accessible and can be read by everyone who wants to read it. Note that this may be subject to change. The person who “owns” the information and the system can, of course, shut down the machine at any time and refuse further public access. You cannot do anything about that.

On the AS/400 system, the anonymous FTP is enabled by using an FTP server logon exit program and an FTP server request validation exit program. When an anonymous FTP is enabled on the AS/400 system, the FTP server requests an e-mail address instead of a password in an anonymous logon.

6.6 FTP exit programs

An *exit point* is a specific point in the TCP/IP application where control may be passed to an exit program. The FTP exit programs are used to control the use of the FTP server and FTP client. These exit programs offer additional security and transaction logging on an AS/400 FTP server.

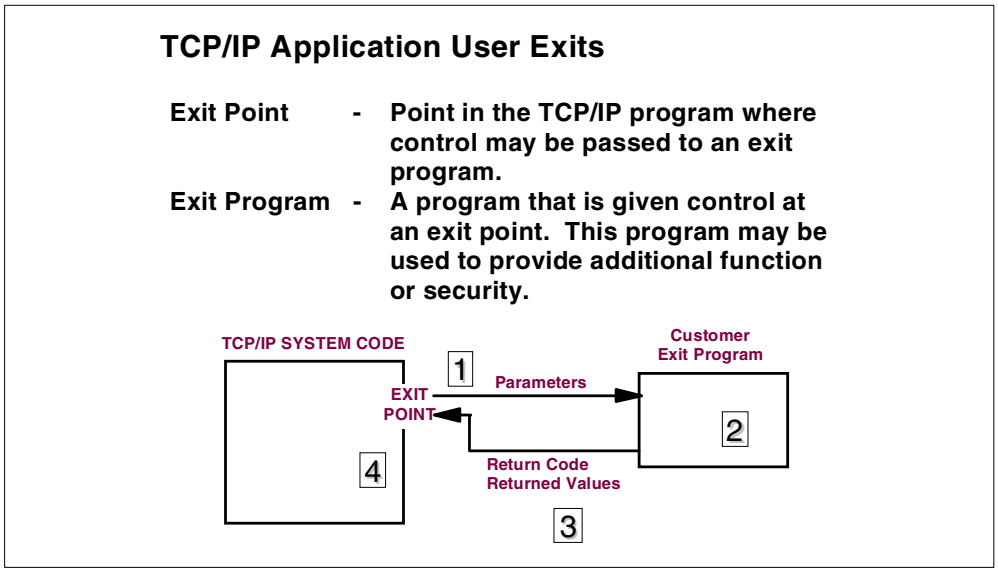


Figure 358. TCP/IP exit point processing

The processing flow shown in Figure 358 is shown here:

1. The TCP/IP application passes request parameters to the exit program.
2. The exit program processes request parameters.
3. The exit program returns information to the TCP/IP application.
4. The TCP/IP application performs an operation based on the exit program response.

Three different exit points are provided for FTP (Table 11). The first one is used to validate requests processed by the FTP client program, the second is used to validate requests processed by the FTP server program, and the third controls the logon requests to the FTP server program.

Table 11. FTP exit points

FTP application	Exit point	Exit point format
FTP client	QIBM_QTMF_CLIENT_REQ	VLRQ0100
FTP server	QIBM_QTMF_SERVER_REQ	VLRQ0100
FTP server	QIBM_QTMF_SVR_LOGON	TCPL0100
		TCPL0200 (V4R4)

The same format is used for both the FTP client and FTP server for request validation. This enables the use of one program for both client and server request validation.

Note: To enable anonymous FTP, you must define exit programs for *both* FTP server exit points.

The FTP server logon exit program permits or denies a logon to an FTP server based on one or more of the following options:

- User ID
- Password
- Remote IP address

FTP request validation exit programs (FTP client or FTP server) permit or deny a specific FTP operation based on one or more of the following options:

- User profile
- Remote IP address
- Directory, library, files (path names)
- CL commands

Exit programs are defined for exit points using the OS/400 registration facility. You can use Work with Registration Information (WRKREGINF) command (Figure 359) to display a list of exit points or simply use Add Exit Program (ADDEXITPGM) command.

```
                    Work with Registration Information

Type options, press Enter.
  5=Display exit point   8=Work with exit programs

                           Exit
     Exit                  Point
Opt  Point                 Format    Registered  Text
     QIBM_QTA_STOR_EX400   EX400200     *YES
     QIBM_QTA_STOR_EX400   EX400300     *YES
     QIBM_QTA_TAPE_TMS     TMS00200     *YES
     QIBM_QTF_TRANSFER     TRAN0100     *YES     Original File Transfer Functi
     QIBM_QTG_DEVINIT      INIT0100     *YES     Telnet Device Initialization
     QIBM_QTG_DEVTERM      TERM0100     *YES     Telnet Device Termination
     QIBM_QIMF_CLIENT_REQ  VLRQ0100     *YES     FTP Client Request Validation
     QIBM_QIMF_SERVER_REQ  VLRQ0100     *YES     FTP Server Request Validation
     QIBM_QIMF_SVR_LOGON   TCPL0100     *YES     FTP Server Logon
     QIBM_QIMF_SVR_LOGON   TCPL0200     *YES     FTP Server Logon    (V4R4)
     QIBM_QIMT_WSG         QAPP0100     *YES     WSG Server Sign-On Validation
                                                                      More...
Command
===>
F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
```

Figure 359. Work with Registration Information

6.6.1 FTP server logon exit program

The FTP server logon exit program allows you to enforce your own rules for handling logons to an FTP server. The exit program receives as parameters:

- Application identifier
- User identifier
- Authentication string (password or e-mail address from FTP logon)
- Remote IP address
- Length fields for some of the previously mentioned parameters

The exit program returns a parameter whether the logon is rejected, accepted, or if the logon operation should be continued. When the logon is accepted, a user profile is returned and a new current library can be returned as a parameter to override the one specified in the user profile. When the logon is continued, the program can return a new user profile, a new password, and a new current library as parameters to override either entered values or the one specified in the user profile.

Note

The password on the output parameter *must* match the password specified in the user profile for the logon to succeed, but we strongly recommend that passwords *never* be coded in an exit program.

Overriding the initial current library of a user (set in the user profile) is called *direct routing*. Direct routing support allows you, depending on the clients network address, to route the client directly to a specific library immediately following the logon on the AS/400 FTP server. This allows you, for example, to route sales representatives to their own library. In such a private library, you can keep the files that are unique for a specific sales representative.

Starting in V4R4, you can use exit point format TCPL0200 to specify an initial directory in the IFS name format rather than library format. This must be a direct path to a directory in the IFS starting at the root. For more information, refer to the AS/400 Information Center at: `http://www.as400.ibm.com/infocenter`

Using the search function, search for `FTP exit point`.

The exit program can control the access to the server based on the requester's address or user ID. The exit program can also be used to force your anonymous FTP user to give a valid-looking e-mail address as their password (one that contains a “@” character). The exit program cannot give any informative messages to the clients, so they see only that a logon is either accepted or rejected.

Specifying an FTP server logon exit program is one-half of enabling the anonymous FTP. You should create a separate user profile for this. We strongly suggest that this user profile has a password of *NONE. In the exit program, you can also force some of your local users to use this user profile for FTP. All server logon requests can, and all anonymous server logon requests *should*, be logged to see who accesses your AS/400 system. Because the exit program receives a valid user password as a parameter, take care in the information that you log. The importance of logging is even greater if you are connected to the Internet. You might be compelled later on to maintain a “black list” of IP addresses whose owners have tried to do something harmful and block their entrance to your system. Although most of the general Internet users can be considered to be quite harmless, there are the fringe groups that just may have something against your area of business.

6.6.2 FTP client and server request validation exit programs

Because both client and server request validation exit programs use the same exit point format (parameters), it is easier to describe their functions in the same

chapter. Exit point programs can be one program, but they can also be two separate programs.

The FTP request validation exit program gives you control over whether an operation (that is, an FTP subcommand) is performed. The decisions made by the exit programs are in addition to any validation performed by the application program. When installed, the exit program is called each time one of the following requests are processed:

- Session initialization/login (server exit program only during session initialization)
- Directory/library creation
- Directory/library deletion
- Setting current directory
- Listing file names
- File deletion
- Sending a file
- Receiving a file
- Renaming a file
- Executing a CL command on the FTP server

The exit program receives as parameters:

- Application identifier (server or client request)
- Operation identifier
- User profile
- Remote IP address
- Some operation specific data (path name, CL command, and so on)
- Length fields for some of the previously mentioned parameters

It returns as a parameter, whether the operation is rejected or accepted. An operation can be rejected completely for the remainder of the session, rejected this time, allowed this time, or allowed unconditionally for the remainder of the session.

Your user exit program can be based upon the user's IP address and then provide different access to AS/400 data. For example, your employees may have unlimited access, but your customers are allowed to see (and “get”) only the product information that is available to the public. You can also limit FTP users from using certain CL commands, but allow other commands to be used.

The other half of enabling the anonymous FTP is specifying an FTP server exit program. You should create a good protection scheme against your FTP clients. Although the users may be your customers, you must consider them to be threats. Remember that one of the common threats to a computer system is users who do not really know what they are doing but have too much access authority. A good idea is to limit their access to downloading files from selected libraries or directories. The need to upload files must be considered carefully. If you allow it, limit uploading to selected libraries or directories that are not the same as download directories. Limit the use of CL commands and deleting objects as well as limiting renaming and creating objects. All in all, limit everything that you do not explicitly allow. Remember, a too tight security scheme is always better than a bad and leaking one. You may also want to use symbolic links in your “public” directory and have the files actually reside in a more secure directory or folder.

The question of using a client exit program is quite different from the server program. Does the task of protecting a server belong to the administrators of the server system? That is correct, but if you are connected to the Internet, do you want your employees to search all day for data in the vast selection of FTP servers? If you have business needs for some data that can be found in the Internet, allow only selected users to do the retrieval. Otherwise, you may have to buy more storage to accommodate all of the interesting information your users have found on the Internet.

The functions that are executed during a validation of a request are not restricted. A lot of nice things, and lot of not so nice things, can be done. For example, your exit program can save a library to a save file whenever this save file is requested by a client. Another example may be to refresh the contents of a database file immediately prior to the client's attempt to download it. Although these functions are nice (and perhaps cool), they must be considered to be “Trojan horses”. A Trojan horse is a function that is built into a program and triggered by some event usually to do something harmful. Take care when building these functions to the exit programs. It is also good to regularly check that the exit programs used are really the correct ones.

6.7 Batch FTP

FTPBATCH is a small application for transferring files in an automated manner using FTP. It consists of one command, one CL program, and two RPG programs. The application can be easily integrated to existing applications. The command FTPBATCH (Figure 360) is the interface to other applications. Because it returns a parameter value, it must be used from a CL program or REXX procedure. The command processing program for FTPBATCH is STRFTP1C.

```
/*****************************************************************/
/*                                                               */
/*                          ** NOTE **                           */
/*                                                               */
/* This material contains programming source code for your       */
/* consideration. These examples have not been thoroughly tested */
/* under all conditions. IBM, therefore, cannot guarantee or imply */
/* reliability, serviceability, performance or function of these */
/* programs.  All programs contained herein are provided "AS IS". */
/* THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A   */
/* PARTICULAR PURPOSE ARE EXPRESSLY DISCLAIMED.                  */
/*                                                               */
/*****************************************************************/
 FTPBATCH:   CMD        PROMPT('START A FTP BATCH TRANSFER')
             PARM       KWD(RMTSYS) TYPE(*CHAR) LEN(50) MIN(1) +
                          PROMPT('Remote system for FTP transfer')
             PARM       KWD(LCLFIL) TYPE(*CHAR) LEN(50) MIN(1) +
                          PROMPT('Local file''s name')
             PARM       KWD(RMTFIL) TYPE(*CHAR) LEN(50) MIN(1) +
                          PROMPT('Remote file''s name')
             PARM       KWD(FUNCTION) TYPE(*CHAR) LEN(1) RSTD(*YES) +
                          VALUES(P G) SPCVAL((p P) (g G)) MIN(1) +
                          PROMPT('Function (Put/Get)')
             PARM       KWD(USR) TYPE(*CHAR) LEN(10) MIN(1) +
                          PROMPT('User profile in remote system')
             PARM       KWD(PWD) TYPE(*CHAR) LEN(10) MIN(1) +
                          PROMPT('Password to remote system')
```

Figure 360. FTPBATCH command

The main program for the application is STRFTP1C. It receives as parameters:

- IP address or host name of remote system
- Local name for the file to be transferred
- Remote name for the file to be transferred
- Transfer function to be executed (Put or Get)
- User ID for the remote system
- Password for the remote system
- Three FTP commands to be executed before actual transfer

STRFTP1C returns as a parameter, which is an indicator of the success of the transfer.

Notes: See Figure 361 through Figure 363 on page 270, and the following list:

1. STRFTP1C calls program BLDFTP1R to create a file containing all of the needed FTP commands.
2. STRFTP1C calls program CHKFTP1R to check the result of the transfer.
3. All work files are created in the library QTEMP.
4. FTP is run using system defaults. If you need to use special conversion tables for the transfer, modify the parameters of STRTCPFTP command in the program.

```
/*********************************************************************/
/*                                                                   */
/*                     PROGRAM FUNCTION                              */
/*                                                                   */
/* This is the CPP for command FTPBATCH.                             */
/*                                                                   */
/* It performs a batch FTP transfer of one file with a TCP/IP        */
/* network connected FTP server.                                     */
/*                                                                   */
/* Program parameters:                                               */
/*                                                                   */
/*         &TGTSYS    (in )   -   Target system                      */
/*         &LCLFIL    (in )   -   Local file name                    */
/*         &RMTFIL    (in )   -   Remote file name                   */
/*         &FUNCTION  (in )   -   Function to be executed            */
/*                           values: 'P' = Put                       */
/*                                   'G' = Get                       */
/*         &USR       (in )   -   User ID in remote system           */
/*         &PWD       (in )   -   Password in remote system          */
/*         &CMD1      (in )   -   Optional FTP subcommand            */
/*                                (Executed before transfer)         */
/*         &CMD2      (in )   -   Optional FTP subcommand            */
/*                                (Executed before transfer)         */
/*         &CMD3      (in )   -   Optional FTP subcommand            */
/*                                (Executed before transfer)         */
/*         &SUCCES    (out)   -   Success indicator                  */
/*                           values: '0' = Transfer failed           */
/*                                   '1' = Transfer successful       */
/* All work files are created into library QTEMP.                    */
/*                                                                   */
/* Programs called:                                                  */
/*                                                                   */
/*       1 BLDFTP1R (ILE RPG) -   Creates the FTP script             */
/*                                for transfer                       */
/*       2 CHKFTP1R (ILE RPG) -   Checks the success of transfer     */
/*                                by reading the FTP transfer log    */
/*                                                                   */
/*********************************************************************/
/*                                                                   */
/*                          ** NOTE **                               */
/*                                                                   */
/* This material contains programming source code for your           */
/* consideration. These examples have not been thoroughly tested     */
/* under all conditions. IBM, therefore, cannot guarantee or imply   */
/* reliability, serviceability, performance or function of these     */
/* programs.  All programs contained herein are provided "AS IS".    */
/* THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A       */
/* PARTICULAR PURPOSE ARE EXPRESSLY DISCLAIMED.                      */
/*                                                                   */
/*********************************************************************/
PGM        PARM(&TGTSYS   +
                           &LCLFIL   +
                           &RMTFIL   +
                           &FUNCTION +
                           &USR      +
                           &PWD      +
                           &CMD1     +
                           &CMD2     +
                           &CMD3     +
                           &SUCCES   )
```

Figure 361. CL program STRFTP1C (Part 1 of 3)

```
/*****************************************************************/
/* Parameters                                                    */
/*****************************************************************/
            DCL        VAR(&TGTSYS  ) TYPE(*CHAR) LEN(50)
            DCL        VAR(&LCLFIL  ) TYPE(*CHAR) LEN(50)
            DCL        VAR(&RMTFIL  ) TYPE(*CHAR) LEN(50)
            DCL        VAR(&FUNCTION) TYPE(*CHAR) LEN(1 )
            DCL        VAR(&USR     ) TYPE(*CHAR) LEN(10)
            DCL        VAR(&PWD     ) TYPE(*CHAR) LEN(10)
            DCL        VAR(&CMD1    ) TYPE(*CHAR) LEN(20)
            DCL        VAR(&CMD2    ) TYPE(*CHAR) LEN(20)
            DCL        VAR(&CMD3    ) TYPE(*CHAR) LEN(20)
            DCL        VAR(&SUCCES  ) TYPE(*CHAR) LEN(1)
/*****************************************************************/
/* Parameters                                                    */
/*****************************************************************/
            DCL        VAR(&TGTSYS  ) TYPE(*CHAR) LEN(50)
            DCL        VAR(&LCLFIL  ) TYPE(*CHAR) LEN(50)
            DCL        VAR(&RMTFIL  ) TYPE(*CHAR) LEN(50)
            DCL        VAR(&FUNCTION) TYPE(*CHAR) LEN(1 )
            DCL        VAR(&USR     ) TYPE(*CHAR) LEN(10)
            DCL        VAR(&PWD     ) TYPE(*CHAR) LEN(10)
            DCL        VAR(&CMD1    ) TYPE(*CHAR) LEN(20)
            DCL        VAR(&CMD2    ) TYPE(*CHAR) LEN(20)
            DCL        VAR(&CMD3    ) TYPE(*CHAR) LEN(20)
            DCL        VAR(&SUCCES  ) TYPE(*CHAR) LEN(1)

/*****************************************************************/
/* Local variables                                               */
/*****************************************************************/
            DCL        VAR(&OKCODE  ) TYPE(*CHAR) LEN(3)
            DCL        VAR(&ON      ) TYPE(*CHAR) LEN(1) VALUE('1')
            DCL        VAR(&OFF     ) TYPE(*CHAR) LEN(1) VALUE('0')

/*****************************************************************/
/* Delete/Create the FTP script file                             */
/*****************************************************************/
          3 DLTF       FILE(QTEMP/FTPCMD)
            MONMSG     MSGID(CPF0000)
            CRTSRCPF   FILE(QTEMP/FTPCMD) MBR(FTP)

/*****************************************************************/
/* Build the FTP script                                          */
/*****************************************************************/
          1 OVRDBF     FILE(FTPCMD) TOFILE(QTEMP/FTPCMD) MBR(FTP)
            CALL       PGM(BLDFTP1R) PARM(&USR &PWD &FUNCTION +
                                  &LCLFIL &RMTFIL &CMD1 &CMD2 &CMD3)

/*****************************************************************/
/* Delete/Create the FTP transfer log file                       */
/*****************************************************************/
          3 DLTF       FILE(QTEMP/FTPLOG)
            MONMSG     MSGID(CPF0000)
            CRTPF      FILE(QTEMP/FTPLOG) RCDLEN(132) MBR(FTPLOG)

/*****************************************************************/
/* Execute FTP transfer                                          */
/*****************************************************************/
            OVRDBF     FILE(INPUT ) TOFILE(QTEMP/FTPCMD) MBR(FTP)
            OVRDBF     FILE(OUTPUT) TOFILE(QTEMP/FTPLOG) MBR(FTPLOG)
          4 STRTCPFTP  RMTSYS(&TGTSYS)
            DLTOVR     FILE(*ALL)
```

Figure 362. CL program STRFTP1C (Part 2 of 3)

```
/*****************************************************************/
/* Check FTP transfer log file to find out if the transfer was   */
/* successful                                                    */
/*****************************************************************/
            CALL       PGM(CHKFTP1R) PARM(&OKCODE)
            IF         COND(&OKCODE *EQ 'YES') THEN(CHGVAR +
                         VAR(&SUCCES) VALUE(&ON))
            ELSE       CMD(CHGVAR VAR(&SUCCES) VALUE(&OFF))

            ENDPGM
```

Figure 363. CL program STRFTP1C (Part 3 of 3)

Program BLDFTP1R is called by STRFTP1C. It writes into a source physical file that the FTP command needed to execute the file transfer. It receives as parameters:

- User ID for the remote system
- Password for the remote system
- Transfer function to be executed (Put or Get)
- Local name for the file to be transferred
- Remote name for the file to be transferred
- Three FTP commands to be executed before actual transfer

Notes: See Figure 364 through Figure 367 on page 274 and the following notes:

1. File names need to contain a full path unless you use optional commands to change current directories to point into the correct directories/libraries.
2. Optional commands are not checked to contain valid FTP commands and are added to the file if they are non-blank.
3. The replace option is always used with the GET command. You can modify the application to make this optional.

```
F/TITLE CREATE FTP SCRIPT
 ********************************************************************
 *                      PROGRAM FUNCTION                            *
 *                                                                  *
 * This program creates the FTP script for FTPBATCH command.        *
 *                                                                  *
 ********************************************************************
 *                                                                  *
 *                          ** NOTE **                              *
 *                                                                  *
 * This material contains programming source code for your          *
 * consideration. These examples have not been thoroughly tested    *
 * under all conditions. IBM, therefore, cannot guarantee or imply  *
 * reliability, serviceability, performance or function of these    *
 * programs.  All programs contained herein are provided "AS IS".   *
 * THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A      *
 * PARTICULAR PURPOSE ARE EXPRESSLY DISCLAIMED.                     *
 *                                                                  *
 ********************************************************************
F/SPACE 3
 ********************************************************************
 *                                                                  *
 *                       INDICATOR USAGE                            *
 *                                                                  *
 *   IND.  DESCRIPTION                                              *
 *                                                                  *
 *    LR - CLOSE FILES ON EXIT                                      *
 *                                                                  *
 ********************************************************************
F/EJECT
 ********************************************************************
 *FILES USED BY THIS PROGRAM
 ********************************************************************
FFTPCMD    O    F   92        DISK                                   FTP script file
D/EJECT
 ********************************************************************
 * DATA STRUCTURES USED BY THIS PROGRAM
 ********************************************************************
D/SPACE 2
 *
 * Source file record format
 *
D CmdRec          DS
D  SRCSEQ                 1      6  2
D  SRCDAT                 7     12  0
D  SRCDTA                13     92
C/EJECT
 ********************************************************************
 * VARIABLE DEFINITIONS AND LISTS USED BY THIS PROGRAM              *
 ********************************************************************
```

Figure 364. ILE RPG program BLDFTP1R (Part 1 of 4)

```
     *
     * Define parameter list
     *
C      *ENTRY        PLIST
C                    PARM                    RMTUSR          10            Remote user ID
C                    PARM                    RMTPWD          10            Remote password
C                    PARM                    FUNCTION         1            Function to execute:
     *                                                  possible values:   P = PUT
     *                                                                     G = GET
C            1       PARM                    LCLFIL          50            Local file name
     *                                                                       (fully qualified)
C            1       PARM                    RMTFIL          50            Remote file name
     *                                                                       (fully qualified)
C            2       PARM                    CMD1            20            1st optional FTP cmd
C            2       PARM                    CMD2            20            2nd optional FTP cmd
C            2       PARM                    CMD3            20            3rd optional FTP cmd
C/EJECT
     ********************************************************************
     * The Main Program                                                 *
     ********************************************************************
     *
C                    EXSR      CF001
     *
C                    EVAL      *INLR = *ON
C                    RETURN
C*
C/EJECT
     ********************************************************************
     * S U B R O U T I N E S                                            *
     ********************************************************************
     * Main Processing                                                  *
     ********************************************************************
C      CF001         BEGSR
     *
C                    Z-ADD     *ZERO         SEQ              6 2
     *
C                    EXSR      CF002
C                    EXSR      CF003
C                    EXSR      CF004
C                    EXSR      CF005
     *
C                    ENDSR
C/EJECT
     ********************************************************************
     * Build and Write User ID and Password record                      *
     ********************************************************************
C      CF002         BEGSR
     *
C                    Z-ADD     *ZERO         SRCDAT
C                    ADD       1             SEQ
C                    Z-ADD     SEQ           SRCSEQ
C      RMTUSR        CAT(P)    RMTPWD:1      SRCDTA
C                    WRITE     FTPCMD        CmdRec
     *
C                    ENDSR
C/EJECT
```

Figure 365. ILE RPG program BLDFTP1R (Part 2 of 4)

```
 ****************************************************************
 * Write Optional FTP command records                           *
 ****************************************************************
C     CF003         BEGSR
 *
C     CMD1          IFNE      *BLANK
C                   Z-ADD     *ZERO         SRCDAT
C                   ADD       1             SEQ
C                   Z-ADD     SEQ           SRCSEQ
C                   MOVEL(P)  CMD1          SRCDTA
C                   WRITE     FTPCMD        CmdRec
C                   ENDIF
 *
C     CMD2          IFNE      *BLANK
C                   Z-ADD     *ZERO         SRCDAT
C                   ADD       1             SEQ
C                   Z-ADD     SEQ           SRCSEQ
C                   MOVEL(P)  CMD2          SRCDTA
C                   WRITE     FTPCMD        CmdRec
C                   ENDIF
 *
C     CMD3          IFNE      *BLANK
C                   Z-ADD     *ZERO         SRCDAT
C                   ADD       1             SEQ
C                   Z-ADD     SEQ           SRCSEQ
C                   MOVEL(P)  CMD3          SRCDTA
C                   WRITE     FTPCMD        CmdRec
C                   ENDIF
 *
C                   ENDSR
C/EJECT
 ****************************************************************
 * Build and Write PUT/GET Command Record                       *
 ****************************************************************
C     CF004         BEGSR
 *
C                   Z-ADD     *ZERO         SRCDAT
C                   ADD       1             SEQ
C                   Z-ADD     SEQ           SRCSEQ
 *
C                   SELECT
C     FUNCTION      WHENEQ    'P'                                 PUT function
C     'PUT'         CAT(P)    LCLFIL:1      SRCDTA
C     SRCDTA        CAT(P)    RMTFIL:1      SRCDTA
C                   WRITE     FTPCMD        CmdRec
 *
C     FUNCTION      WHENEQ    'G'                                 GET function
C     'GET'         CAT(P)    RMTFIL:1      SRCDTA
C     SRCDTA        CAT(P)    LCLFIL:1      SRCDTA
C     SRCDTA        CAT(P)    '(Replace':1  SRCDTA              We use in this
 *                                                               example always
 *                                                               REPLACE option in
 *                                                               GET. Can be made
 *                                                               optional.
C                   WRITE     FTPCMD        CmdRec
C                   ENDSL
 *
C                   ENDSR
C/EJECT
```

Figure 366. ILE RPG program BLDFTP1R (Part 3 of 4)

```
*****************************************************************
     * Build and Write QUIT Record                                 *
     ****************************************************************
    C     CF005         BEGSR
     *
    C                   Z-ADD     *ZERO         SRCDAT
    C                   ADD       1             SEQ
    C                   Z-ADD     SEQ           SRCSEQ
    C                   MOVEL(P)  'QUIT'        SRCDTA
    C                   WRITE     FTPCMD        CmdRec
     *
    C                   ENDSR
```

Figure 367. ILE RPG program BLDFTP1R (Part 4 of 4)

Program CHKFTP1R is called by STRFTP1C. It reads through the physical file used as an FTP log to find out the result of the file transfer. It returns as a parameter, which is an indicator of the success of the transfer.

Notes: See Figure 368 and Figure 369 on page 276, and these notes:

1. FTP reply codes 226 and 250 are used by FTP servers to indicate the end of the transfer. FTP reply code 250 is used, for example, in transfers between two AS/400 systems.
2. Because FTP reply code 250 is used also for other functions, some of the explanatory part of the message has to be checked.

```
F/TITLE CHECK FTP LOG
   ************************************************************************
   *                                                                      *
   *                      PROGRAM FUNCTION                                *
   *                                                                      *
   * This program reads through the FTP transfer log file and             *
   * tries to find out if the transfer was successful.                    *
   *                                                                      *
   ************************************************************************
   *                                                                      *
   *                          ** NOTE **                                  *
   *                                                                      *
   * This material contains programming source code for your              *
   * consideration. These examples have not been thoroughly tested        *
   * under all conditions. IBM, therefore, cannot guarantee or imply      *
   * reliability, serviceability, performance or function of these        *
   * programs.  All programs contained herein are provided "AS IS".       *
   * THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A          *
   * PARTICULAR PURPOSE ARE EXPRESSLY DISCLAIMED.                         *
   *                                                                      *
   ************************************************************************
  F/SPACE 3
   ************************************************************************
   *                                                                      *
   *                       INDICATOR USAGE                                *
   *                                                                      *
   *   IND.  DESCRIPTION                                                  *
   *                                                                      *
   *    99 - FTPLOG EOF                                                   *
   *    LR - CLOSE FILES ON EXIT                                          *
   *                                                                      *
   ************************************************************************
  F/SPACE 3
  F/EJECT
   ************************************************************************
   *FILES USED BY THIS PROGRAM
   ************************************************************************
  FFTPLOG    IF   F  132        DISK                                      FTP transfer log fil
  F/EJECT
   ************************************************************************
   * DATA STRUCTURES USED BY THIS PROGRAM                                 *
   ************************************************************************
   *
   * Define constants
   *
  D FileXfer        C                   CONST('File transfer')
   *
   * Define input record format
   *
  D LogRec          DS           132
  D  ReplyCode                1      3
  D  ExtraInfo                5     17
  C/EJECT
```

Figure 368. ILE RPG program CHKFTP1R (Part 1 of 2)

```
     *****************************************************************
     * VARIABLE DEFINITIONS AND LISTS USED BY THIS PROGRAM           *
     *****************************************************************
    C/SPACE 2
    C     *ENTRY        PLIST
     * Return parameters:
    C                   PARM                    OK                3
     *                                                  possible values:   NOT = Not successfull
     *                                                                     YES = Successfull xfer
    C/EJECT
     *****************************************************************
     * The Main Program                                              *
     *****************************************************************
     *
    C                   MOVE      'NOT'         OK
     *
    C     *IN99         DOWEQ     *OFF
    C                   READ      FTPLOG        LogRec                  99
    C     *IN99         IFEQ      *OFF
     *
    C                   SELECT
     * Remote system was something else than an AS/400 or any other that uses 250 as a response
    C     ReplyCode     WHENEQ    '226'
    C                   MOVE      'YES'         OK
    C                   LEAVE
     *
     * Remote system was an AS/400 or any other that uses 250 as a response
    C     ReplyCode     WHENEQ    '250'
    C     ExtraInfo     ANDEQ     FileXfer
    C                   MOVE      'YES'         OK
    C                   LEAVE
     *
    C                   ENDSL
    C                   ENDIF
    C                   ENDDO
     *
    C                   EVAL      *INLR = *ON
    C                   RETURN
    C/EJECT
     *****************************************************************
     *                                                               *
     * ...and here be the dragons.                                   *
     *                                                               *
     *****************************************************************
```

Figure 369. ILE RPG program CHKFTP1R (Part 2 of 2)

After running successfully and transferring one source file member from one AS/400 system to another, the contents of the work files are shown in Figure 370.

```
 ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+
*************** Beginning of data *********************
SUNSHINE PASSWORD
namefmt 1
cd cool2red_1.lib
BINARY
GET hot.savf hot.savf (Replace
QUIT
****************** End of data ************************
```

Figure 370. FTP commands from FTPBATCH

Note: When you are using NAMEFMT 1 file names, notice that the maximum length for one FTP command, in this case, is 80 characters. You may want to use one or two of the optional FTP commands to change the current directories to proper ones.

```
Output redirected to a file.
Input read from specified override file.
Connecting to remote host 10.2.69.233 using port 21.
220-QTCP at as22.othercompany.com.
220 Connection will close if idle more than 5 minutes.
Enter login ID (sunshine):
331 Enter password.
230 SUNSHINE logged on.
 OS/400 is the remote operating system. The TCP/IP version is "V4R3M0".
250  Now using naming format "0".
257 "QGPL" is current library.
Enter an FTP subcommand.
> namefmt 1
250  Now using naming format "1".
Server NAMEFMT is 1.
Client NAMEFMT is 1.
Enter an FTP subcommand.
> cd cool2red_1.lib
250 "/QSYS.LIB/COOL2RED_1.LIB" is current library.
Enter an FTP subcommand.
> BINARY
200 Representation type is binary IMAGE.
Enter an FTP subcommand.
> GET hot.savf hot.savf (Replace
227 Entering Passive Mode (10,2,69,233,4,3).
150 Retrieving member HOT in file HOT in library COOL2RED_1.
250 File transfer completed successfully.
3493248 bytes transferred in 9.507 seconds. Transfer rate 367.446 KB/sec.
```

Figure 371. FTP log from FTPBATCH

6.8 FTP exit programs examples

For information about exit programs, and for the latest samples, refer to the AS/400 Information Center at: `http://www.as400.ibm.com/infocenter`

Using the search function, find `ftp exit program` information.

6.9 FTP security issues

If you use your AS/400 system as an FTP server on the Internet, be aware that it is accessible by the entire world. You can be sure that hackers will try to attack your computer. This is one reason why you *must* do something to secure your computer. We can give you some hints (without going into too much detail) about which points you must consider. Note that this is not a complete list:

- Is your Internet AS/400 system connected to any other system?
- Did you secure your AS/400 system or your network?
- Do you control the audit files on a regular base?
- Did you change *all* default passwords?
- Consider not forwarding IP datagrams (set by CHGTCPA).
- Control how many FTP servers you want to allow (set by CHGFTPA).

- Set a proper value for INACTTIMO (set by CHGFTPA).
- Have you implemented an object level security scheme?
- How reliable are your employees?

For additional security information, see Appendix C in the *TCP/IP Configuration and Reference*, SC41-5420.

We advise that you do not use your AS/400 system as an FTP server on the Internet without the following safeguards:

- A firewall should exist between the AS/400 system and the Internet.
- Use a non-production AS/400 system as your FTP server system. This AS/400 system should not be network-attached to the rest of your company's LANs or WANs.
- The user exit programs must be coded and tested to ensure that no security holes exist.

Even though your AS/400 system is not connected to the Internet, implement your own security rules and standards through the use of exit programs. See 6.6, “FTP exit programs” on page 262, for more information.

If you want to allow FTP clients to access your system, be aware of the following issues:

- Your object authority scheme may not provide detailed enough protection when you allow FTP on your system. For example, when a user has authority to view a file (*USE authority), the user can also copy the file to a PC or to another system. You may want to protect some files from being copied to another system.

  Beginning with V3R2, FTP exit programs are available to restrict the FTP operations that users can perform. You can use the FTP Request Validation Exit to control what operations you allow. For example, you can reject GET requests for specific database files. You can use the FTP Server Logon Exit to authenticate users who log on to the FTP server. An FTP exit program also defines an anonymous FTP user profile. Within this profile, you can limit the storage size allotted for an anonymous FTP user, therefore reducing the impact of being “bombed” by hackers. For an example of an FTP exit program, see 6.6, “FTP exit programs” on page 262. Also see “TCP/IP User Exits” in the *TCP/IP Configuration and Reference*, SC41-5420, for more information on FTP exit programs.
- As with Telnet, FTP passwords flow "in the clear" between the client and the server. Your FTP application may be vulnerable to sniffing.
- The QMAXSIGN system value applies to Telnet, but not to FTP. With FTP, the system breaks the connection after five unsuccessful sign-on attempts, but the user can simply establish a new connection. Theoretically, an FTP user has *unlimited* attempts to break into your system.

  The system writes message CPF2234 to the QHST log for each unsuccessful attempt. You can write a program to monitor the QHST log for the message. If the program detected repeated unsuccessful attempts, it can end the FTP server.
- If you plan to use FTP batch support to transfer files between systems, remember that the program must send a both user ID and password to the

server system. Either the user ID and password must be coded in the program, or the program must retrieve them from a file. Both of these options are a potential security exposure. You must use object security to protect the user ID and password information. You should also use a single user ID that has limited authority on the target system.

- FTP provides remote-command capability, just as APPC and AS/400 Client Access do. The Remote Command (RCMD) FTP-server subcommand is the equivalent of having a command line on the system. Before you allow FTP, you must ensure that your object security scheme is adequate. FTP exit points can be used to restrict the use of the RCMD subcommand.

If you *do not* want the FTP server to run on your system, perform the following steps:

1. Type:

   ```
   CHGFTPA AUTOSTART(*NO)
   ```

2. Make sure that the public authority for the Start TCP/IP Server (STRTCPSVR) command is *EXCLUDE.
3. To prevent someone from associating a user application, such as a socket application with the port that the system normally uses for FTP, complete the following tasks:
   a. Type `ADDTCPPORT` (Add TCP/IP Port Restriction).
   b. For the lower port range, specify `20`.
   c. For the upper port range, specify `21`.
   d. For the protocol, specify `*TCP`.
   e. For the user profile field, specify a user profile name that is protected on your system (not shared or adopted). By restricting the port to a specific user, you automatically exclude all other users. You can also specify a user profile that is used as a group user profile.

Again, port restrictions take effect the next time you start TCP/IP. If TCP/IP is active when you change the port restrictions, you should end TCP/IP and start it again. Make note of these changes in case you decide in the future that you want to remove these restrictions.

Chapter 7. Implementing the AS/400 system as a SOCKS client

Starting in V4R2, you can configure the AS/400 system as a SOCKS client. This allows you to use more functions from the AS/400 system through a firewall. You configure SOCKS support by using Operations Navigator.

7.1 Using Operations Navigator to access the SOCKS configuration

To access the SOCKS configuration function by using Operations Navigator, follow these steps:

1. Start Operations Navigator by clicking **Start->Programs->IBM AS400 Client Access->AS/400 Operations Navigator**. The AS/400 Operations Navigator window appears (Figure 372).

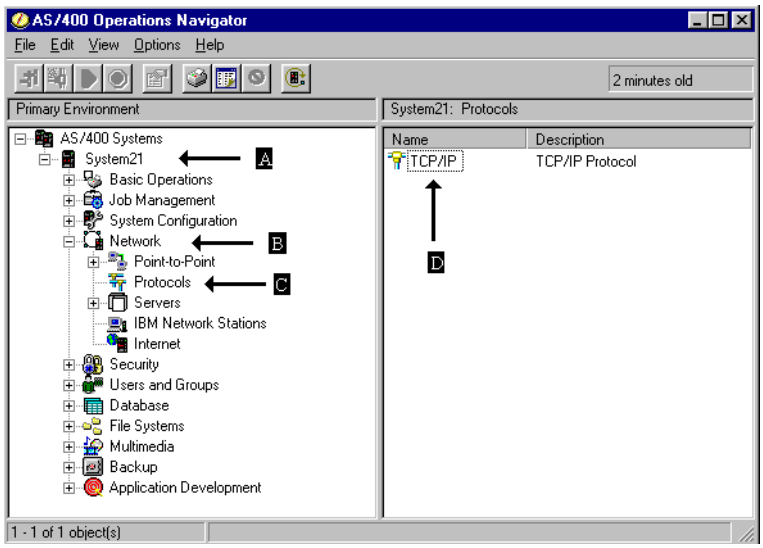


Figure 372. Operations Navigator: Network Protocol

2. Double-click the system icon (A) for the AS/400 system that you are configuring. The system components appear.
3. Double-click the **Network** icon (B). The network components appear.
4. Double-click the **Protocols** icon (C). The available protocols appear.
5. Double-click the **TCP/IP** icon (D) in the right window. The TCP/IP Properties window appears.
6. On the TCP/IP Properties window, click the **SOCKS** tab. The SOCKS information window appears (Figure 373 on page 282).

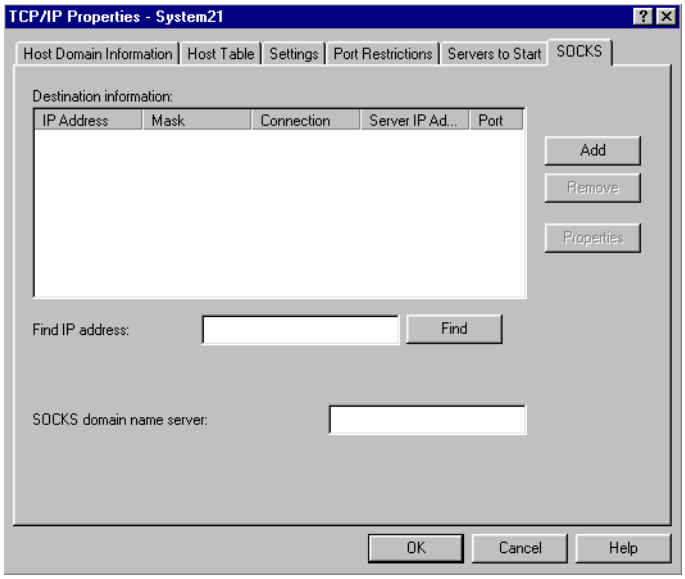


Figure 373. Operations Navigator: TCP/IP properties SOCKS before configuration

You are now ready to configure SOCKS information for your AS/400 system.

7.2 Configuring SOCKS for the AS/400 system

To configure the SOCKS information for the AS/400 system, you must provide at least two pieces of information:

- The network that is directly connected to the AS/400 system. A SOCKS server is not needed to reach the network.
- The network that requires the use of a SOCKS server for access and the SOCKS server to use to access the network.

As an option, you can add a DNS server to be used by SOCKS.

Note

The destination information in the SOCKS configuration is processed sequentially from the top of the list to the bottom. The first rule that matches the destination address is used to attempt the connection. Rule matching is determined by the following method. The destination IP address is ANDed with the mask in the table entry. If the result matches the IP address of that rule, the rule is used to process the packet. This makes it critical that any generic or catch-all entries (for example, address 0.0.0.0 with subnet mask 0.0.0.0) be positioned at the bottom of the list.

7.2.1 Defining the direct network

Do not use the SOCKS server to connect to any network that is directly connected to the system. To prevent the AS/400 system from connecting through the SOCKS server, the directly connected network should be defined.

To define the directly connected network, complete these steps:

1. In the SOCKS information window, click the **Add** button. The Add SOCKS Destination window appears (Figure 374).

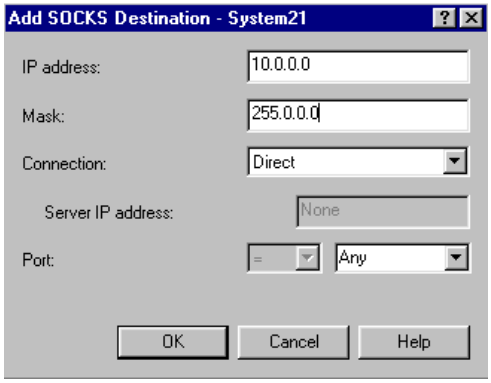


Figure 374. Add SOCKS Destination with direct connection information

2. Type the network address of the secure network in the IP address field. In our sample network, we use 10.0.0.0.
3. Type the subnet mask that describes your secure network in the Mask field. In our sample network, we use a subnet mask of 255.0.0.0.
4. Click the down arrow in the Connection field, and select **Direct** from the list of options.
5. Click **OK** to add the destination information.

You have now defined the “10.” network as a direct network. SOCKS does not access any host with an address that starts with “10.”

7.2.2 Defining the network connection using SOCKS

Now you must define the network to use with the SOCKS server. In this example, we use the SOCKS server to access all networks except the direct connection.

To define the network for use with SOCKS, follow these steps:

1. In the SOCKS information window (Figure 376 on page 285) highlight the IP Address (10.0.0.0 in this example) you want to insert this SOCKS Destination information after in the list. Click the **Add** button. The Add SOCKS Destination window appears (Figure 375).

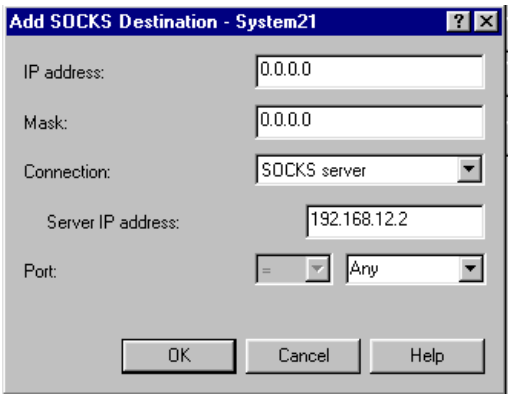


Figure 375. Add SOCKS Destination with SOCKS server connection

2. Type the address `0.0.0.0` in the IP address field.
3. Type the subnet mask `0.0.0.0` in the Mask field.

   When a destination address is “ANDed” with a mask of 0.0.0.0, the result is 0.0.0.0. By specifying a mask and address of all zeros, all IP addresses match this destination description.
4. Click the down arrow in the Connection field, and select **SOCKS Server** from the list of options.
5. Type the IP address of the SOCKS server in the Server IP Address field. On the AS/400 system with the firewall installed, this is the IP address of the *INTERNAL port of the firewall. On other AS/400 systems in the secure network, this is the IP address of the secure port of the firewall.
6. Verify that the Port field is set to **Any**. This specifies the remote ports for which this connection can be used.
7. Click **OK** to add the destination information.

You have now defined the destination information for SOCKS. You may also need to configure the SOCKS domain name server.

7.2.3 Defining the SOCKS domain name server

The SOCKS domain name server field specifies the IP address of a DNS server that can resolve names or IP addresses that reside on a non-secure network. Leave this field blank if the domain name servers configured with TCP/IP resolve the addresses.

For name or IP address resolution, the system queries the DNS servers configured with TCP/IP first. If they cannot resolve the name or address, the system queries the DNS server that you specify.

Note

At least one DNS server must be configured using CFGTCP option 12 before SOCKS checks the domain name server configured for SOCKS.

If you do not have an internal DNS server, point the AS/400 system at the firewall for DNS services. If the internal DNS server cannot resolve external information, type the IP address of the firewall in the SOCKS domain name server field. On the AS/400 system with the firewall installed, this is the IP address of the *INTERNAL port of the firewall. On other AS/400 systems in the secure network, this is the IP address of the secure port of the firewall.

After you enter all of your SOCKS information, your SOCKS information window should look similar to Figure 376. Click **OK** to save the configuration. The Operations Navigator window appears.

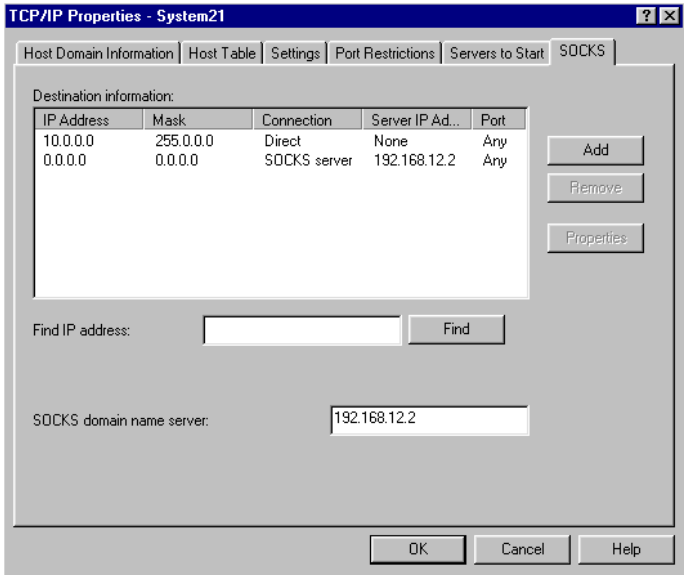


Figure 376. Pointing to the SOCKS domain name server

7.2.4 Testing your AS/400 SOCKS configuration

To quickly test your configuration, you can start a Telnet session with a system in the non-secure network (assuming that Telnet was enabled in the SOCKS server during firewall configuration). To test the configuration, perform these steps:

1. Sign on the AS/400 system.
2. On the AS/400 command line, type this command:

   ```
   telnet locis.loc.gov
   ```

   The US government's Library of Congress Information System display appears. To exit the system, type `12`, and press Enter. Then, type `12` and press Enter again.

If you do not receive the menu, you may have a problem with the DNS, firewall configuration, or network connection.

Chapter 8. Getting started with DNS on the AS/400 system

This chapter covers an overview of the DNS basic concepts as well as the basics of setting up the AS/400 as a DNS server. The Domain Name System protocol is described in RFC 1034 and RFC 1035.

Early Internet configurations required users to use only numeric IP addresses. This evolved quickly to using symbolic host names, which should be translated in some way to the corresponding IP address. It also introduces the problem to maintain the mapping between the IP addresses and high-level machine names in a centralized way. This mechanism became too cumbersome due to the explosive growth of the Internet and was replaced by DNS concept.

8.1 DNS overview

The Domain Name System is a distributed database, with a structure that's very similar to the structure of a file system. The complete database or file system can be described as an inverted tree with the root at the top. Each node in this tree represents a partition of the database. Each domain or directory can be further divided into partitions, called subdomains (such as the file system's subdirectories).

The domain name space has a "tree" structure. The top-level domains divide the Internet domain name space organizationally. Some examples of top-level domains are:

- **com**: Commercial organizations, such as IBM (ibm.com), CNN (cnn.com), mycompany (mycompany.com). Here, ibm is a subdomain of the top-level domain com.
- **edu**: Educational organizations, such as University of Minnesota (umn.edu), New York University (nyu.edu).
- **gov**: Government organizations, such as the Federal Bureau of Investigation (fbi.gov), and the National Science Foundation (nsf.gov).

Each node in the tree is labeled with a name (see Figure 377 on page 288). The root has a null label (" "). The full domain name of any node in the tree is the sequence of names on the path from the node up to the root with a dot between node names. For example, in Figure 377 on page 288, if you follow the arrows from the bottom label to the top, from the host www to the root label, you can form the full domain name for that host: `www.as400.ibm.com`

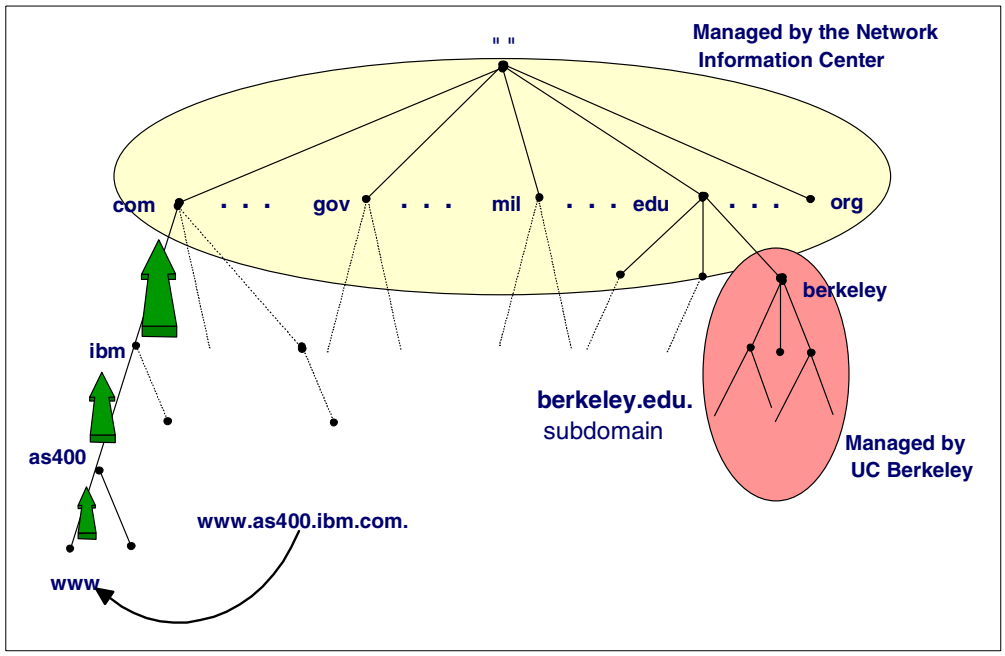


Figure 377. DNS name space

Each domain can be administered by a different organization. Those domains can be broken down into several subdomains for which other organizations can be responsible. This is because DNS uses a distributed database where you can manage your own domain (*company.com*), or parts of the name space (subdomains) can be delegated to other servers (*department.company.*com).

The DNS servers that are responsible for the top-level Internet domains, such as *com* or .*gov*, are also called *Internet root servers* that manage information about the top-level domains. For example, the Internet's Network Information Center runs the *edu* domain, but assigns U.C. Berkeley authority over the *berkeley.edu* subdomain.

Domains can contain hosts and other domains (those are their corresponding subdomains). The *ibm.com* domain, for example, contains hosts such as *www.ibm.com and* subdomains such as *as400.ibm.com*.

Domain names are used as indexes into the DNS database. Each host on a network has a domain name with a DNS server that points to information about the host. This information may include an IP address, information about mail routing, and so on.

Why all this complicated structure? It helps to solve the problems that a host table has. For example, making names hierarchical eliminates the problem of name collisions. Domains are given unique domain names, so organizations are free to choose names within their domains. Whatever name they choose, it does not conflict with other domain names, since it has its own unique domain name.

For example, you can have several hosts named *www*, such as *www.ibm.com* and *www.yahoo.com*, because they are in different domains managed by different organizations. See Figure 378.

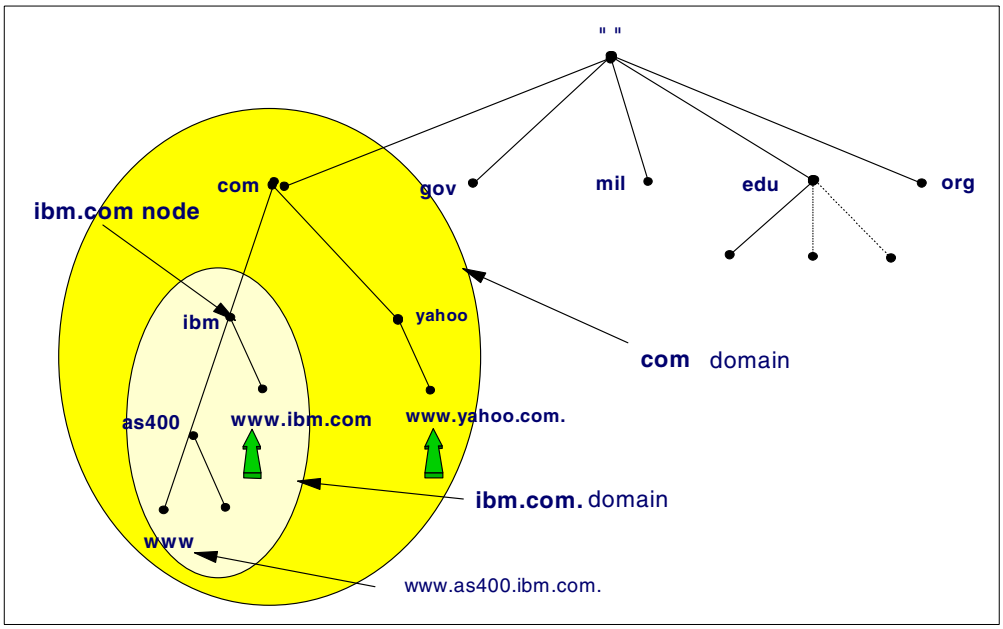


Figure 378. Same host names within different domains

Figure 378 shows that we can have www as a host in the domain ibm.com, which that has the same host name within the subdomain as400.ibm.com.

8.2 Domain versus zone of authority

Decentralization is one of the main goals of the design of the Domain Name System. This is achieved through delegation. The company's domain can be divided into subdomains by the central DNS administrator within your company. Each subdomain can be delegated to other administrators who will be responsible for maintaining it.

A *domain* is a subset or subtree of the name space tree. A subdomain is a subset of the domain itself. Figure 379 on page 291 shows the domain mycompany.com as a subset of the .com name space. Under mycompany.com, there are other subdomains, such as endicott.mycompany.com, rochester.mycompany.com, and otherdomain.mycompany.com.

Name servers are programs running on a system, such as the AS/400 system, with DNS support. In Figure 379 on page 291, as1.mycompany.com, rst.rochester.mycompany.com, and otherhost.otherdomain.mycompany.com are hosts running name server programs. They are called *Domain Name System* (DNS) servers, or simply *name servers*.

Name servers have information about some part of the domain name space called a *zone* or *zone of authority*. Both domains and zones are subsets of the domain name space. A zone contains host information and data that the domain contains excluding the information that is delegated somewhere else. If a subdomain of a domain is not delegated, the zone contains host information and data for the subdomain as well.

Name servers have complete host information and data for a specific zone. Name servers are said to be authoritative for the zone for which they have this complete host information and data.

Refer to Figure 379. The mycompany.com domain is divided into the subdomains endicott.mycompany.com, rochester.mycompany.com, and otherdomain.mycompany.com. The zone mycompany.com contains the hosts as1.mycompany.com, as2.mycompany.com, as5.mycompany.com, and NTserver1.mycompany.com.

It also contains the host information and data in the subdomain endicott.mycompany.com: host1.endicott.mycompany.com and host2.endicott.mycompany.com. The subdomain endicott.mycompany.com has not been delegated, and its host information and data remain in the mycompany.com zone. The administration of the endicott.mycompany.com is the responsibility of the mycompany.com administrator. AS1.mycompany.com is the name server that has complete host information and data for the mycompany.com zone of authority.

The zone mycompany.com does not contain information in the subdomains that have been delegated.

rochester.mycompany.com is a subdomain of mycompany.com, and its administration has been delegated. The zone rochester.mycompany.com includes host information and data in the subdomain rochester.mycompany.com: rst.rochester.mycompany.com, host1.rochester.mycompany.com, and host2.rochester.mycompany.com. rst.rochester.mycompany.com is the DNS server that has complete host information and data for the rochester.mycompany.com zone.

otherdomain.mycompany.com is a subdomain of mycompany.com and its administration has been delegated. The zone otherdomain.mycompany.com includes host information and data in the subdomain otherdomain.mycompany.com: otherhost.otherdomain.mycompany.com, otherprinter.otherdomain.mycompany.com, and otherserver.otherdomain.mycompany.com. otherhost.otherdomain.mycompany.com is the DNS server that has complete host information and data for the otherdomain.mycompany.com zone.

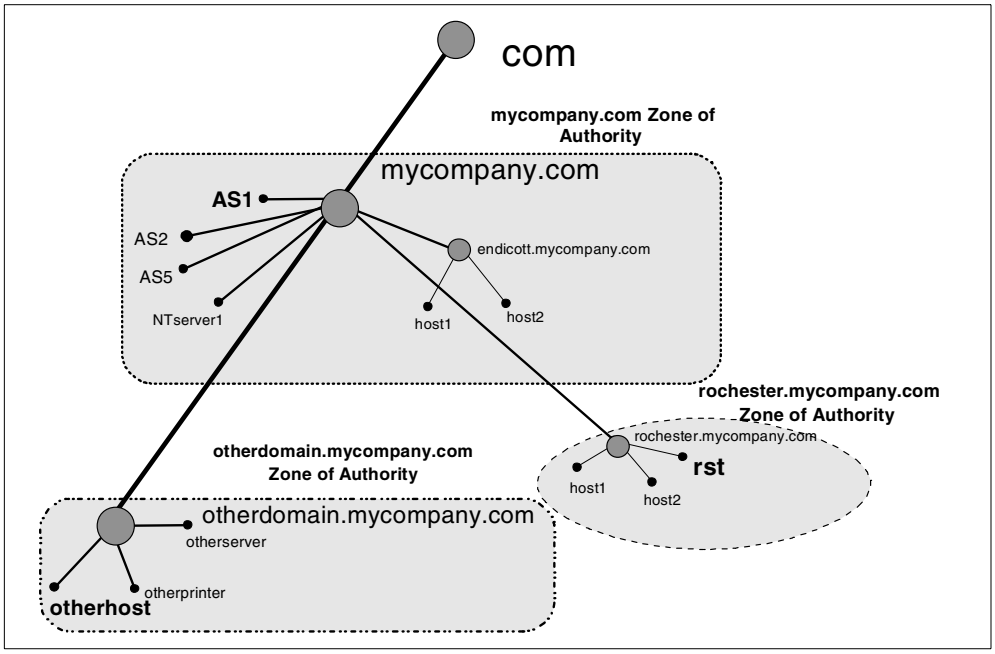


Figure 379. Domain, subdomain, delegation, and Zone of Authority

8.3 Name resolution process

DNS is an example of the client server principle. Name servers are programs representing the server half of the DNS client/server mechanism. They contain information about some segment of the DNS database and make this available to clients (called *resolvers*).

The Domain Name System has two major components:

- **Name servers**: Programs that hold information about the domain name space. It may cache information about any part of the domain tree. In general, a particular name server has complete information about a subset of the domain space and pointers to other name servers that can be used to lead to information from any part of the domain tree. The part of the domain space the name server has complete information for is called a zone. It is said that the name server is *authoritative* for that zone. Name servers can be authoritative for multiple zones.
- **Resolvers**: Programs that extract information from name servers in response to client requests. Resolvers must be able to access at least one name server and use that name server's information to answer a query. A resolver is typically a system routine that is directly accessible to user programs. No protocol is necessary between the resolver and the user program.

The process of mapping names to addresses (also called domain name resolution), is provided by independent, cooperating systems called *servers*. A name server is a server program answering requests from clients (also called *name resolvers*).

Each name resolver is configured with a name server to use (and possibly a list of alternatives to contact in case the primary name server is not available).

Figure 380 describes how a program uses a name resolver to convert a host name to an IP address on the Internet. A user provides a host name, and the user program uses a system function, called a resolver, to communicate with a name server that resolves the host name to an IP address and returns it to the resolver, which returns it to the main program. The name server may obtain the answer from its name cache (if it has tried to resolve the same name already before), its own database, or another name server.

In Figure 380, the resolver sends a query for www.as400.ibm.com to its DNS server (this one is labeled primary name server). If the query is for information out of the name server's zone of authority (it does not know the answer), the name server sends another query to the Internet root name server, which responds back, "I don't know but query this next DNS server (the com DNS server)." And the query is iterated to various DNS servers down the "com" branch of the Internet DNS name space until the DNS server is found that is authoritative (is responsible for) the as400.ibm.com domain. This last DNS server has the answer and sends the response back to the original DNS server the resolver asked for, which then passes the response back to the resolver.

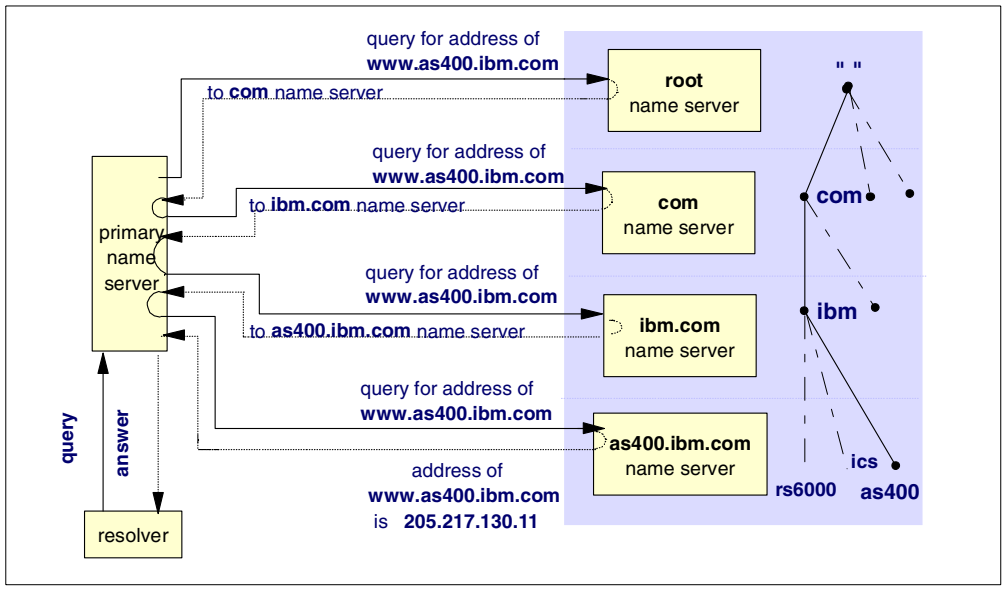


Figure 380. Example of the name resolution process

8.3.1 Recursive versus iterative queries

There are two types of DNS queries: *recursive* or *iterative*. Figure 380 shows an example of one recursive query and several iterative queries. The first query from the resolver to the primary name server is a recursive query. A recursive query requests that if the name server does not know the answer to the query, it queries other name servers until it finds the answer and then sends the answer back to the resolver. Notice in Figure 380, the primary name server did a lot of work. It kept querying other name servers on behalf of the resolver until it could supply the answer. A DNS server is configured to accept recursive queries or only accept iterative queries. The primary name server in Figure 380 was configured to allow recursive queries.

The other name servers queried in Figure 380 (root name server, com name server, ibm.com name server) were not configured to allow recursive queries. When the primary name server queried the root name server, the query was an iterative query. This means the root name server responded to the query with the best information it had, which was, “I don’t know but check the next DNS server: com name server.” The recursive query versus iterative query only comes into play when the name server queried does not know the answer to the query.

8.4 Name server types

This section describes the types of name servers. These include:

- **Primary name server**

  This is the server on which the hosts in the zone of authority are configured. This server is configured and maintained by the DNS administrator. When this server gives responses to queries from its primary domain files, those responses are called authoritative. A name server for a primary domain reads the primary domain configuration information directly from files configured by its DNS administrator.

- **Secondary name server**

  This server has the same information as the primary name server. It gets its information from another name server through zone transfers over the network, instead of getting its information directly from the DNS administrator who is configuring it. The information that a secondary name server obtains from a zone transfer is read into cache, as is data stored from queries.

  Note

  A DNS server can be a primary name server for one or more domains, as well as a secondary name server for one or more domains. It can be a name server for primary and secondary domains.

  A *zone transfer* is a transfer of domain files from another DNS server (this is also called a master name server) by using TCP/IP. This is done automatically when the secondary name server starts and also when the secondary name server detects its domain files are downlevel compared to the master name server’s domain files. The zone transfer is initiated from the secondary name server and cannot take place if the master name server is not active.

  A secondary name server is used for two reasons:

  – The DNS query workload is spread over more than one server.
  – It serves as a backup in case the primary name server stops responding.

  When a client is configured with more than one DNS server and the first name server (the primary) does not respond, the client can query the second name server (the secondary). When the secondary name server gives out a response to a query, the response is also called authoritative.

- **Master name server**

  This is the name server from which a secondary name server gets its zone transfer. A master name server can be a primary name server, but it can also be another secondary name server.

- **Caching-only name server**

  A name server that does not have authority over any zone is called a caching-only name server. It gets all of its information by querying. A caching-only name servers stores the responses that it receives to queries. When another query for the same information comes in, it serves the information it has stored rather than issuing another query for the information. If the first response to the query came from an authoritative source, the response that the caching-only server sends out is also authoritative.

- **Authoritative name server**

  A server that is considered to be authoritative for a domain is either the primary server for that domain or a secondary server for that domain.

- **Parent and child name servers**

  The concept of parent and child domains is equivalent to the concept of domain and subdomain. If your domain is growing continuously, you may need to distribute management by delegating authority of part of your domain to one or multiple subdomains. The upper-level domain is the parent and its subdomains are the children.

  The name server authoritative for the parent domain is the parent name server and the one authoritative for the subdomain is the child name server. In Figure 379 on page 291, OTHERDOMAIN is a subdomain of the mycompany.com domain. If a DNS server, AS1, is configured to be responsible for the mycompany.com zone of authority and the authority for the zone OTHERDOMAIN.mycompany.com is delegated to another DNS server, OTHERHOST, AS1 is considered to be the parent name server, and OTHERHOST is considered to be the child name server.

- **Root name servers**

  Internet root name servers know where name servers authoritative for the top-level domains are, and most of the Internet root name servers are authoritative for the top-level organizational domains (.com, .edu, .net, and so on). The top-level domain servers have information about the second-level domain in which a given domain is located.

  A company can implement internal root name servers. In this case, given a query for a company's subdomain, the internal root name server can provide information for the second-level subdomain in which the queried subdomain is located.

  A root name server is configured in a lower level name server to help it to navigate the name space tree top down when it cannot answer a query with authoritative data or data in its cache.

- **Forwarders**

  A DNS server can also be configured to send the queries it does not know the answer, to a DNS server called a *forwarder name server*. Where going to a root name server for help in answering a query can be thought of as going to the top of the DNS name space tree, going to a forwarder can be thought of as going side-ways in the DNS name space tree for help. The DNS administrator has to configure which DNS server needs to be a forwarder. Usually, several DNS servers are configured to have the same forwarder. Then, the forwarder name server is configured with the root name servers (for example, the Internet root name servers). If the forwarders cannot answer the query, they

query the root name servers, get the answer, and cache it. This way, a forwarder name server can build up a large cache of information. As the cache increases, chances are that the forwarder will receive a query for which it has a cached answer. This, in turn, reduces the number of times a root name server needs to be queried. Using a forwarder name server is an opportunity to build a large cache of information on one (or just a few) name server.

If your DNS server is behind a firewall, you will point a forwarder to the firewall because your DNS queries are typically blocked by the firewall. The firewall receives the query, processes it, and returns the results of the query back to your DNS server.

8.5 DNS file types

This section describes the various DNS file types. These files include:

- **Primary domain files**

  These files are the files configured on the primary name server. These can be found on the AS/400 system within the IFS directory: /QIBM/UserData/OS400/DNS

  These domain files end with a .DB extension

- **Secondary domain backup files**

  These files contain information that is acquired from zone transfers from the primary name server. These exist on the secondary name server. A secondary name server loads these files and uses them to answer queries provided the zone transfer was successful.

  These domain files end with a .DB extension

- **Forward mapping files**

  Forward mapping primary domain files reside on the primary name server. They contain all data for mapping host names to IP addresses in a zone. A DNS server is authoritative for a certain part of the DNS name space tree. This part of the tree is called a zone or the DNS server's zone of authority.

  Note

  Every forward mapping primary domain file should be configured with the host localhost with an IP address of 127.0.0.1.

- **Reverse mapping files**

  The reverse mapping primary domain files reside on the primary DNS server. They contain the information for mapping IP addresses to host names in a zone. They are also called the in-addr.arpa files. An example of a reverse mapping file is the 69.5.10.in-addr.arpa file. This is the file a DNS server uses if a client resolver queries with an IP address of 10.5.69.222 and asks the DNS server to supply the host name belonging to that IP address. The 69.5.10.in-addr.arpa file also resides in the AS/400 IFS directory /QIBM/UserData/OS400/DNS with a file name of 69.5.10.in-addr.arpa.DB.

  These domain files end with a .DB extension

- **Boot file**

  The boot file is the file that the DNS server first reads when it starts. It contains such information as:

  – The type of name server
  – The zones for which this name server is authoritative
  – Where (file location) the name server should get its information

  The boot file is also located within the directory /QIBM/UserData/OS400/DNS.

> **Note**
>
> If the AS/400 DNS has never been configured, the boot file does not exist. The first time a user clicks on DNS configuration within Operations Navigator, the wizard windows are shown and the boot file is created.

- **Cache file**

  The cache file contains information about the root name servers. This is where the DNS server should go when it cannot resolve a query itself. This file is located in the /QIBM/UserData/OS400/DNS directory.

- **Local file**

  The local file contains the PTR record for the local loopback interface. The loopback interface (or localhost) has the IP address of 127.0.0.1. Hosts use the 127.0.0.1 IP address to direct TCP/IP traffic to themselves.

8.6 DNS record types

The information contained in forward and reverse primary domain files are organized into records called *resource records*. There are several types of resource records. We try to explain the most common ones in the following list, although the list is not complete. For more details on resource records, see the third edition of *DNS and BIND* by Albitz & Liu.

- **A record**:

  This record maps a host name to an IP address. There is one A record for every host configured in the DNS server. Consequently, a query that supplies the host name and asks for the IP address is sometimes called an A record query. A records are contained in the forward mapping primary domain file. This type of query is also called a forward mapping query.

- **PTR record**:

  This record maps an IP address to a host name. There is usually one PTR record for every host configured in the DNS server. These records are located in the reverse mapping primary domain files, which are also called the *in-addr.arpa files*. A query supplying the IP address and requesting the host name is sometimes called a *reverse mapping query*, a *reverse lookup*, or an *in-addr.arpa query*.

- **SOA record**:

  This is the first record in the forward and reverse mapping primary domain files. The SOA record marks the zone of authority in the domain name space. It contains the domain name, the name of the DNS server that is primary for

this zone of authority, and the e-mail address of the zone's technical contact. The SOA record also contains the file's serial number. The serial number can be thought of as the change level of the data in this zone. In other words, if a DNS configuration change is made to this zone, the serial number must be incremented (Operations Navigator does this automatically). Also, the SOA record contains refresh timers, retry rates, and expire timers, all having to do with secondary name servers.

- **CNAME record**:

  The CNAME record defines the canonical name of an alias. It is used to specify an alias name for a host.

- **MX record**:

  This record defines a mail exchanger host for a particular domain. This record is used by SMTP to send mail.

- **NS record**:

  This record defines a name server to this name server: either itself or another name server. The other name server can be a name server authoritative over another domain. Or, the other name server can be a secondary name server to this same zone of authority. It is the NS records that allow each name server shown on the right side of Figure 380 on page 292 to tell the primary name server where to query next when it is searching for the answer to the resolver's query. NS records allow a DNS server to find other DNS servers authoritative for other zones.

8.7 AS/400 DNS implementation

This section provides an overview of the AS/400 implementation of DNS support. It provides the basic information you need to know to install, configure, and use the DNS support on the AS/400 system. This section also provides some work management and debugging information.

8.7.1 Software prerequisites

To run the native DNS support on the AS/400 system in V4R2, the following products are required:

- 5769-SS1 OS/400 V4R2 option 31: Domain Name System
- 5763-XD1 V3R1M3: Client Access for Windows 95/NT

8.7.2 DNS installation

For the AS/400 DNS support, you need to install the following products:

- 5769-SS1 OS/400 V4R2 option 31: Domain Name System on the AS/400 system. You can use `GO LICPGM` option `11` (Install licensed programs) to install the DNS OS/400 option.
- Client Access for Windows 95/NT (5763-XD1 V3R1M3) on your administrator's workstation.

The installation program performs the following operations:

- Installs the product library QDNS. It includes the product's objects (programs, message files, job descriptions, and so on).

- Creates two IFS subdirectories: /QIBM/ProdData/OS400/DNS and /QIBM/UserData/OS400/DNS.
- Creates two files: TEMPLATE and ROOT.FILE. These are put in the /QIBM/ProdData/OS400/DNS subdirectory. TEMPLATE is used as a template to create all the DNS configuration files (BOOT, CACHE, and configuration files). ROOT.FILE holds information on root name servers needed to initialize the cache of Internet domain name servers.
- Creates the ATTRIBUTE file and TMP directory under the /QIBM/UserData/OS400/DNS subdirectory.

You can now proceed with the DNS server configuration using the Operations Navigator interface. Figure 381 gives an overview of the AS/400 DNS server installation and configuration.

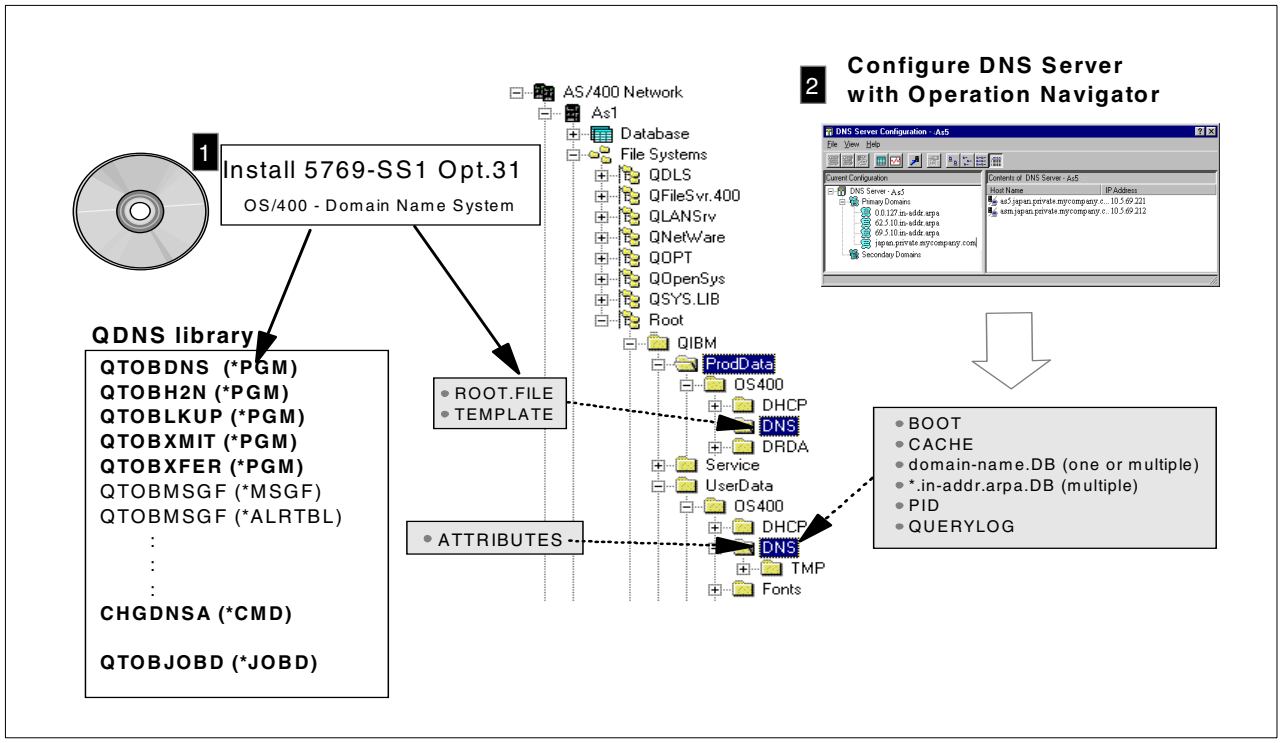


Figure 381. DNS support installation and configuration overview

8.7.3 DNS server jobs

The DNS server jobs run in the QSYSWRK subsystem. These jobs include:

- **QTOBDNS**: This is the DNS server job. This starts with the job description QDNS/QTOBJOBD. DNS is using well-known port 53. DNS server messages are directed to the QTOBDNS job log. Use the Work with Spooled File (WRKSPLF) command for User QTCP to browse the DNS server job log.
- **QTOBXMIT**: This is the zone transfer job that runs on the AS/400 system acting as the primary master name server for a specific domain.
- **QTOBXFER**: This is the zone transfer job that runs on the AS/400 system acting as the secondary name server for a specific domain.

8.7.4 DNS configuration files

All of the DNS configuration files can be found in the IFS directory /QIBM/UserData/OS400/DNS. These files include:

- **Domain or forward mapping file (Domain_Name.DB)**: Maps host names to IP addresses. The entries in this file are called *resource records*. This file has the same name as the domain with the .db extension.
- **Reverse mapping files (IP_address.in-addr.arpa.DB)**: Map addresses back to host names. There is one file for each subnet address in the network where the domain's hosts reside.
- **Loopback address file (0.0.127.in-addr.arpa.db)**: Covers the loopback network used by the hosts to direct traffic to themselves.
- **BOOT file (BOOT)**: The DNS server startup file that ties all the DNS configuration files together.

Figure 382 shows the relationship between the BOOT file and the *.db files.

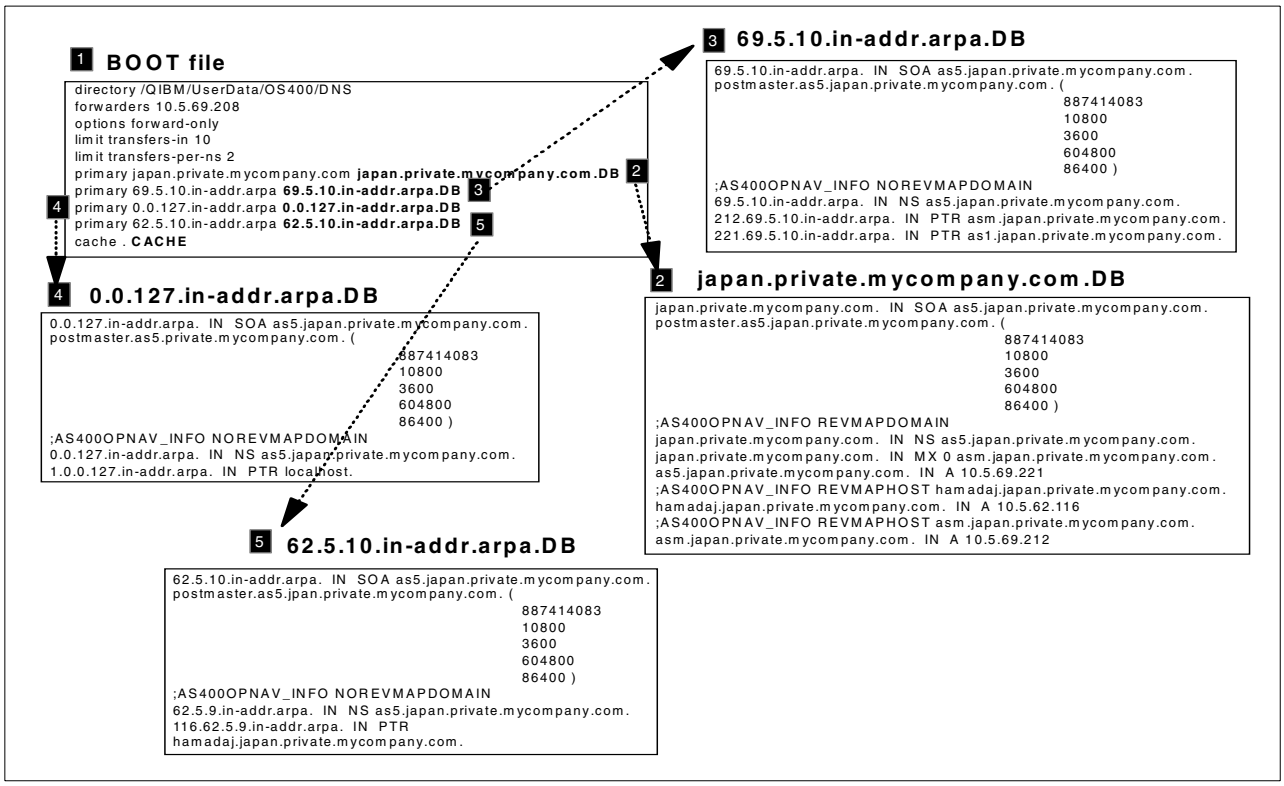


Figure 382. DNS configuration files overview

8.7.5 Logging and problem determination

The following files log the DNS server activity and can be used for problem determination reasons:

- **QUERYLOG**: The DNS server uses this file to log each query that it receives if it is configured to do so. You can use the Operations Navigator to view the contents of the log. The file name is QUERYLOG in the directory path FileSystems\Root\QIBM\UserData\OS400\DNS for your AS/400 system.

Carefully consider whether you need to log all queries and for how long. There is no limit to the size of the log file. Once you turn it on, it remains on until you disable logging and reboot the DNS server. Figure 383 shows how to specify that you want the DNS server to log all the queries it receives in the QUERYLOG file.

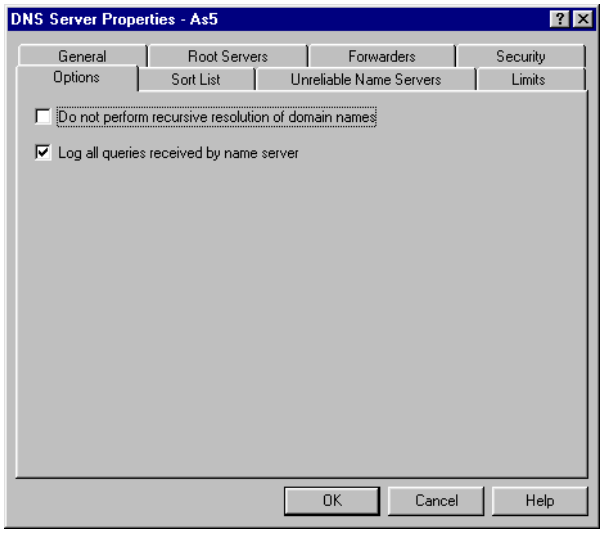


Figure 383. Configuration of the DNS server logging: QUERYLOG file

- **STATISTICS**: This file enables DNS server statistics logging. It summarizes the number of query hits the server received and the number of output packets it sent since the last time the server was rebooted or reloaded its database. You should delete this file when it becomes too large. Otherwise, you may need to scroll down several times to find the information for which you are looking. You can find it through the Operations Navigator interface. The file name is STATISTICS in the FileSystems\Root\QIBM\UserData\OS400\DNS directory path for your AS/400 system. Figure 384 shows how to display the DNS server statistics.

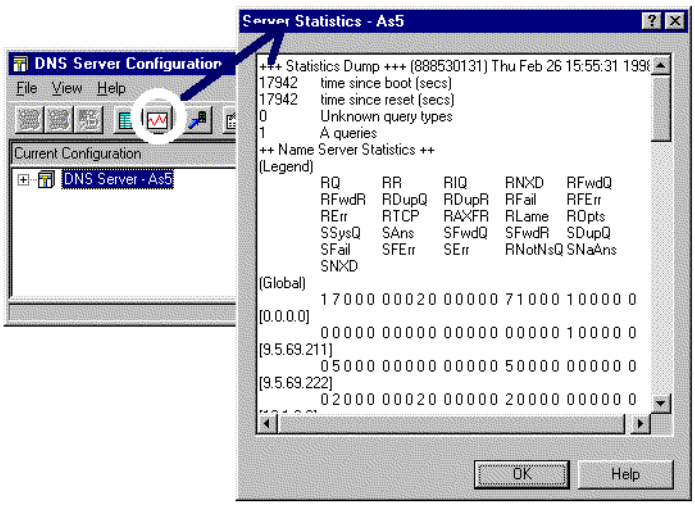


Figure 384. DNS Server Statistics

- **DUMPDB**: This file contains a dump of the DNS database for this server. You can use this database dump as a debugging tool to determine whether the DNS server is resolving IP addresses to host names correctly. You can match the contents of the database dump to the contents of a particular host's property pages. The database dump includes the DNS server's authoritative data and cache data as well as information about its root servers. Figure 385 shows how to display the dump of the DNS server database. Monitor the size of this file to prevent it from growing too large.

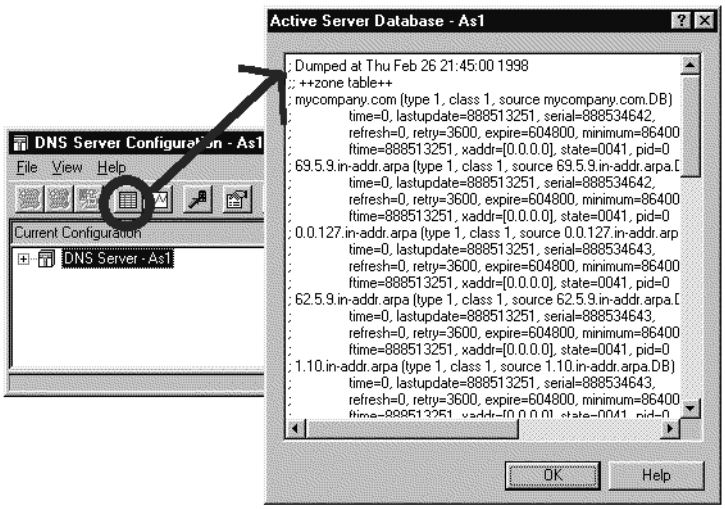


Figure 385. DNS Server Database display

- **RUNDEBUG**: It logs any debugging information. You can use Operations Navigator to find this file in the FileSystems\Root\QIBM\UserData\OS400\DNS directory for your AS/400 system. You must reboot the server for your changes take effect. Figure 386 shows how to you specify the debug level within Operations Navigator.

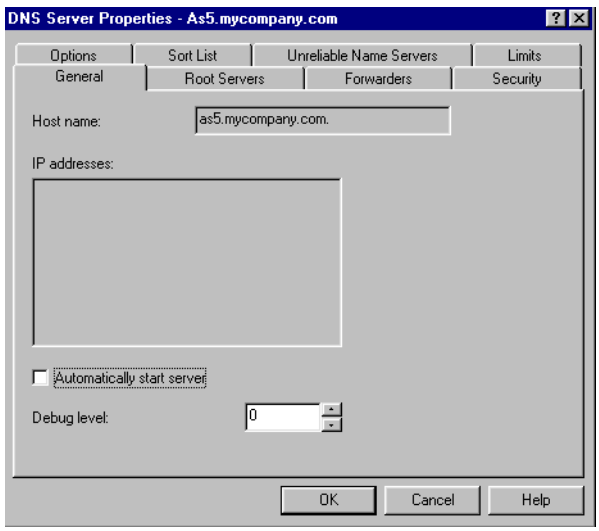


Figure 386. Debug level specification

A Debug level of zero means that no debug information is logged. A Debug level 1 through 11 means logging will occur. Level 3 or greater will result in a lot of data.

If you specify a Debug level other than zero, the system continuously appends information to RUNDEBUG until you reboot the server again. Monitor the RUNDEBUG file often enough to ensure that it does not grow too large for your needs. Information continually appends to this file until you delete the file.

- **ATTRIBUTES**: This file contains the DNS server version, debug level, and autostart attribute.
- **PID**: This file contains a process ID, and it is used for DNS to send signals for Dump Database, Dump Statistics, and Update Server.

Figure 387 provides an overview of the DNS server jobs, files, and logs.

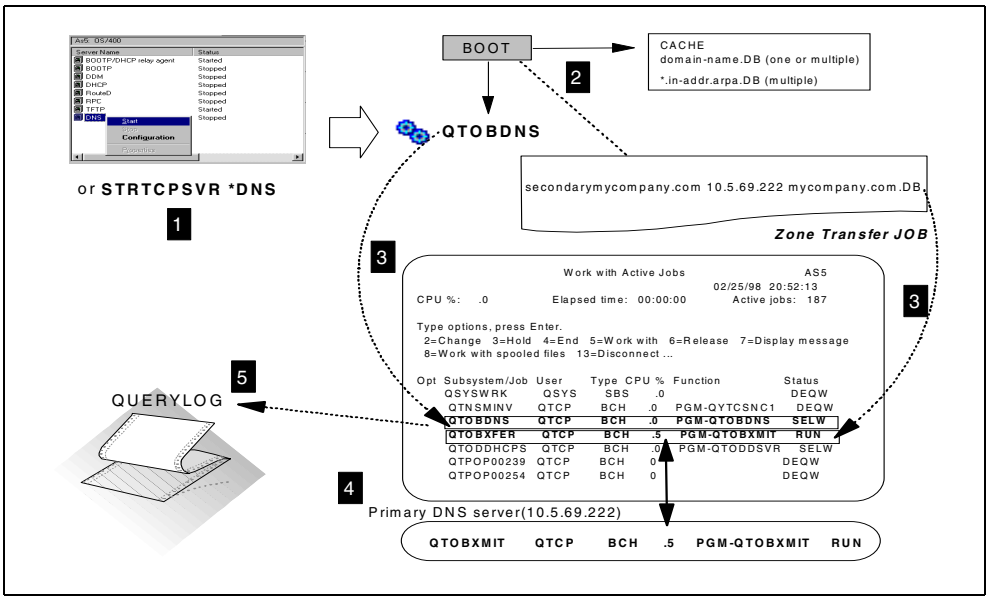


Figure 387. DNS server jobs, files, and logs

The following process occurs:

1. The user starts the DNS server.
2. The BOOT file provides such startup information as:
   - The location of the configuration files
   - The server role (primary and/or secondary for specific domains)
   - The CACHE file with root name servers data (if acting as a secondary name server)
   - The IP address of the primary master server to transfer zone data from
   - Forwarders information
3. The DNS and zone transfer jobs start.
4. The name server is now ready to answer queries.
5. DNS queries are logged in the QUERYLOG file if logging is turned on.

8.7.6 User interface

This section describes the user interface available in AS/400 DNS server. The configuration of the AS/400 DNS server is done through Operations Navigator. It is the only configuration interface for the DNS server. The Operations Navigator DNS configuration wizard provides a simple process for quickly getting an initial DNS server up and running.

To start the DNS server configuration through the Operations Navigator interface, select your AS400 system name. Then, select **Network->Servers->TCP/IP**. Right-click **DNS**, and select **Configuration** as shown in Figure 388.

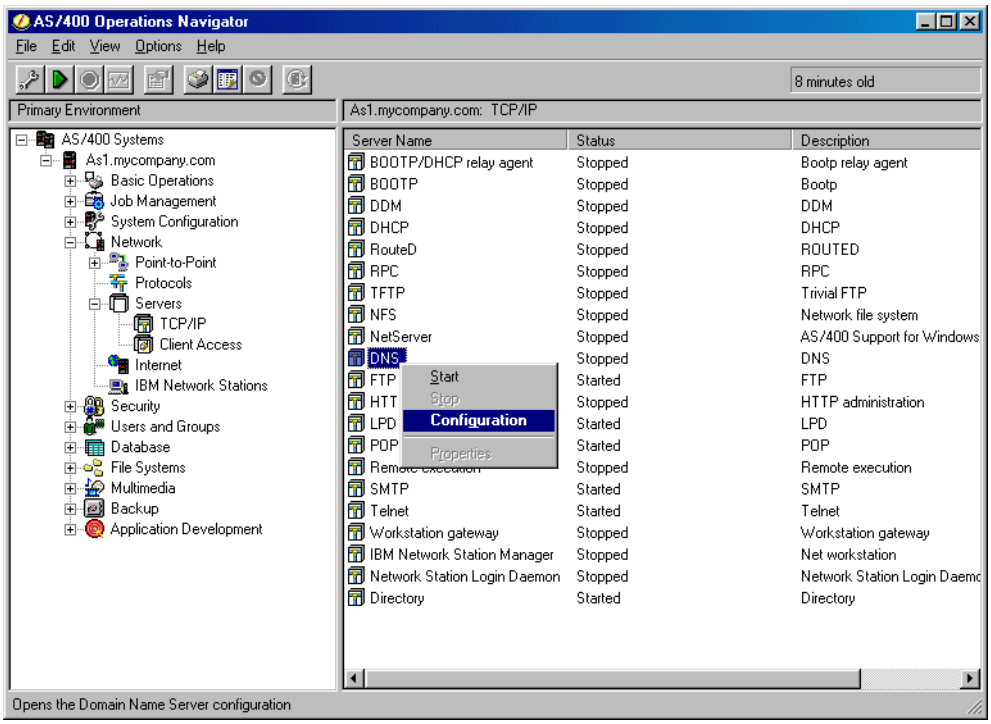


Figure 388. DNS configuration using Operations Navigator

To use Operations Navigator, you need to install Client Access/400 for Windows 95/NT V3R1M3 on the PC of the DNS Administrator. The host servers need to be started on your AS/400 system. You can use the Start Host Server (`STRHOSTSVR`) command to start them.

8.7.6.1 Changing the DNS attributes

You can use the Change DNS Attributes (`CHGDNSA`) command to set the AUTOSTART attribute. It determines whether the DNS server will start automatically when TCP/IP is started using the STRTCP command. This attribute is ignored by the STRTCPSVR command. The STRTCPSVR *DNS command starts the DNS server, regardless of the value of the AUTOSTART attribute. This attribute can also be set from the Operations Navigator interface. The CHGDNSA command allows you to set the debug level that can also be specified through Operations Navigator.

8.7.6.2 Starting the DNS server

Use the `STRTCPSVR SERVER(*DNS)` command to start the DNS server. This function can also be performed through the Operations Navigator interface. To start the

DNS server through the Operations Navigator interface, select your AS400 system name. Then, select **Network->Servers->TCP/IP**. Right-click **DNS**, and select **Start** as shown in Figure 389.

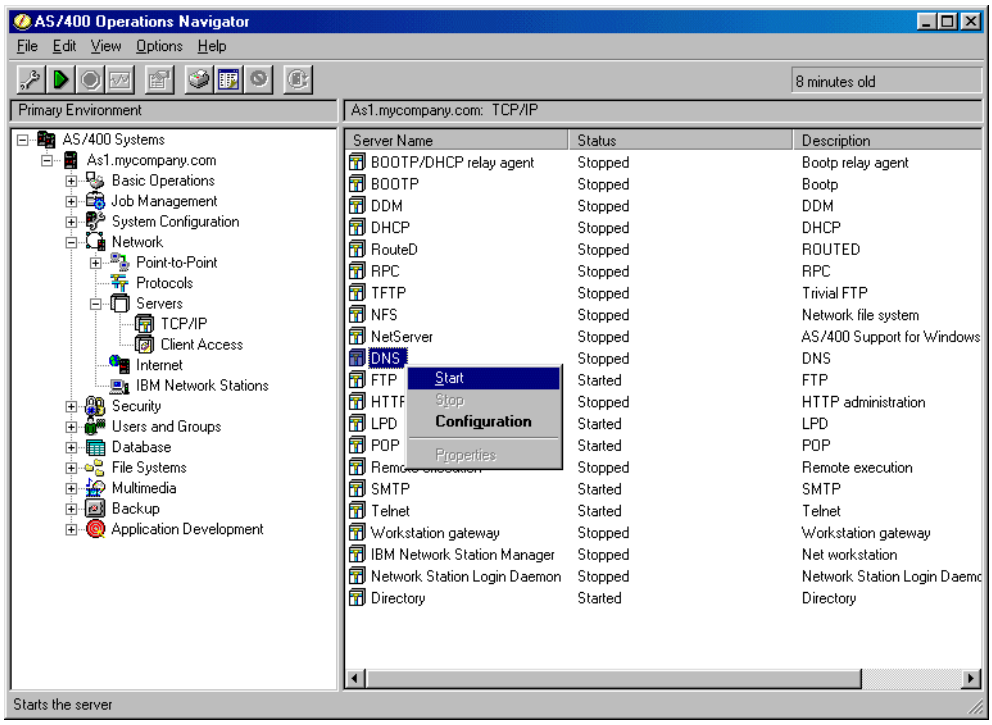


Figure 389. Starting the DNS server through Operations Navigator

8.7.6.3 Ending the DNS server

Use the `ENDTCPSVR SERVER(*DNS)` command to end the DNS server. This function can also be performed through the Operations Navigator interface. To stop the DNS server configuration through the Operations Navigator interface, select your AS400 system name. Then, select **Network->Servers->TCP/IP**. Right-click **DNS**, and select **Stop** as shown in Figure 390.

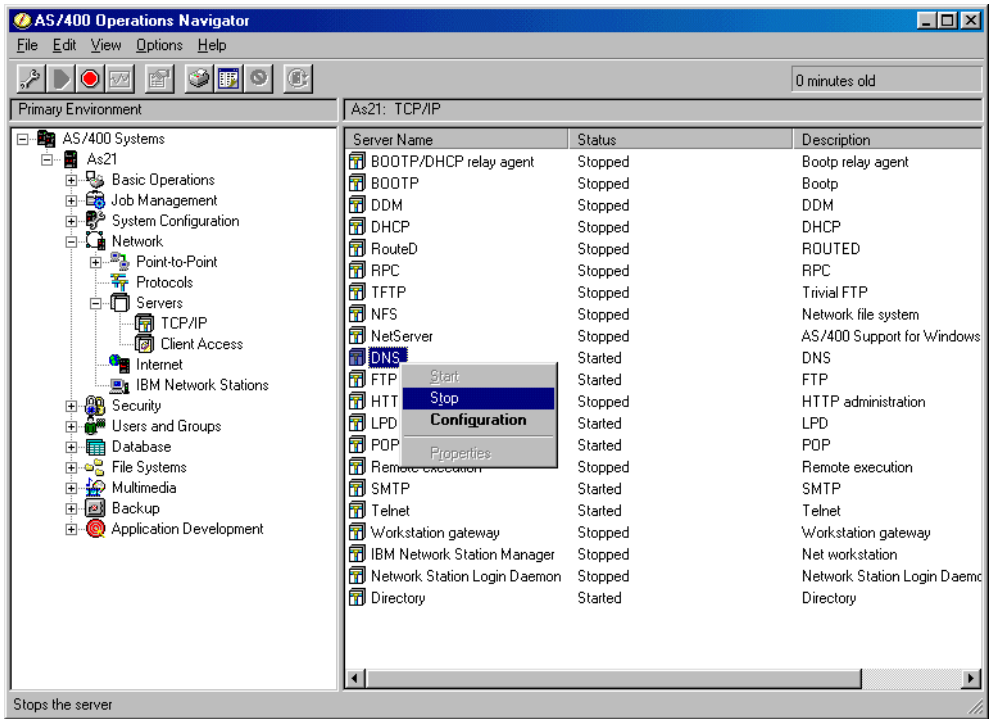


Figure 390. Ending the DNS server through Operations Navigator

8.8 The NSLOOKUP program

The AS/400 name server lookup (nslookup) program queries domain name servers in interactive mode. This allows you to query name servers to get information about various hosts and domains or to display a list of hosts within that domain.

You may also consider using a tool on a PC to do nslookup functions. One tool that we used during our testing is *CyberKit*. For this and many more tools, visit the Web site at: `http://www.tucows.com/`

To access the AS/400 nslookup function in V4R2, you must make a program call. Use the command:

```
call pgm(qdns/qtoblkup)
```

To access the nslookup function in V4R3 and beyond, use the NSLOOKUP command. On an AS/400 command line, type:

```
NSLOOKUP
```

Press Enter.

You may also type the AS/400 style command `STRDNSQRY` and press Enter. Either command starts the nslookup function.

To do a simple lookup, type the name, and press Enter. The information returned by the DNS appears on the screen.

Refer to the Web site at: `http://www.as400.ibm.com/infocenter/`

At the site, search for NSLOOKUP for more information on the nslookup program. You can also find information in the book *AS/400 TCP/IP Autoconfiguration: DNS and DHCP Support,* SG24-5147.

8.9 Setting up a simple DNS server

This section describes the tasks that you must perform to configure an internal AS/400 DNS to handle a domain and a single mail server. If you have a simple network with only a few servers, you should consider using this procedure. If you have a larger number of servers, and have a host table configured on the system, you may want to use the procedure found in 8.10, “Migrating from host table name entries to the AS/400 DNS” on page 310. If the DNS is not already installed, refer to 8.7.2, “DNS installation” on page 297.

8.9.1 Task summary

To configure the AS/400 DNS for this scenario, perform the following steps:

1. Configure the AS/400 DNS to handle the internal domain.
2. Add a host name to the domain.
3. Configure the MX record for the domain.
4. Configure the internal DNS to forward the queries to a firewall (optional).

8.9.2 Configuring the AS/400 DNS to handle the internal domain

To configure the AS/400 DNS, use Operations Navigator. It is included as part of Client Access Express for Windows.

To access the DNS configuration, select your AS/400 system name. Then, select **Network->Server->TCP/IP**. Double-click **DNS**. Click the **+** symbol beside the DNS Server - Home400 (system name) entry. The window shown in Figure 391 is displayed. If this is the first time you access DNS, you may see the DNS configuration wizard.

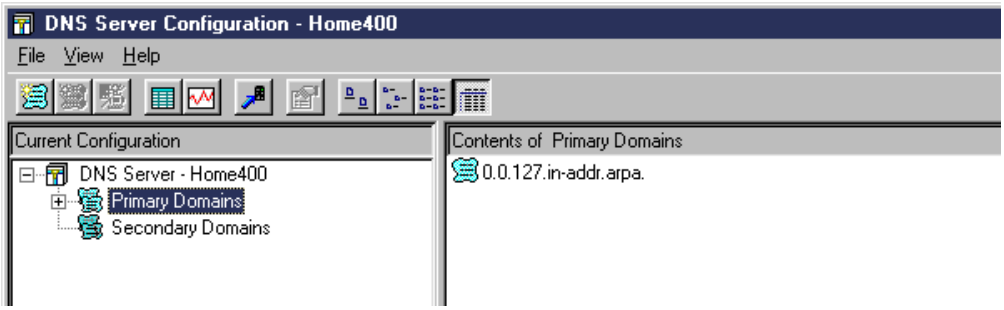


Figure 391. Configuring the AS/400 DNS to handle the internal domain domain.com

To add a primary domain, perform the following tasks.

1. Right-click on **Primary Domains**. Select **New Primary Domain**. The window shown in Figure 392 is displayed.

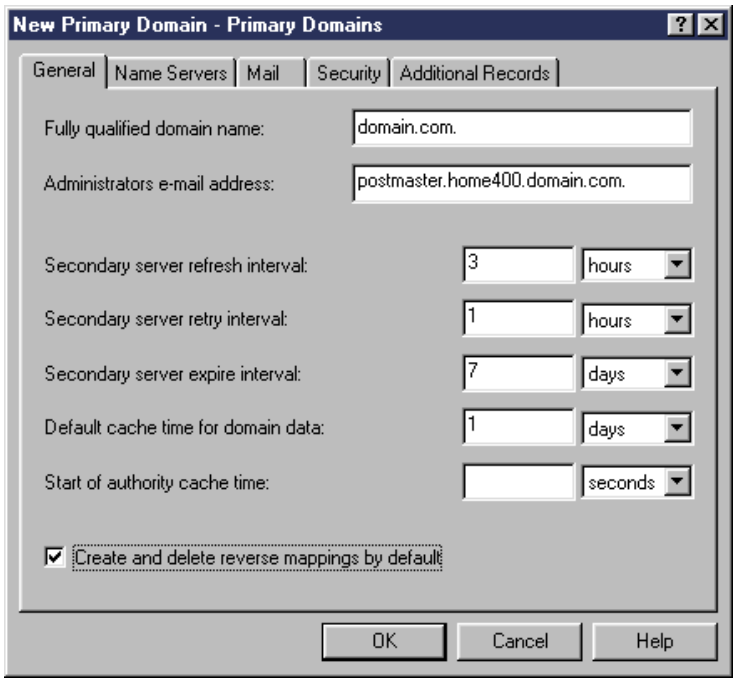


Figure 392. New Primary Domain domain.com

2. Enter the domain name `domain.com`. You *must* put a dot at the end of your domain because it is a fully qualified domain name.
3. Select **Create and delete reverse mappings by default**.
4. Click **OK**. The window shown in Figure 393 is displayed. Your domain name is displayed in the right-hand frame.
5. Right-click on the domain name you added. A drop-down menu appears. Click **Enable**. This enables the domain in the DNS.

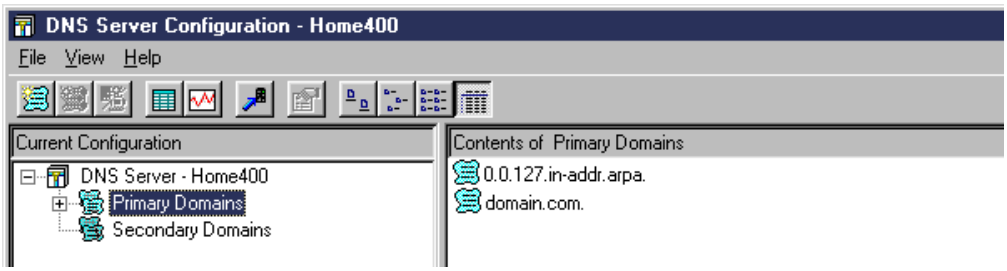


Figure 393. Contents of Primary Domains after creating domain.com

You have now successfully created your domain.

8.9.3 Adding host names to the domain

After creating the domain, you need to add the host name to each domain. Start from the window shown in Figure 393. To add the systems, perform the following steps:

1. Right-click **domain.com**.
2. Select **New Host**.

3. Click **Add**. The New Host window is displayed (Figure 394).

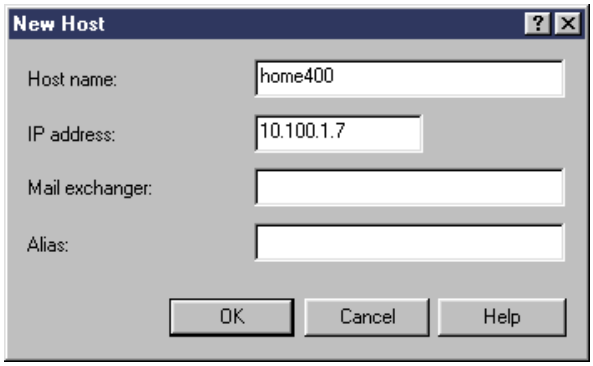


Figure 394. Adding the AS/400 host name

4. Enter the AS/400 host name and the IP address.
5. Click **OK**.

Repeat the steps in this section to add each host name of *domain.com* that you require for your environment. Only the host names that are on your internal network need to be stored in the DNS.

Now you need to add the mail exchange (MX) information for the mail domain, if you have one.

8.9.4 Configuring the MX record for your domain

The MX record tells the DNS client (it can be either a PC or another DNS) the name of the SMTP server that processes mail for the domain. Start from the window shown in Figure 393 on page 307. To add the MX record, perform the following steps:

1. Right-click **domain.com**.
2. Select **Properties**.
3. Click the **Mail** tab.
4. Click **Add**. The window shown in Figure 395 is displayed.

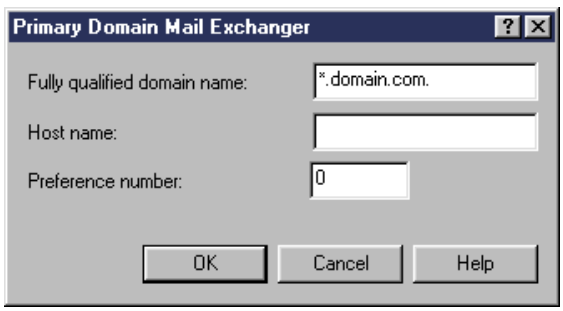


Figure 395. Adding an MX record in a domain

5. Remove the asterisk (*) from the front of the default domain name (`*.domain.com.`) to change it to `domain.com.` Change the preference number to `10`.

6. Enter the host name of the SMTP server. In a small network, this is typically the name of the AS/400 system.
7. Click **OK**.
8. Click **OK** a second time to exit the Properties window.

You have now successfully created the MX record for *domain.com*.

8.9.5 Configuring the internal DNS to forward the queries to the firewall

The internal DNS cannot answer the queries that are intended for the Internet. It needs to be linked with the DNS of your firewall. If you do not have a firewall, and are not connected to the Internet, skip this section.

If e-mail is sent to somebody@us.ibm.com, it first goes to the internal SMTP server. Then, it is forwarded to the firewall. From the firewall, it is sent to the Internet.

To set up DNS forwarding, you must change the DNS properties. You should start at the DNS Server Configuration window shown in Figure 396.

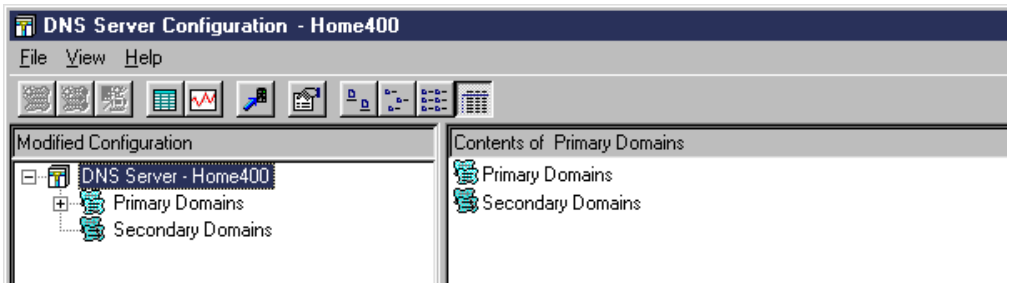


Figure 396. Configuring the internal DNS to forward queries to the firewall

Use the following procedure to change the properties of the DNS:

1. Right-click **DNS Server - Home400**.
2. Select **Properties**.
3. Click the **Forwarders** tab. The window shown in Figure 397 on page 310 is displayed.

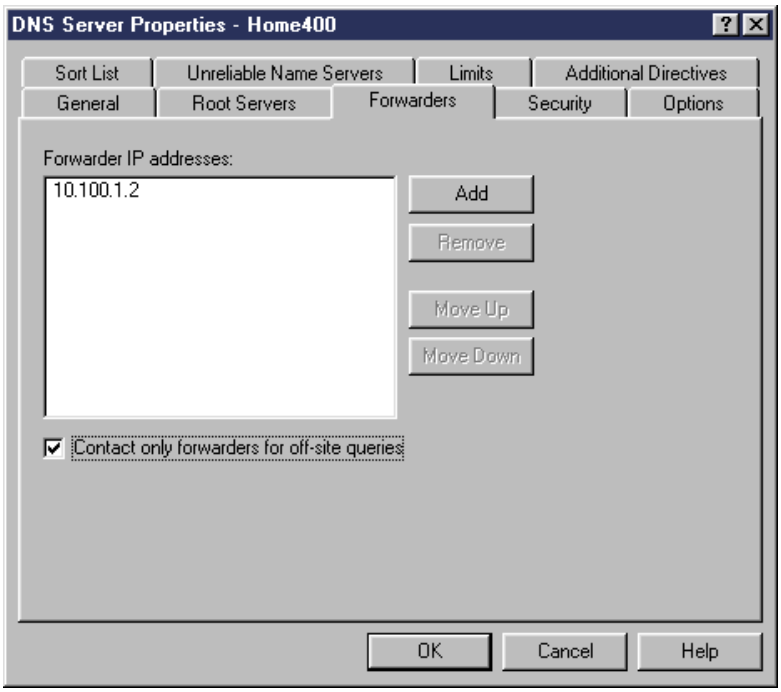


Figure 397. Adding the IP address of the firewall to the forwarders list

4. Click **Add**.
5. Enter the secure IP address of the firewall.
6. Check **Contact only forwarders for off-site queries**.
7. Click **OK**.

The DNS configuration is now ready to handle your SMTP mail. Stop and start the DNS server, or click **File->Update Server** to update the DNS server configuration and make your configuration available.

8.10 Migrating from host table name entries to the AS/400 DNS

The existing host name table entries on the AS/400 system can be migrated to files that can be maintained by the Operations Navigator DNS configuration. This migration is a two-step process:

1. The program QDNS/QTOBH2N must convert the AS/400 host table entries that you specify to DNS formatted files.
2. You must convert each of the DNS formatted files created by the QDNS/QTOBH2N program to a format compatible with the Operations Navigator DNS configuration. The Operations Navigator Import Domain Data function performs this conversion process.

You can refer to the Web site at: `http://www.as400.ibm.com/infocenter/`

At the site, search for `DNS program` for more information related to the host table migration program.

8.11 DNS server backup and recovery

You should plan to back up the DNS server configuration files on a regular basis or every time the DNS server configuration is updated by the DNS administrator.

You can use the SAV command to back up the DNS configuration files in the /QIBM/UserData/OS400/DNS IFS directory. The files in this directory are customer created DNS configuration files. These files must be backed up frequently as part of your regular backup plan. These files include:

- **BOOT**
- **Primary domain files** (both forward and reverse mapping): Be sure to include the 0.0.127.in-addr.arpa.DB reverse mapping file created by the wizard
- **CACHE** (list of root servers)

The files in the /QIBM/UserData/OS400/DNS IFS subdirectory that should not be backed up and restored are DUMPDB, STATISTICS, RUNDBG, QUERYLOG, and any files in the TMP subdirectory. These files should be deleted when you no longer want them or they are too large. Backing up and restoring PID is probably of no use either unless the SAME server job is running before and after restore.

Note

When the Operations Navigator DNS server configuration creates a file in the /QIBM/UserData/OS400/DNS directory, the file is created with the Owner value set to the AS/400 user profile that was used to start the Client Access connection with the AS/400 system. When this user profile is deleted with the parameter Owned object value *DLT, the objects which are owned by the user profile are deleted as well. In this case, any IFS DNS configuration files owned by this user profile are also deleted.

8.12 Implementing primary DNS servers

In this section, we show how to get started with the implementation of a DNS server on your AS/400 system. We discuss step-by-step how to migrate from your existing name resolution process based on the AS/400 host table to a full implementation of a primary name server.

Many companies have a simple internal network consisting of one or two subnets and use AS/400 host tables and PC client host tables to resolve TCP/IP host names to IP addresses. The disadvantage of this name resolution method is that every addition of a host may require an update to every client that needs to contact this new host. Configuring one AS/400 system to be a primary DNS server solves this problem because adding or deleting a host and its IP address is done only once on the primary name server.

8.12.1 Scenario overview

In this scenario, we use a network that consists of three subnets connected by routers as shown in Figure 398 on page 312. This network is not connected to the Internet and does not have a firewall installed anywhere within the network. This

network initially does not include a DNS server and relies on host tables to resolve host names to IP addresses.

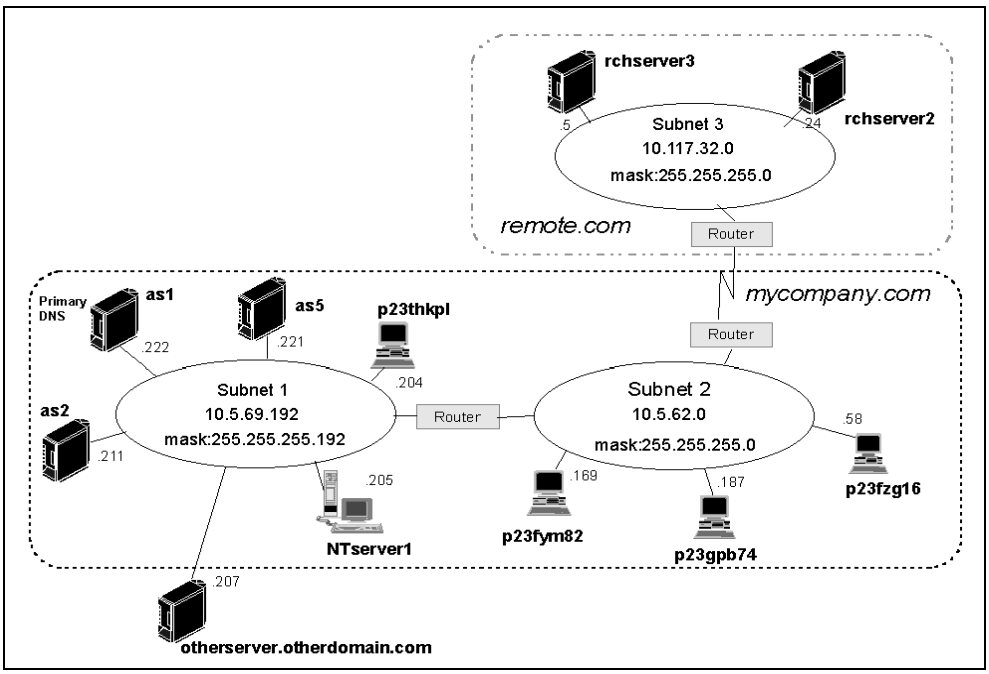


Figure 398. Network layout used in the scenario

The network mycompany.com is an internal network, and it is not connected to the Internet. The three subnet network IDs and subnet masks are:

- 10.5.69.192 with subnet mask 255.255.255.192
- 10.5.62.0 with subnet mask 255.255.255.0
- 10.117.32.0 with subnet mask 255.255.255.0

The primary DNS will run on AS1. This AS/400 system is configured to be primary for the mycompany.com domain. Because the mycompany.com domain includes the subnets 10.5.69.192 and the 10.5.62.0, AS1 is also configured with the primary domain files 69.5.10.in-addr.arpa and 62.5.10.in-addr.arpa (as explained for the reverse mapping files described in 8.7.4, “DNS configuration files” on page 299).

We will plan the primary domain and configure a primary DNS server on AS1. We will migrate the AS1’s host table to create the initial DNS configuration on AS1.

8.12.2 Scenario advantages and disadvantages

This scenario offers the following advantages:

- It assumes the customer is coming from an environment that does not have a name server in the internal network. Therefore, this scenario makes a good starting place for customers with little or no experience with name servers.
- It discusses how an AS/400 host table can be migrated into the DNS server configuration, which can make the initial name server configuration go faster and smoother.

This scenario has the following disadvantages:

- It describes how to configure a primary name server in a small internal network that does not have access to the Internet and does not have a firewall installed in the network. This type of network and name server configuration do not meet the needs of network installations that require Internet access and a firewall.
- It describes how to configure DNS servers for one domain; mycompany.com. Thus, all hosts included in this name server configuration must have the domain name of mycompany.com.
- It does not describe how to handle subdomains in the mycompany.com domain.

8.12.3 Task summary

We assume that the TCP configuration on the AS/400 systems in the network and all other hosts in the network is completed, and TCP/IP connectivity has been verified.

The tasks for this scenario include:

1. Planning the primary domain mycompany.com
2. Creating the DNS primary name server on AS/400 system AS1 using the following substeps:
   a. Preparation of the local host table for migration on AS/400 system AS1
   b. Migration of the AS1 host table into DNS formatted files (an AS/400 program is used to do this)
   c. Using Operations Navigator Import domain data to migrate the DNS formatted files to files that can be maintained by the Operations Navigator DNS configuration
   d. Using Operations Navigator DNS server configuration to make final and ongoing configuration changes, if necessary
3. Starting the DNS server on AS/400 system AS1
4. Verifying that the DNS is operational
5. Reconfiguring the clients to use a DNS server instead of using host tables

8.12.3.1 Primary domain planning

When moving from host tables to a name server, we should determine which domain will become the primary domain. In this scenario, mycompany.com is the primary domain on the AS1 name server. The dotted line box in Figure 398 indicates which hosts belong to the mycompany.com domain. All the hosts but one on the 10.5.69.192 network are included in this domain. The host OTHERSERVER, although it is on the 10.69.192 network, is in the domain OTHERDOMAIN.com domain. It is not part of the mycompany.com domain. All hosts on the 10.5.62.0 network are included in this domain, but no host on the 10.117.32 network is included in the mycompany.com domain because the hosts on this network are part of remote.com domain. In other words, the hosts located in the remote.com domain are excluded in the migration. Consequently, the AS1 name server is unaware of the remote.com and its hosts. It is assumed that remote.com will continue to use host tables to resolve hostnames to IP addresses.

The 10.117.32.0 network is not included in the migration, but both networks 10.5.69.192 and 10.5.62.0 are included. The specific host OTHERSERVER is excluded from the migration because it is part of another domain of OTHERDOMAIN.com, even though it is part of the 10.5.69.192 network, as indicated in the network diagram in Figure 398 on page 312, and in the host table in Figure 399 on page 315.

8.12.3.2 Creation of the primary name server on the AS/400 system

We now discuss all the different steps that need to be followed to configure AS1 as a primary name server.

Preparing the host table for migration

It is possible to use the Operations Navigator interface to configure DNS from the beginning. But since the AS1 host table contains the host names and IP addresses of the hosts to be included in the mycompany.com domain, the host table migration method will be much easier to migrate into a DNS configuration.

Consider cleaning up the AS1 host table because it is the starting point for the migration. Perform these steps:

1. Delete any hosts from the table that no longer exist in the network.
2. Make sure all hosts in the mycompany.com domain are listed in the AS1 host table.
3. Check for incorrect IP addresses and typing mistakes in the AS1 host table names.
4. Verify that the hosts listed in the client's host tables are included in the AS1 host table.
5. Check for all host names in the host table with domain names other than mycompany.com.

   Do they belong to another domain or should they be included in the mycompany.com domain? If they belong in the mycompany.com domain, now is the time to change the domain name on the host itself to mycompany.com and update the AS1 host table to reflect this change. However, when changing the domain name of a host, be aware of the impact the change can have on the clients that possibly use this host as a server. If the host you are changing the domain name of is a mail server, the domain name change can have a wide-spread effect.

You should also plan for the hosts that are not included in the migration. Figure 398 on page 312 specifies three hosts in the network that are not included in the mycompany.com domain. The DNS server we want to implement on AS1, with one primary domain of mycompany.com, will not resolve queries for host OTHERSERVER, nor will it resolve queries for Rchserver2 and Rchserver3. If the AS1 system is the only host that needs to access these systems, leaving their host names/IP addresses in the AS1 host table may be sufficient, since an AS/400 system can be configured to check its local host table first. If the answer is not in the table, query the DNS server. But, if other clients need access to OTHERSERVER, Rchserver2, or Rchserver3, you need to decide how the clients will resolve those hosts names.

Consider host OTHERSERVER in the domain OTHERDOMAIN.com. It is a good idea to review the AS1 host table and determine if this host really needs to belong

in a domain of OTHERDOMAIN.com or if it can belong in the mycompany.com domain. If it can be included in the mycompany.com domain, now is a good time to change its domain name and change the AS1 host table to also reflect this change so OTHERSERVER can be included in the migration as well. For purposes of illustrating the example of excluding a host from the migration, consider OTHERSERVER as part of OTHERDOMAIN.com, and the migration will exclude this host.

Consider the situation with hosts Rchserver2 and Rchserver3. The AS1 host table shown in Figure 399 indicates these two hosts are part of the remote.com domain. If the DNS server running on AS1 really needs to resolve DNS queries for these hosts, a second primary domain of remote.com on AS1 DNS server can be created and configured through the Operations Navigator interface. In this particular case, AS1 has a DNS server running on it and is responsible for two primary domains: mycompany.com and remote.com. But we will now only concentrate on creating the primary domain of mycompany.com on AS1. The remote.com domain is excluded over here. You should be aware that it is possible to create additional primary domains and secondary domains if the domain naming scheme and the network require it.

```
                    Work with TCP/IP Host Table Entries
                                                                 System:   AS1
Type options, press Enter.
  1=Add   2=Change   4=Remove   5=Display   7=Rename

     Internet         Host
Opt  Address          Name

     10.5.62.58       p23fzg16
                      p23fzg16.mycompany.com
     10.5.62.169      p23fym82
                      p23fym82.mycompany.com
     10.5.62.187      p23gpb74
                      p23gpb74.mycompany.com
     10.5.69.204      p23thkp1
                      p23thkp1.mycompany.com
     10.5.69.205      NTserver1
                      NTserver1.mycompany.com
     10.5.69.207      otherserver
                      otherserver.otherdomain.com
     10.5.69.211      as2
                      as2.mycompany.com
    10.5.69.221       as5
                      as5.mycompany.com
    10.5.69.222       as1
                      as1.mycompany.com
    10.117.32.5       Rchserver3
                      Rchserver3.Remote.com
    10.117.33.24      Rchserver2
                      Rchserver2.Remote.com
    127.0.0.1         LOOPBACK
                      LOCALHOST
```

Figure 399. AS1 host table

Migration of the AS/400 host table to DNS formatted files

QTOBH2N is the AS/400 program that is used to migrate the AS/400 host table to DNS formatted files. Several options can be used with this program.

A complete list of options is described in the Information Center at:
`http://www.as400.ibm.com/infocenter/`

Search for `DNS program`. You will find an article that provides more information related to the host table migration program. We now only cover the options that we used to migrate the AS1 host table in Figure 399 on page 315.

This migration step is one of the few DNS configuration steps that can still be executed from an AS/400 green screen. The following steps migrate the AS1 host table to DNS formatted files:

1. Make sure the AS1 host table is cleaned up and accurate.
2. Add library QDNS to the user's library list with the AS/400 command:

   ```
   addlibl libe(QDNS)
   ```

3. Grant the user profile that will run the program QDNS/QTOBH2N *ALLOBJ special authority.
4. Change the job Coded Character Set ID (CCSID) for the user job that will run the program QDNS/QTOBH2N to 37. Be sure to record the original job coded character set ID so that you can change it back. Change the CCSID just before you run the program QDNS/QTOBH2N. Change the CCSID back immediately after you run this program.

   Note

   Changing the CCSID back may not be a simple task because of the interaction of DFTCCSID and CCSID when the CCSID is set to 65535. It may be better to run the host table migration program from a batch job. Attempting to change back the CCSID may leave the keyboard in an unusable state in some countries (for example, Japan).

   To change the CCSID of the user's job, you should perform the following steps:

   a. Enter the AS/400 `CHGJOB` command.
   b. Press F4 to prompt.
   c. Press F10 to select additional parameters.
   d. Press the Page Down key twice to the parameter Coded Character Set ID.
   e. Record the current value for Coded Character Set ID.
   f. Change the coded character set ID to `37`.
   g. Press Enter.

The program QTOBH2N migrates the AS/400 host table into DNS formatted files. On the AS1 AS/400 system, you should issue the following command:

```
call pgm(qdns/qtobh2n) parm('-d' 'mycompany.com' '-n' '10.5.62:255.255.255.0'
'-n' '10.5.69:255.255.255.0' '-e' 'otherdomain.com' '-M')
```

In our example, the following three files are created by the preceding program:

- h2n.mycompany
- h2n.10.5.62
- h2n.10.5.69

After the command completes, the job log contains the message DNS0417:

```
Process completed successfully. DNS formatted files prefixed by h2n built
in directory /QIBM/USERDATA/OS400/DNS.
```

The text of your message may be different from the example shown, depending on your release.

Note

At this point, it is important to remember to change the CCSID on the user's job back to what it was before it was changed to 37. Set the DFTCCSID first and then the CCSID. Both should be set to the same value.

The options used to run the QTOBH2N program specify the way the host table should be migrated. An explanation of the options used follows here:

- The `-d mycompany.com` option indicates the domain that the name server is primary for is mycompany.com.
- The `-n 10.5.62:255.255.255.0` and `-n 10.5.69.69.255.255.255.0` indicate that hosts listed in the AS/400 system's host table with IP addresses included in the networks 10.5.62 and the 10.5.69 are included in the migration.
- Any hosts in the preceding two networks that are in the domain OTHERDOMAIN.com are not included in the migration.
- The migration does not create any MX records because the -M option was used. If the -M option is not used, an MX record is created for *every* host included in the migration.

  Note 1: The -e option needs further explaining. Remember, every host that is included in the migration is included in the mycompany.com domain. If the OTHERSERVER host is not excluded with the -e option, then OTHERSERVER is migrated with a domain of OTHERDOMAIN.com.mycompany.com. Even if OTHERDOMAIN is a subdomain of mycompany.com, the absolute domain name of OTHERDOMAIN.com.mycompany.com is not correct. Making OTHERDOMAIN a subdomain of mycompany.com is discussed in Chapter 5.

  Note 2: Rchserver2 and Rchserver3 are not included in the migration by default, and it is not necessary to eliminate them explicitly with the -e option. This is because only the hosts residing in the networks specified with the -n options are included in the migration. Because Rchserver2 and Rchserver3 reside on the 10.117.32.0 network, they are not included in the migration.

  Note 3: Hosts AS1, NTserver1, AS2, AS5, p23thkp1,and OTHERSERVER have subnet masks of 255.255.255.192. These hosts are in the 10.5.69.192 network. The migration program does not handle subnetting into the fourth octet. Therefore, if the AS1 host table included hosts in the 10.5.69.0 (the 10.5.69.64 or the 10.5.69.128 networks), the migration program includes these hosts regardless of whether we want to include them in the migration.

  In this scenario, the migration program creates three files in the /QIBM/UserData/OS400/DNS directory:

  ```
  h2n.mycompany
  h2n.10.5.62
  h2n.10.5.69
  ```

You can verify that these files are in the /QIBM/UserData/OS400/DNS directory. You can use the AS/400 command:

```
wrklnk '/QIBM/UserData/OS400/DNS'
```

You can then enter option `5` to view the next level of the DNS directory.

```
                          Work with Object Links

Directory  . . . . :   /QIBM/UserData/OS400/DNS

Type options, press Enter.
  3=Copy   4=Remove   5=Next level   7=Rename   8=Display attributes
  11=Change current directory ...

Opt   Object link            Type     Attribute    Text
      h2n.mycompany          STMF
      h2n.10.5.62            STMF
      h2n.10.5.69            STMF
      ATTRIBUTES             STMF
      TMP                    DIR
```

Figure 400. Work with Object Links

The three h2n files were created by the QTOBH2N program. The ATTRIBUTES file and the TMP directory existed before the QTOBH2N program has been run. They were automatically created when you installed the OS/400 Domain Name System option 31.

To view the contents of h2n.mycompany, you can use the Display File (`DSPF`) or the Edit File (`EDTF`) command on the AS/400 command line, or you may use Operations Navigator. The DSPF and EDTF commands are included in V4R4. For other releases, IBM provides tools for working with stream files directly from the AS/400 command line as well. The tools are provided in a PTF (SF38832 V3R7, SF41518 V4R1, SF45296 V4R2, SF49052 V4R3). You can also download the tools from the Internet at: `http://service.software.ibm.com/dl/sap/saptools-d`

The current list of tools (provided “as is”) in the package includes:

DSPSTMF	Display Stream File
EDTF	Edit File
FINDBNDSP	Find Bound Service Program
FINDMODS	Find Modules
MODEXPORTS	List Module Exports
RCLSPACE	Reclaim Space
SQLUTIL	SQL Utility
SAVTOSTMF	Save To Stream File
RSTFRMSTMF	Restore From Stream File
CPYFRMSAVF	Copy From Save File
CPYTOSAVF	Copy To Save File

You may find EDTF to be the most useful tool in the package. After you install the tools, you can test the EDTF command. Type `EDTF` and press F4. The command prompt shown on Figure 401 appears.

```
                    Edit Files (EDTF)

Type choices, press Enter.

Stream file to edit: . . . . . .   '*DBFILE'

DataBase file to edit: . . . . .                Name
  Library: . . . . . . . . . . .     *LIBL      Name, *LIBL, *CURLIB
Member to be edited: . . . . . .   *FIRST       Name, *FIRST
```

Figure 401. EDTF command prompt

The EDTF command parameters are shown in Table 12.

Table 12. EDTF command parameters

Parameter	Description
Stream file to edit	The full path and name of the stream file you wish to edit. *DBFILE lets us edit database file whose name is written in the database file to edit parameter.
Member to be edited	The member name we wish to edit.

Press Enter. You will see a screen-line editor, which is similar to an SEU as shown on Figure 402. The editor automatically detects the file type (ASCII or EBCDIC). Press F1 to display a help screen with the available line commands. To activate the command line, enter the command to the left of the requested line. For example, enter `D` to the left of a specific line to delete that line.

On the editor screen, press F3 to exit and save the edited file.

```
 Edit File: h2n.mycompany
 Record . :       1 of       9 by   8           Column:    1 of  111 by  74
 Control  :

CMD ....+....1....+....2....+....3....+....4....+....5....+....6....+....7....+
     ************Beginning of data**************
    mycompany.com. IN  SOA AS1.mycompany.com. POSTMASTER.AS1.mycompan.com  ( 1 1
      IN  NS   AS1.mycompany.com.

    localhost            IN  A     127.0.0.1
    p23fym82             IN  A     10.5.62.169
    p23gpb74             IN  A     10.5.62.187
    p23fzg16             IN  A     10.5.62.58
    p23thkp1             IN  A     10.5.69.204
    p23thkp1             IN  A     10.5.69.204
    ntserver1            IN  A     10.5.69.205
    as2                  IN  A     10.5.69.211
    as5                  IN  A     10.5.69.221
    as1                  IN  A     10.5.69.222
     ************End of Data*********************
 F2=Save   F3=Save/Exit   F12=Exit   F15=Services   F16=Repeat find
 F17=Repeat change  F19=Left  F20=Right
            (C) COPYRIGHT IBM CORP. 1980,  1999.
```

Figure 402. EDTF editing display

To view the contents of h2n.mycompany, use Operations Navigator to perform the following steps:

1. Click the + sign next to **as1.mycompany.com**.
2. Click the + sign next to **File System**.
3. Click the + sign next to **root**.
4. Click the + sign next to **QIBM**.
5. Click the + sign next to **UserData**.
6. Click the + sign next to **OS400**.
7. Click the + sign next to **DNS**.

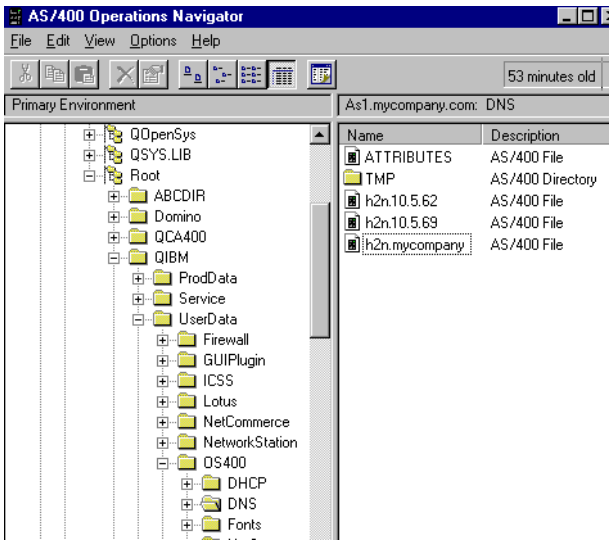


Figure 403. DNS directory contents through Operations Navigator

Figure 403 shows information similar to the WRKLNK command. However, double-clicking on h2n.mycompany.com brings up an Open window that allows you to choose your favorite program to view the content of the DNS files. We configured this to use Netscape to browse the h2n.mycompany file shown in Figure 404.

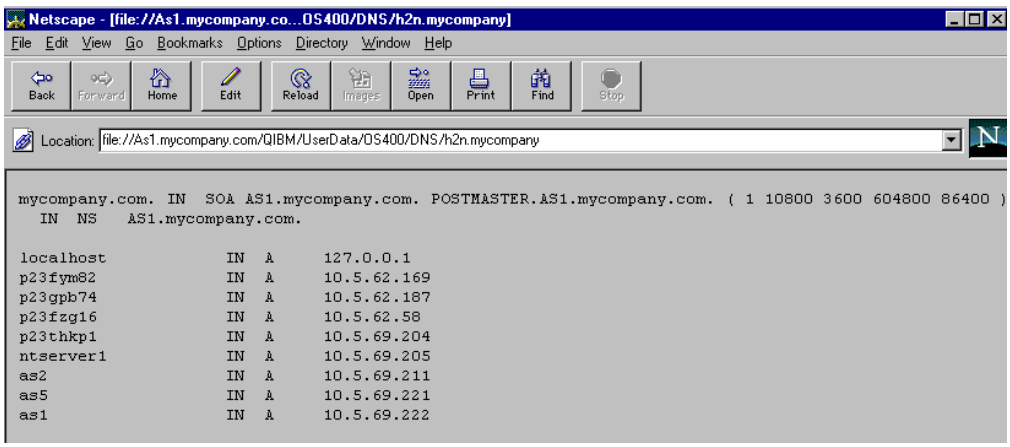


Figure 404. Using Netscape to view the contents of the h2n.mycompany file

Figure 405 and Figure 406 show the contents of h2n.10.5.69 and h2n.10.5.62 browsed with Netscape.

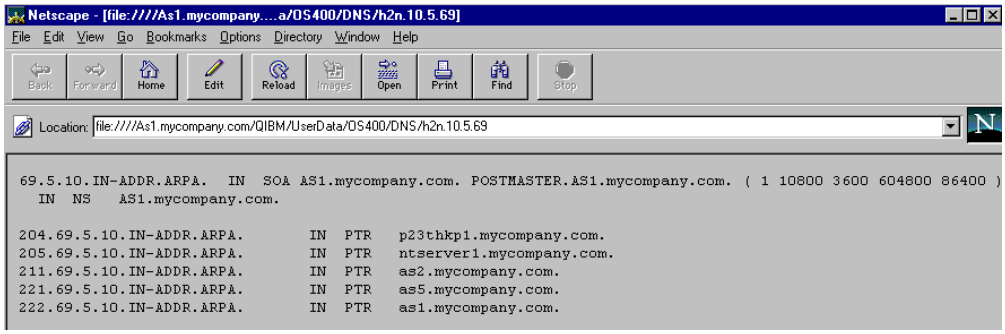


Figure 405. Using Netscape to view the h2n.10.5.69 file

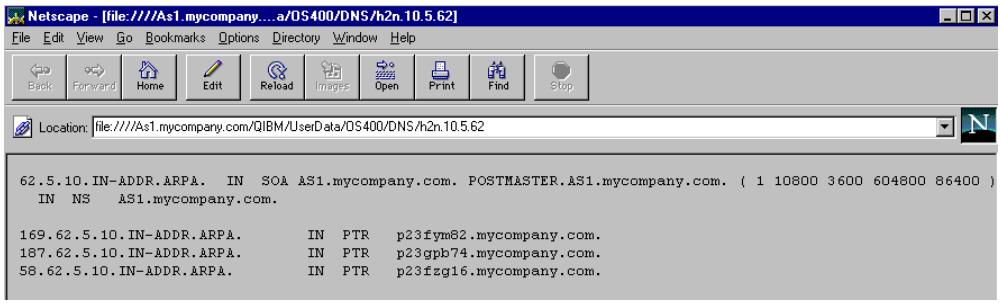


Figure 406. Using Netscape to view the h2n.10.5.62 file

Importing the DNS formatted files to Operations Navigator

The AS/400 host table has now been migrated to DNS formatted files using the QTOBH2N program. It is now time to migrate the DNS formatted files to Operations Navigator DNS files. We will build the DNS as a Cache-only server. We use Operations Navigator DNS Configuration Import Domain function to accomplish this step. When the import takes place, the server is changed to a primary DNS server.

From a Client Access Windows 95 client, you should bring up Operations Navigator, and follow these instructions:

1. Select your AS/400 system name **AS1.mycompany.com**.
2. Click the + sign next to **AS1.mycompany.com**.
3. Click the + sign next to **Network**.
4. Click the + sign next to **Servers**.
5. Click the + sign next to **TCP/IP**.

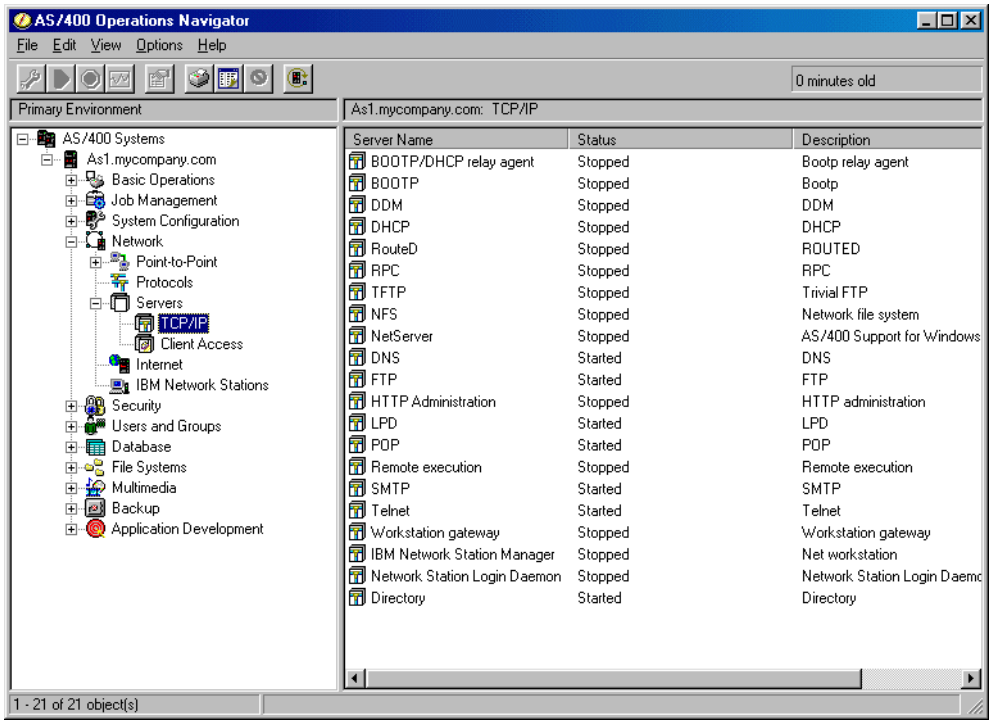


Figure 407. DNS server in Operations Navigator

Double-clicking **DNS** brings up the DNS server configuration wizard. The wizard automatically starts when you enter the DNS configuration for the first time (Figure 408).

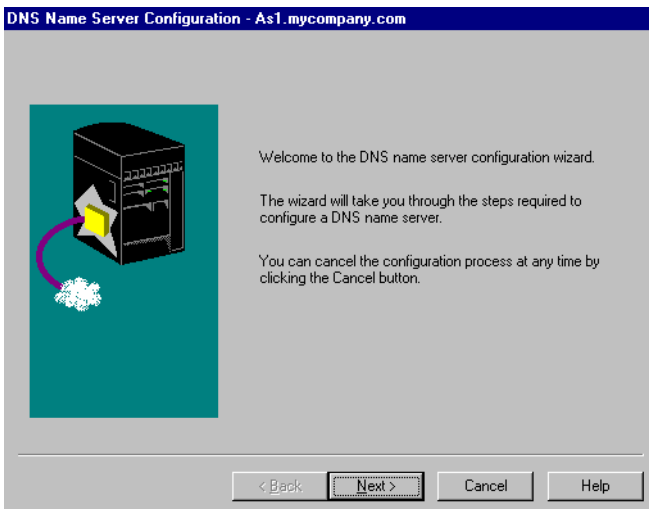


Figure 408. DNS configuration wizard in Operations Navigator

6. Click **Next**.

   The next wizard window allows you to add IP addresses for root servers. We do not use root servers. Click **Next** to bypass the root server window. The display shown in Figure 409 appears.

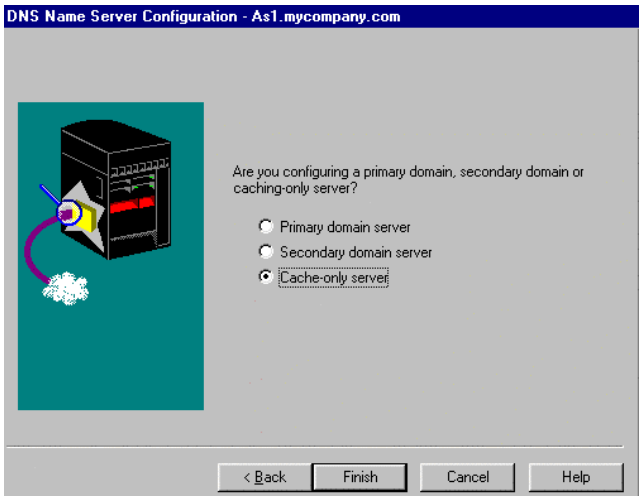


Figure 409. Primary domain type within the DNS server configuration wizard

AS1 will be the primary domain server for the domain mycompany.com but, for now, we want to set up the server as cache-only. Select **cache-only** as shown in Figure 409. When you select Cache-only server, the Next button changes to a Finish button. Click **Finish**. The display shown in Figure 410 appears.

We now run the Import Domain function. It creates the domain files and changes the server into a primary DNS server. To import the data, follow these steps:

1. Right-click **Primary Domains**. A pop-up window appears as shown in Figure 410.

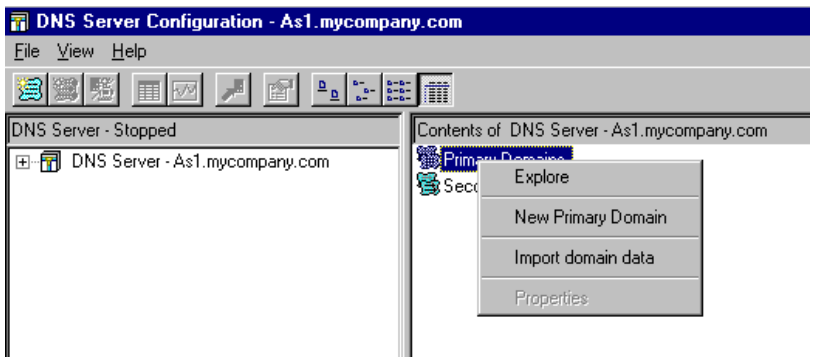


Figure 410. Right-click Primary Domain

2. Select **Import domain data**.
3. A window is shown containing a default path of /QIBM/UserData/OS400/DNS. Add the file you want to import to the path. In this case, the first file to be imported is h2n.mycompany. See Figure 411 on page 324. Click **OK**.

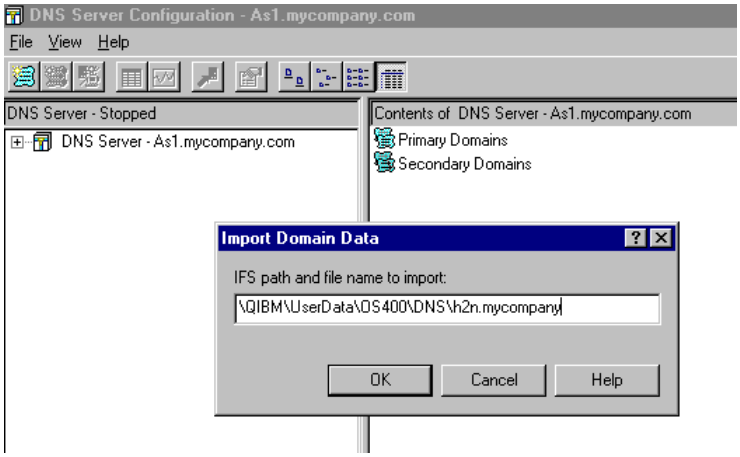


Figure 411. Import Domain Data function

A new file, mycompany.com, is now created under Primary Domains.

You should repeat the Import Domain Data function for every h2n file that the QTOBH2N program created earlier. Repeat the Import Domain Data steps two more times for the remaining two h2n migration files: h2n.10.5.62 and h2n.10.5.69.

When you double-click Primary Domains in the Operations Navigator DNS server configuration now, four files are displayed as shown in Figure 412.

Note

In some releases, if the import file name entered does not exist, an error message is *not* sent to the user.

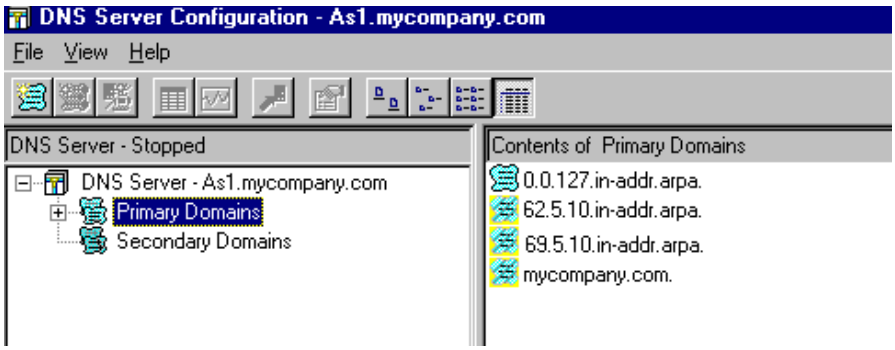


Figure 412. Import Domain Data function results

The four files contained in the AS1 primary domain are the files that the DNS server requires to answer queries for the mycompany.com domain with the exception of a query for a mail server.

Note that the three files, 62.5.10.in-addr.arpa, 69.5.10.in-addr.arpa, and mycompany.com, shown in Figure 412, have an icon to the left of the file names that appears to be “hashed” (some releases have a yellow exclamation mark (**!**)). This indicates that the domain is currently disabled.

The DNS server does not load a disabled domain. A disabled domain is like a “sandbox”. A domain can be created without making it live. To enable each domain, you should right-click on each file name and select **Enable**.

Close the Operations Navigator DNS server configuration window to save the DNS configuration.

The migration of the AS/400 host table is completed now. However, there are a few more DNS configuration changes that are best accomplished with the Operations Navigator DNS server configuration. We discuss these changes in the following sections.

Additional DNS configuration with Operations Navigator
Once the migration of the host table is complete, any additional configuration changes can be made using the Operations Navigator DNS server configuration.

Automatically creating or deleting reverse mapping entries
You should change the configuration to automatically create or delete a reverse mapping entry for every forward mapping entry that is added.

Note

A forward mapping entry is also called an *A record* or *address record*, which is contained in the forward mapping primary domain file. This entry is created by adding a new host to the mycompany.com primary domain file. Forward mapping is a host name to IP address mapping. The reason we make this configuration change can best be explained by the following example.

If a new host named newhost is added to the 10.5.69.192 network with an IP address of 10.5.69.206, the DNS administrator must add a host to the mycompany.com forward mapping primary domain file. This entry allows the DNS server to answer a query for a client who sends the IP address to the DNS server and requests that the DNS server give it the host name for the IP address. If the DNS administrator forgets to add the same new host to the 69.5.10.in-addr.arpa domain, the DNS server cannot answer a query from a client that sends the IP address of 10.5.69.206 and requests its host name. This type of query is sometimes called a “reverse look up”. Consequently, another name for the 69.5.10.in-addr.arpa file is reverse mapping file for the 10.5.69 network.

By configuring the DNS server to automatically create and delete the reverse mapping files, a DNS administrator only has to enter the new host into the forward mapping file: mycompany.com. The matching entry is automatically added in the appropriate reverse mapping file by Operations Navigator.

There are few situations where a DNS administrator wants a host entered into the forward mapping file but not entered into the reverse mapping file. Thus, we recommend this configuration change. It can save time and help prevent mistakes when manually adding new hosts to the primary domain.

To make this configuration change, complete the following steps:

1. Right-click on the file **mycompany.com**.
2. Select **Properties**.
3. Check **Create and delete reverse mappings by default** as shown in Figure 413.
4. Click **OK**.
5. Close the Operations Navigator DNS server configuration to save the configuration changes.

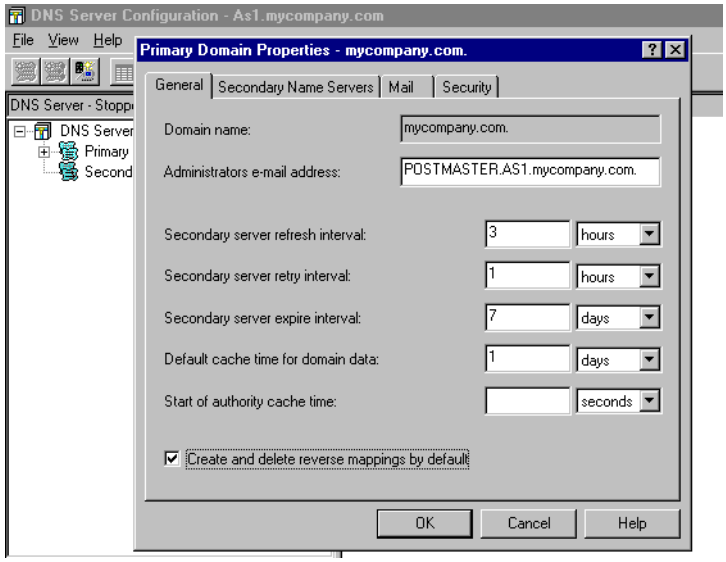


Figure 413. Enable Create and delete reverse mapping by default

Reviewing the primary domain files on the As1 name server

You should review the contents of each primary domain file on AS1 by performing these steps:

1. Double-click **DNS**.
2. Double-click **DNS Server- as1.mycompany.com**.
3. Double-click **Primary Domains**.
4. Double-click the forward mapping file **mycompany.com**.

   Figure 414 shows the contents of mycompany.com forward mapping primary domain file.

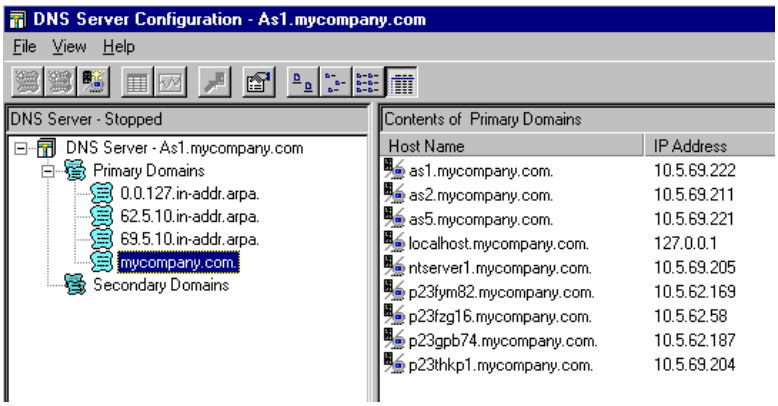


Figure 414. Contents of the mycompany.com primary domain file

5. Double-click the **62.5.10.in-addr.arpa** primary domain file to view the contents of the reverse mapping primary domain file for the 10.5.62 network shown in Figure 415 on page 328.

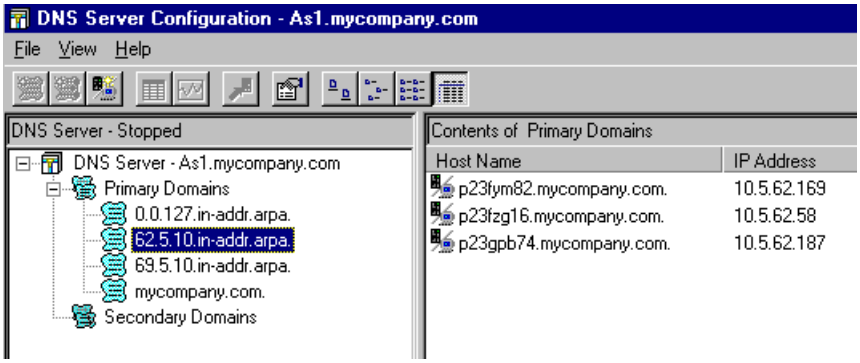


Figure 415. 62.5.10.in-addr.arpa primary domain

6. Double-click the **69.5.10.in-addr.arpa** primary domain file to view the contents of the reverse mapping file for the 10.5.69 network shown in Figure 416.

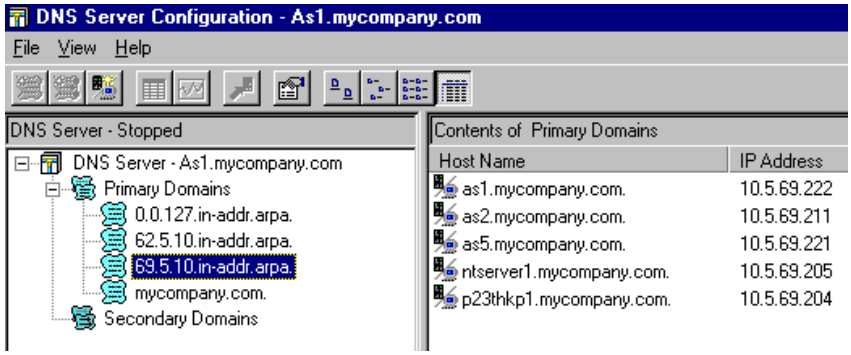


Figure 416. 69.5.10.in-addr.arpa primary domain

The 0.0.127.in-addr.arpa domain was created by the DNS configuration wizard. Figure 417 shows the contents of this primary domain file. Note that the host localhost is contained in the mycompany.com forward mapping file shown in Figure 414 on page 327. You can think of the host localhost as the host that AS1 uses to “talk to itself”. This host is a requirement. It should immediately be added with the Operations Navigator DNS configuration wizard when initially configuring the name server.

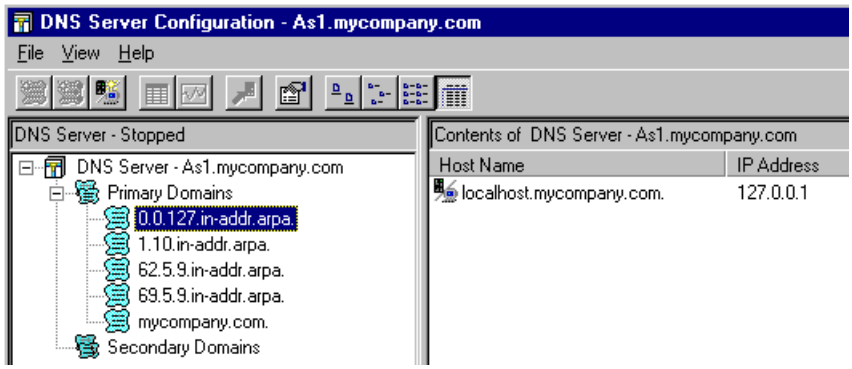


Figure 417. Loopback primary domain

We now finished the configuration of mycompany.com primary DNS server by configuring one forward mapping file (mycompany.com) and two reverse mapping

files, 62.5.10.in-addr.arpa and 69.5.10.in-addr.arpa. All three files are primary domain files. The wizard created a BOOT file, CACHE file that contains the root name servers, and the 0.0.127.in-addr.arpa reverse mapping file automatically. The directives in the BOOT file are created through Operations Navigator. Note that you cannot view the BOOT and CACHE files from Operations Navigator's DNS configuration windows. However, they are located in the IFS directory /QIBM/UserData/OS400/DNS and can be viewed with Operations Navigator:

1. Double-click the AS/400 system where the DNS server is running.
2. Click the + sign next to **File Systems**.
3. Click the + sign next to **Root->QIBM->UserData->OS400**.
4. Double-click **DNS**.
5. Double-click **BOOT or CACHE file**. Choose the program you want to use to view the file.

Later, we say that a name server "caches" information it receives from another name server. This is a way a name server "remembers" information so if it receives a query from a client for the same host, it can respond with an answer from its cache and not query the authoritative name server again. It is important to understand that this cached information is not contained in the /QIBM/UserData/OS400/DNS/CACHE file. The CACHE file contains information about root servers. This scenario does not require the use of root servers. Therefore, the CACHE file in this scenario should remain empty.

Starting the DNS server on the AS/400 system AS1

To start the DNS server on AS1, complete these steps:

1. Close the DNS window in Operations Navigator.
2. From the TCP/IP Server list, right-click **DNS**.
3. Click **Start**.

Figure 418 on page 330 shows the DNS start sequence.

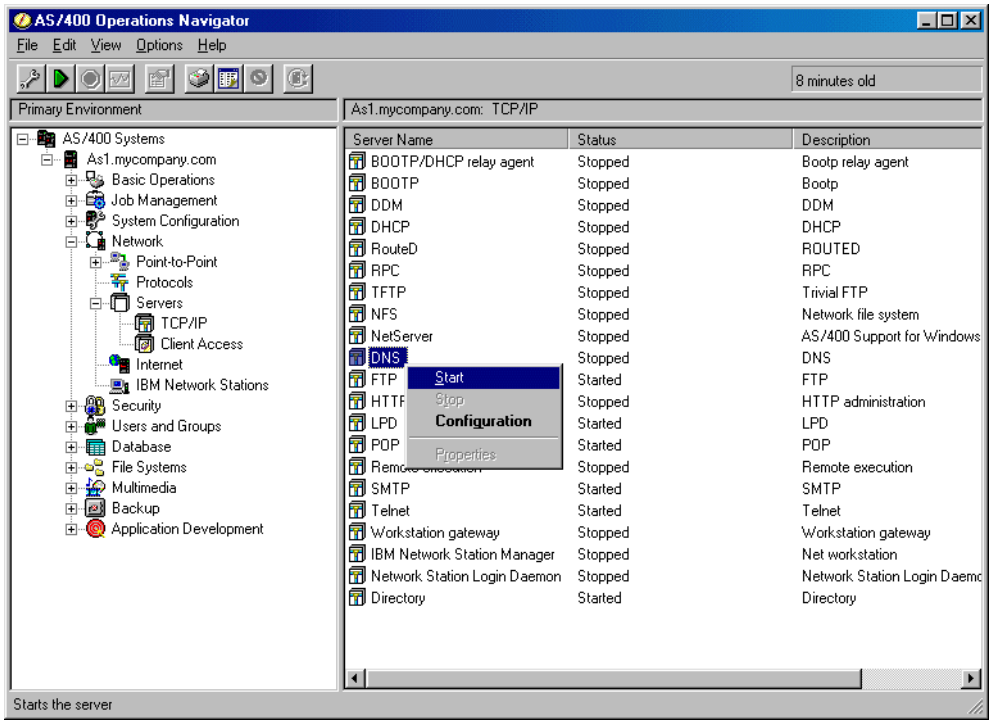


Figure 418. Start the DNS server

The DNS Server status is now started. This may take a minute. Figure 419 shows the active DNS server status. Once the DNS server is started, there should be one job named QTOBDNS active in the QSYSWRK subsystem on AS1.

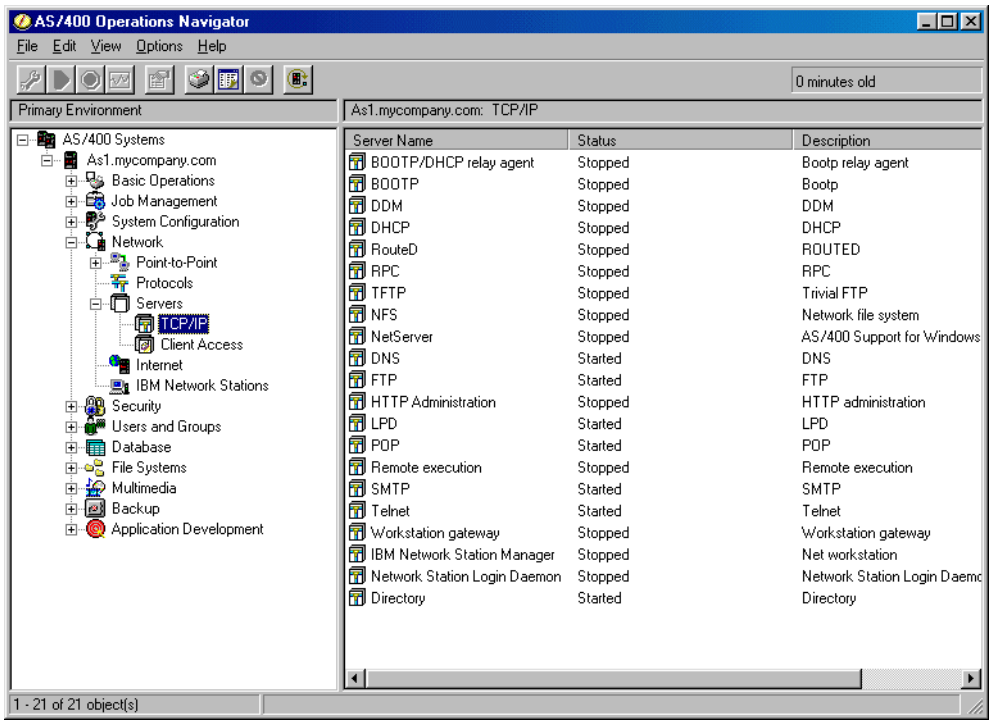


Figure 419. Started DNS server

Verifying that the DNS server is operational

You should make sure that the name server is working properly. The DNS job logs and NSLOOKUP are the best sources to check for errors and verify that the DNS is operating as expected.

Reviewing the DNS job log QTOBDNS for errors

It is always a good idea to review the QTOBDNS job log for any errors. Complete these steps:

1. From the AS1 command line, enter:

   ```
   wrkactjob sbs(qsyswrk) job(qtobdns)
   ```

2. Enter option `5` to work with the job.
3. Enter option `10` to display the job log.
4. Press F10 to display all messages in the job log. You may have to scroll up or scroll down to view all the messages.
5. Review messages for any errors.

Figure 420 shows the QTOBDNS job log after a successful startup of the DNS server.

```
Job . . :   QTOBDNS       User . . :   QTCP          Number . . . :   013973

>> CALL PGM(QDNS/QTOBDNS) PARM('-p' '53' '-d' '0' '-b' '/QIBM/UserData/OS400/
   DNS/BOOT')
   DNS server starting.
   Could not assign address to socket.
   primary zone mycompany.com (serial number 886456347) loaded successfully.
   primary zone 69.5.10.in-addr.arpa (serial number 886456347) loaded
     successfully.
      primary zone 1.1.10.in-addr.arpa (serial number 886456347) loaded
     successfully.
   primary zone 62.5.10.in-addr.arpa (serial number 886456347) loaded
     successfully.
   cache zone . (serial number 0) loaded successfully.
   Ready to answer queries
```

Figure 420. QTODBNS job log

Note

If an IP interface is started after the DNS server starts, the DNS server must be stopped and started again or the Update Server function from Operations Navigator must be run before the name server can accept queries on the newly started IP interface.

Restricting queries by the client's IP address

It is possible to configure each primary domain file to allow clients with only certain IP addresses to query this primary domain data. From Operations Navigator, go into the primary server AS1's DNS configuration. To authorize only certain clients to query AS1, use the following steps:

1. Start AS1 DNS server configuration in Operations Navigator.
2. Double-click **Primary Domains**.

3. Right-click **mycompany.com**.
4. Select **Properties**.
5. Select the **Security** tab.

   Note: When we display the Security tab, the Limit domain data access to subnets list and the Limit domain data access to IP address list are both blank by default. When both of these lists are blank, that means that *any* client that knows the primary name server's IP address and has TCP/IP connectivity to this AS/400 system can successfully query the AS1 name server.
6. Click **Add** to add the subnet that you want to allow.
7. Enter the subnets of clients that you want to allow to query this name server by entering the subnet's network address: 10.5.69.192 and mask 255.255.255.192.
8. Click **OK** (but do not click the second OK just yet).

   Now that we have authorized all clients located in the subnet 10.5.69.192 to query the primary name server AS1, we have implicitly denied clients from all other subnets. We have even denied access from localhost. For this scenario, we should also give access to clients from the 10.5.62.0 network and give access to the explicit address of 127.0.0.1 for localhost.
9. Repeat steps 6 through 8 for the subnet of 10.5.62.0. The subnet mask for 10.5.62.0 subnet is 255.255.255.0.
10. Click the second **Add** button to Limit domain data access to IP addresses.
11. Enter the IP address of localhost `127.0.0.1`. See Figure 421 to view the result.

    Note: Once you specify an address or subnet on the primary domain properties' security tab, it is required to specify the 127.0.0.1 IP address in the Limit domain data access to IP address list.
12. Click **OK**.

The previously explained configuration adds secure-zone TXT records to the primary domain configuration file. The secure-zone record defines an access list of IP addresses allowed to query your name server for data in a particular zone.

We just authorized all clients from two subnets, 10.5.69.192 and 10.5.62.0, to access the mycompany.com primary domain file on the primary name server AS1. However, when a client on one of those networks issues a reverse look-up query to the AS1 name server, will it be successful? Yes, because the security tabs on both the 69.5.10.in-addr.arpa and the 62.5.10.in-addr.arpa primary domain files are still blank. By default, any client has access to these files. If we need to have the same security on 69.5.10.in-addr.arpa and the 62.5.10.in-addr.arpa primary domain files as we do on mycompany.com, we need to repeat steps 3 through 12 for each in-addr.arpa file. Do not forget step 11, which specifies the localhost IP address to be an authorized address on both in-addr.arpa primary domain files' properties security tab.

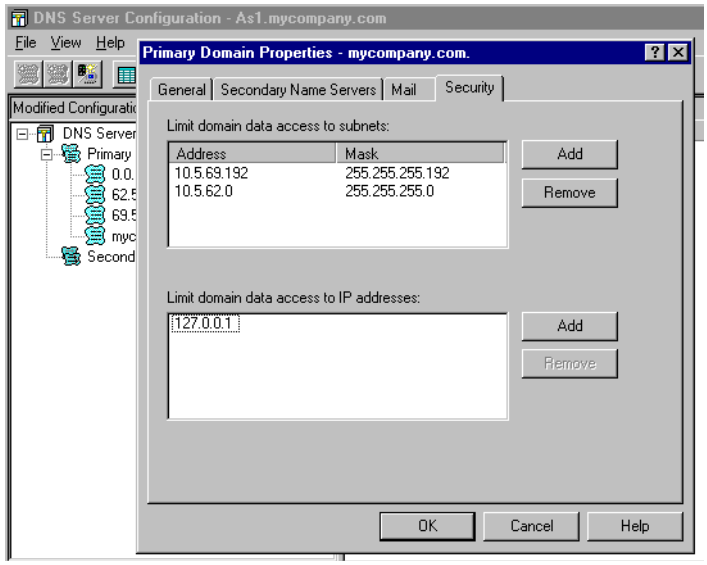


Figure 421. Restricting DNS queries by subnet and client IP address

Now that the primary DNS server is active on AS, it is time to reconfigure clients to start using the DNS server. An AS/400 system can be a client to a name server, so the AS/400 systems in the mycompany.com domain also require a configuration change.

Configuring AS/400 systems to query the DNS server

To reconfigure the AS1 system to query the DNS server, enter the command:

```
CFGTCP    option 12
```

Figure 422 on page 334 shows the resolver configuration on AS1.

Host name search priority specifies whether to search a remote Domain Name Server (DNS) to resolve a TCP/IP host name or to search the local TCP/IP host table first.

*LOCAL means that we want this system to first search the TCP/IP host table located on this system to resolve TCP/IP host names.

Specify *REMOTE if you want this system to search a remote DNS server to resolve TCP/IP host names before searching the local TCP/IP host table. The remote DNS server to use is specified by the Internet address parameter. For this scenario, all the AS/400 systems in the mycompany.com domain except for AS1 specify *REMOTE.

Note

Often some information in the host table and *LOCAL can keep systems operational to some degree even if the system cannot get to the remote name server. If something happens to the interfaces or the servers, applications can hang, trying to contact the servers and never get to the host table if *REMOTE is selected. For example, if local host information is in the host table, local mail can still be delivered if *LOCAL is selected.

You can specify up to three remote Domain Name Servers (DNS) to be used by this system. In our scenario, we specified 10.5.69.222 for the primary name server IP address.

```
                    Change TCP/IP Domain (CHGTCPDMN)

Type choices, press Enter.

Host name  . . . . . .   'AS1'

Domain name  . . . . . .   'mycompany.com'


Host name search priority  . . .   *LOCAL        *REMOTE, *LOCAL, *SAME

  Internet address . . . . . .   '10.5.69.222'
```

Figure 422. AS/400 resolver configuration

Note

If the DNS server specified first cannot be reached, the AS/400 system queries the next name server in the list.

If the name server configured at the top of the list responds, but sends back a negative response (in other words, the first name server does not know the answer), the AS/400 resolver queries the subsequent name servers in the list.

Configuring non-AS/400 clients to query the DNS server

All the clients in your network should have the DNS configuration updated to query the newly implemented primary DNS server. How to make this configuration change depends on your clients DNS support. Therefore, you do not provide instructions on how to update non-AS/400 client's DNS configuration. Figure 423 shows the DNS configuration for a Windows 95 client.

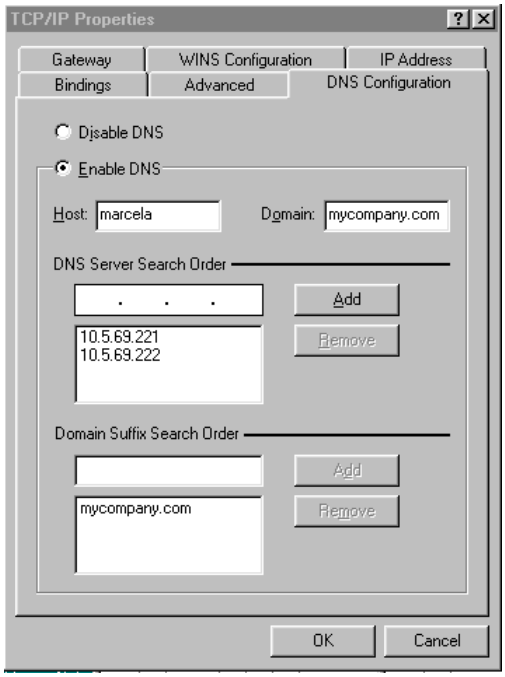


Figure 423. Windows 95 client DNS configuration

8.13 Additional information

We recommend that you refer to these other resources on DNS that complement this redbook:

- *AS/400 Internet Security: IBM Firewall for AS/400,* SG24-2162
- *AS/400 TCP/IP Autoconfiguration: DNS and DHCP Support,* SG24-5147
- *Domain names concepts and facilities* (RFC 1034), *Domain names implementations and specifications* (RFC 1035), and *Common DNS Operational and Configuration Errors* (RFC 1912)
- *DNS and BIND* by Albitz & Liu
- Operations Navigator online help
- The Information Center on the Web at: `http://www.as400.ibm.com/infocenter/` Search for `DNS`.
- comp.protocols.dns.bind newsgroup, which can be located on the Internet by entering: `http://www.dns.net/dnsrd`

  Click on the **Newsgroup** link.

Chapter 9. Getting started with DHCP on the AS/400 system

V4R2 of OS/400 added a new Dynamic Host Configuration Protocol (DHCP) server for automatic IP address assignment. DHCP is an Internet standard for automatically configuring devices on your LAN. It's a client/server protocol that enables you to centrally locate and dynamically distribute configuration information, including IP addresses. It may relieve the network administrator of a great deal of manual configuration work and can be of great benefit to mobile users.

This chapter covers the basics of setting up the AS/400 system as a DHCP server. For more detailed information, refer to *AS/400 TCP/IP Autoconfiguration: DNS and DHCP Support,* SG24-5147.

9.1 Before DHCP

BOOTP was the predecessor of DHCP. It enabled diskless workstations, such as the IBM thin client, to acquire an IP address and get their bootstrap information from a server on the network to which they are both connected.

The BOOTP server is listening on well-known UDP port 67. The BOOTP client uses the special broadcast address of all ones (255.255.255.255) to obtain its IP address. The BOOTP table is used for the mapping between the client hardware MAC address and the IP address.

BOOTP is only the first step in the two-step bootstrap procedure. It provides the client only with the information that is needed to obtain a memory image. Afterwards, the Trivial File Transfer Protocol (TFTP) can be used to get this memory image transferred from the server to the client.

The BOOTP flow between a client and a server is shown in Figure 424 on page 338. When the server receives a BOOTP request from a client, the server looks for a defined IP address based on the client's MAC address. It then replies with the client IP address and the name of the load file. The client initiates a TFTP request to the server for the load file.

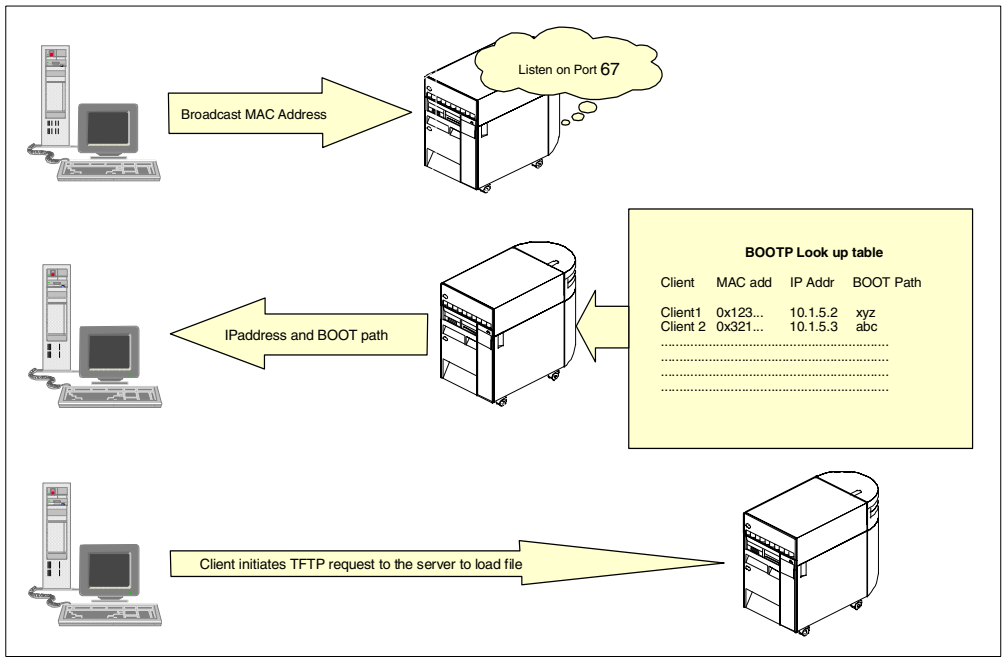


Figure 424. BOOTP flow between the client and server

It requires the server in the same subnet as the client that requests configuration information. BOOTP forwarding is a mechanism for routers to forward a BOOTP request between subnets. The agents that forward the BOOTP packets between clients and servers on different subnets are called *relay agent*s.

DHCP adds the capability of automatically allocating reusable network addresses and distributing additional host configuration options. DHCP clients and servers use existing BOOTP relay agents.

BOOTP clients can interact with DHCP servers and DHCP servers and BOOTP servers can coexist if configured properly. DHCP clients cannot interoperate with BOOTP servers.

The AS/400 DHCP server support in V4R2 accommodates the already existing BOOTP server that was available in earlier releases of OS/400. This AS/400 DHCP server support also accommodates BOOTP clients. Additionally, it performs all of the functions specific to BOOTP, as well as all of the added functionality that a DHCP server is assumed to carry. BOOTP and DHCP servers cannot run at the same time on the same system because both use the well-known UDP ports 67 and 68.

Internet Engineering Task Force (IETF) RFCs 2131 and 2132 describe DHCP protocols.

9.2 Overview of DHCP

Within a TCP/IP network, DHCP is used to pass configuration parameters to hosts. All three DHCP network components are shown in Figure 425:

- **DHCP host clients**: These run the DHCP client programs. The IBM Network Station and the client support included in TCP/IP for Windows 95 are some examples of this. The AS/400 system cannot be a DHCP client.
- **DHCP Servers**: These provide addresses and configuration information to BOOTP and DHCP clients within the TCP/IP network. The AS/400 system can be a DHCP server if V4R2 is installed.
- **BOOTP/DHCP Relay Agents**: If the DHCP server and the DHCP client are on different subnets, Relay Agents are used to forward configuration information between them, thereby eliminating the need for having a DHCP Server on each subnet to serve the existing DHCP clients. The AS/400 system can be a BOOTP/DHCP Relay Agent if V4R2 is installed.

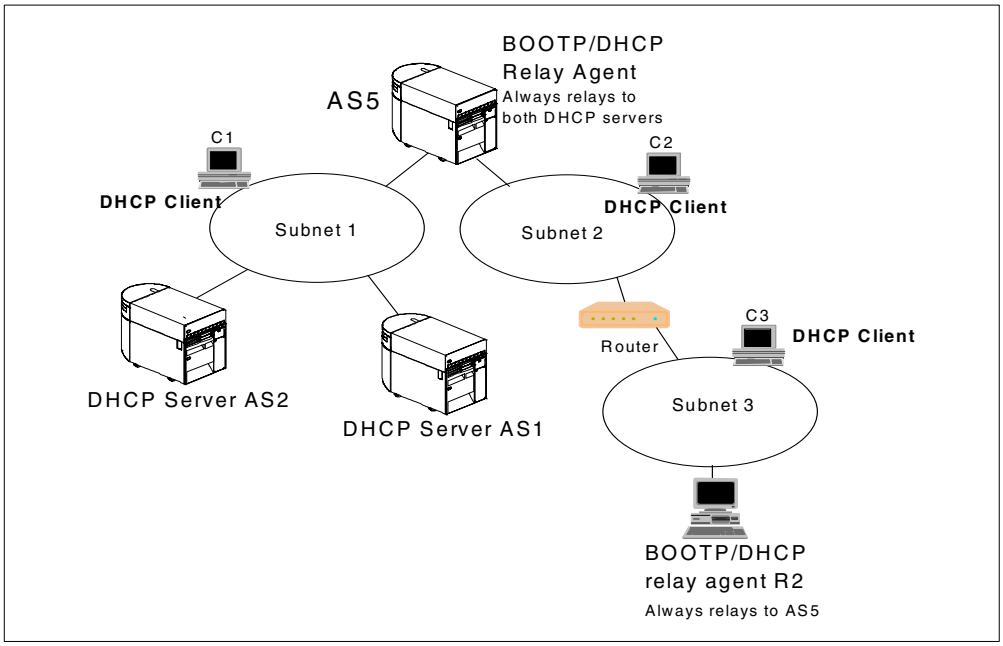


Figure 425. DHCP network components

9.3 How DHCP works

The DHCP protocol allows clients to obtain their IP address and additional IP network configuration from a DHCP server.

The IP addresses can be allocated permanently or leased for a specific time period. In the latter case, the client has to check with the server to re-validate the address and renew the lease on a periodic basis.

9.3.1 Acquiring configuration information

The DHCP clients use RFC-architected messages to accept and use the options served by the DHCP server. Figure 426 on page 340 describes the different steps within a DHCP cycle.

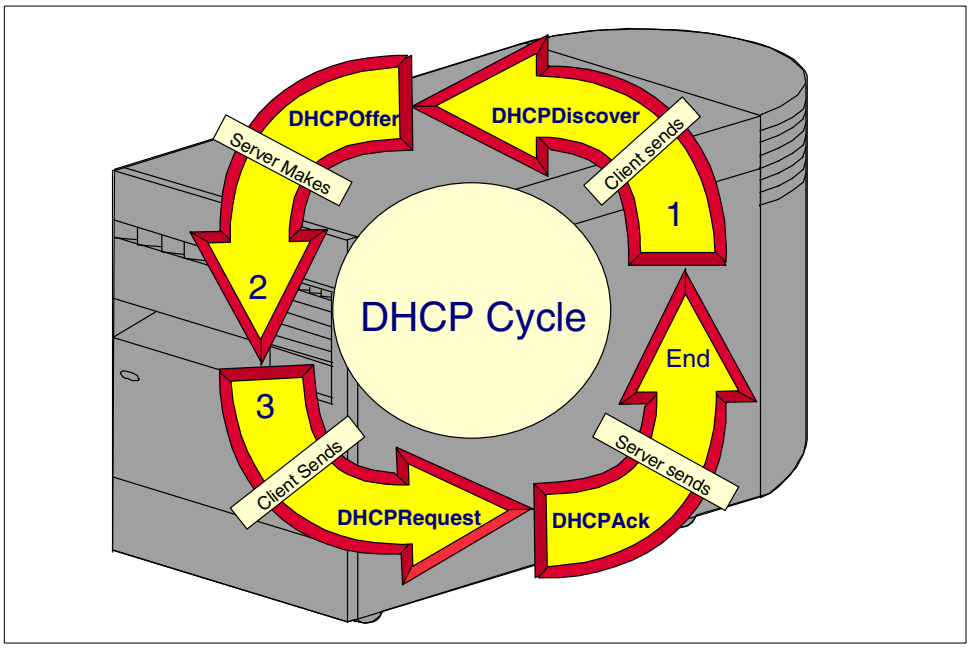


Figure 426. DHCP cycle

The DHCPDISCOVER message is shown in Figure 427. The DHCP client announces its presence by sending a broadcast message that contains its client ID. The message also requests an IP address (DHCPDISCOVER message) and desired options, such as a subnet mask, domain name server, domain name, and static route.

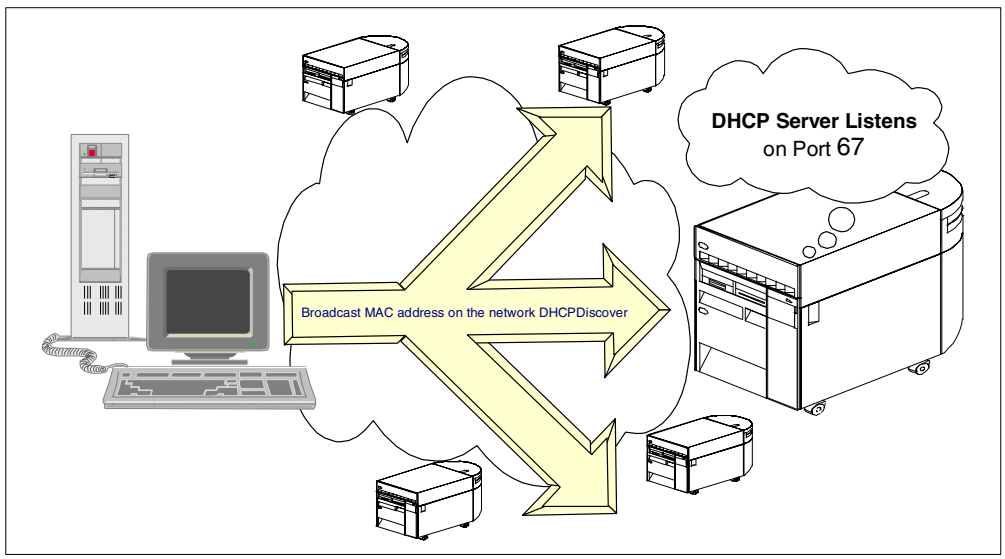


Figure 427. DHCPDISCOVER message broadcast

The DHCPOFFER message is shown in Figure 428. A DHCPOFFER message (offering the client IP address) can be sent by every DHCP server that receives the client's DHCPDISCOVER message. The server checks the configuration file to see if a static or dynamic address needs to be assigned to this client.

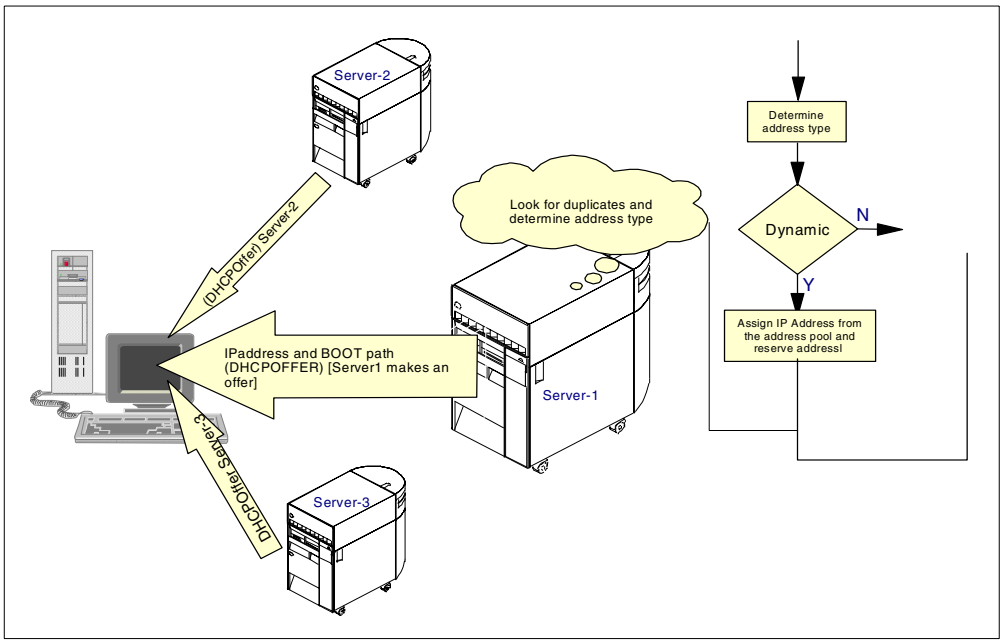


Figure 428. DHCPOFFER message

In case the IP address has not already been assigned before, the DHCP server checks if the IP address won't cause a duplicate IP address conflict before issuing an offer. The DHCP server also determines whether a static or a dynamic IP address needs to be assigned:

- If a dynamic address needs to be assigned, the DHCP server selects the least recently used IP address from the address pool that contains a range of IP addresses leased to clients.
- If a static IP address needs to be assigned, the DHCP server uses a client statement from the server configuration file to assign an IP address to that specific DHCP client.

The DHCP client receives offer messages from several DHCP servers. The client compares all received offers and selects the one that meets its criteria.

Note

Not all DHCP clients can wait for several offers and evaluate them afterwards. Many DHCP clients on the market today accept the first offer they get from a DHCP server.

The DHCPREQUEST message is shown in Figure 429 on page 342. The DHCP client broadcasts a message to tell which server was selected. By sending a DHCPREQUEST message, the client acquires the IP address that was served by that specific DHCP server to use it.

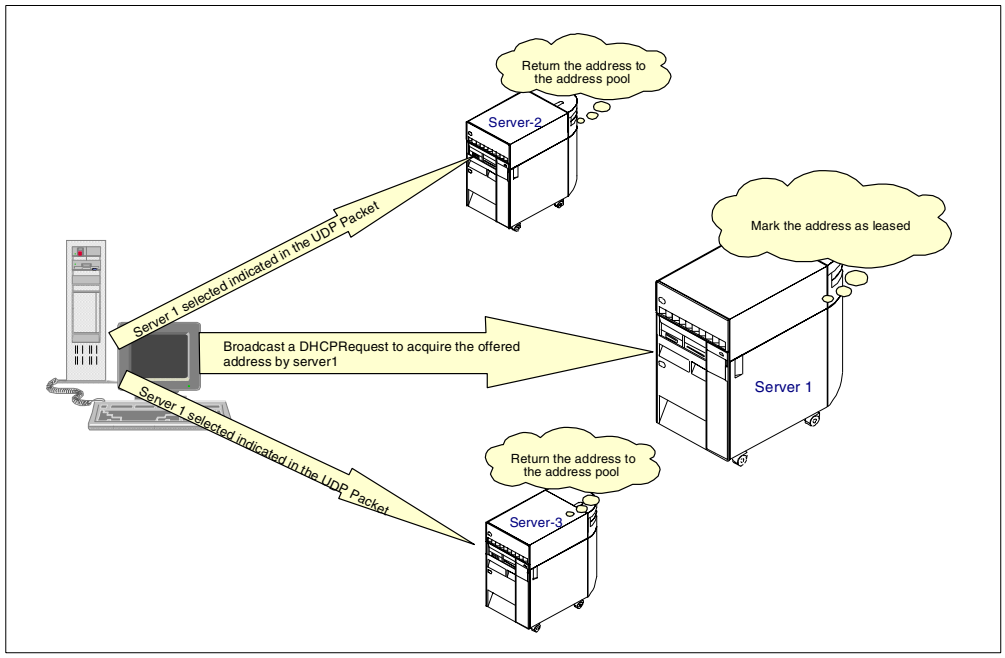


Figure 429. DHCPREQUEST message

A DHCP server that gets a message telling that the client has accepted his offer marks the corresponding IP address as leased. If a server gets a message telling that the client has selected the offer of another server, the server returns the address to the available pool. This is also the case if no message is received within a specific time period.

The DHCPACK message is shown in Figure 430. The server that was selected by the DHCP client to provide an IP address sends an acknowledgment to the client. This will contain additional configuration information.

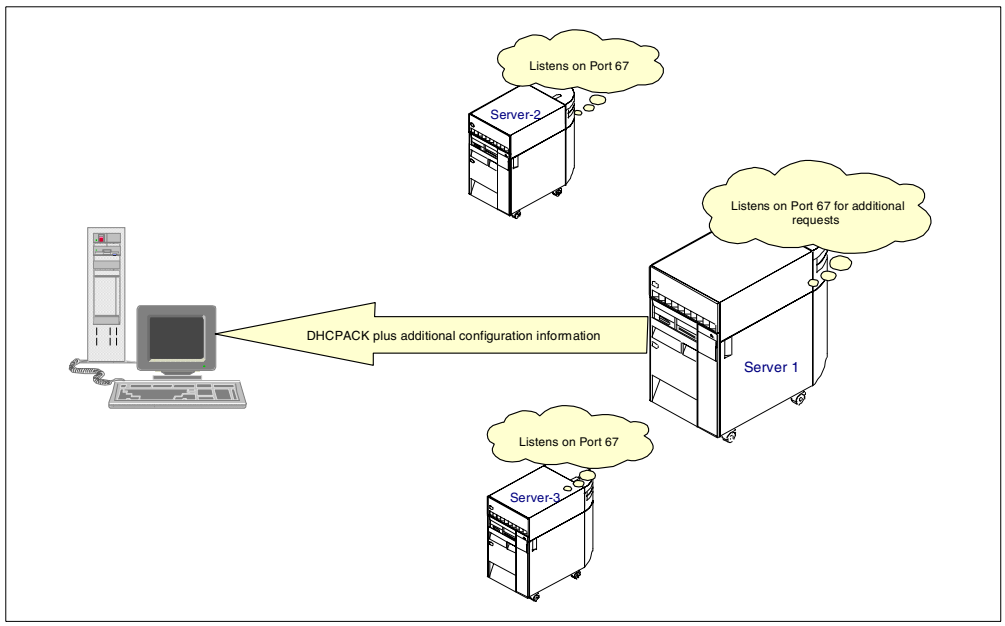


Figure 430. DHCPACK message

The DHCP client then determines if the configuration information is valid. If the lease is valid, the client specifies a BINDING state with the DHCP server and proceeds to use the IP address as well as the specified options.

The default set of client-requested options that IBM provides includes a subnet mask, domain name, domain name server, and static route.

9.4 Lease renewal

The DHCP client keeps track of the time that is still remaining on the lease. Normally, when half of the lease is expired, the client sends a renewal request to the DHCP server containing its current IP address and configuration information. The DHCP client's lease will be renewed if the server responds with a DHCPACK message. The lease timer will then be reset.

The server may also refuse the renewal request. By doing so, the DHCP client continues to use the IP address until the expiration of the lease time. The client then goes through the DHCP cycle to request a new IP address.

9.5 DHCP server configuration changes

A DHCP client retains the DHCP option values being assigned by the DHCP server for the duration of the lease. If the configuration of the DHCP server changes while a client is already up and running, those changes are not processed by the DHCP client until it either attempts to renew its lease or it is restarted.

9.6 BOOTP/DHCP Relay Agent

A Relay Agent forwards any BOOTP/DHCP requests that it receives on its subnet or from other subnets in the direction of the DHCP server. The mechanism of operation of a Relay Agent is described in the following text.

The Relay Agent knows the address of the DHCP server beforehand, and it knows where to forward the requests for that server. The Relay Agent can, therefore, be a router that receives and forwards requests.

The DHCP client creates a packet with a special field called RELAY AGENT. Initially, the client places all zeros in it. The Relay Agent recognizes that the RELAY AGENT field is all zeros and puts its own IP address in this field. It then pushes the packet into the next subnet and increments the hop count.

The next Relay Agent, if any, sees that the RELAY AGENT field in the packet is not all zeros, forwards the packet to the next server, and increments the hop count by one. This process is repeated until the packet reaches the DHCP server.

The DHCP server sends the DHCPOFFER back to the first Relay Agent and the Relay Agent forwards it to the originator client that broadcasted the DHCPDISCOVER. Once the client receives an IP address, the communication is direct between the DHCP server and the DHCP client.

In V4R2, the AS/400 system can either be a DHCP server or a BOOTP/DHCP Relay Agent.

9.7 DHCP server implementation on the AS/400 system

The following section describes the AS/400 DHCP server.

9.7.1 DHCP software prerequisites

The native DHCP support on AS/400 running V4R2 requires the following products:

- 5769-SS1 OS/400 V4R2 option 3
- 5763-XD1 V3R1M3 Client Access for Windows 95/NT

9.7.2 DHCP installation

To install DHCP support on your AS/400 V4R2 system, you should install 5769-SS1 OS/400 V4R2 option 3 on the AS/400 system and Client Access for Windows 95/NT (5763-XD1 V3R1M3) on your administrator's workstation.

The installation program performs the following actions:

- It creates the IFS subdirectories /QIBM/UserData/OS400/DHCP.
- It sets up the IFS files required for DHCP in the preceding directory. If any file already exists, it will remain "as is".

Tip

Perform the following steps to reset an existing configuration and start over:

1. Delete the IFS file `dhcpsd.cfg` file in /QIBM/UserData/OS400/DHCP.
2. From an AS/400 command entry display, enter the command:

   ```
   CALL QSYSDIR/QTODDINS
   ```

   This program will create a blank configuration file that the Operations Navigator GUI can edit.

If you suspect some DHCP files are corrupted, these steps could be performed as well. Reinstalling 5769-SS1 option 3 does not replace existing files.

After the installation, you can proceed with the DHCP server configuration using Operations Navigator. Figure 431 provides an overview of AS/400 DHCP server installation and configuration.

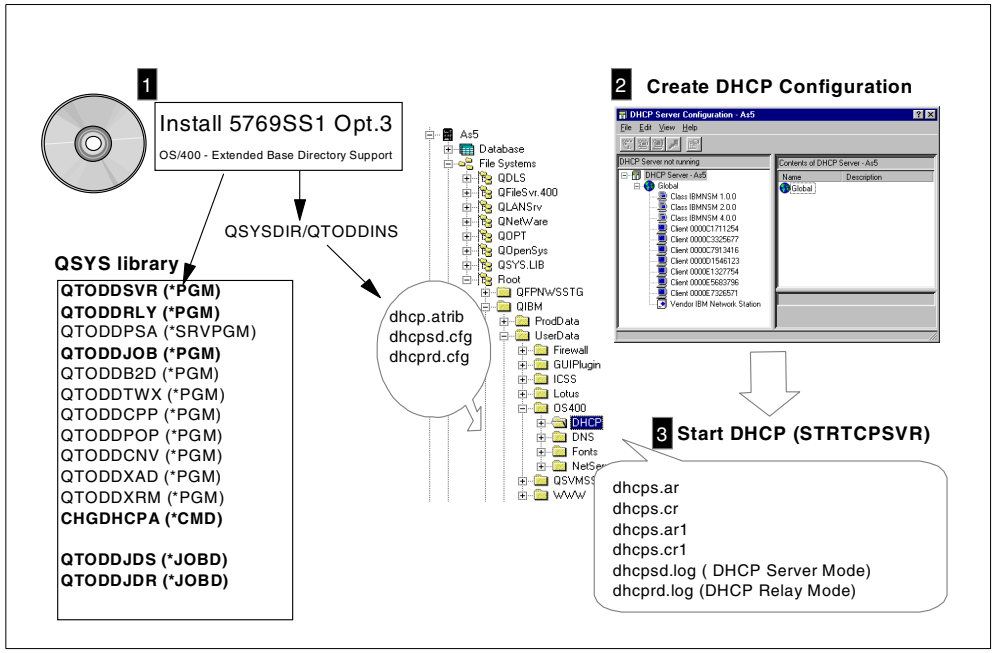


Figure 431. AS/400 DHCP server support Installation and configuration overview

9.7.3 DHCP configuration files

The files that DHCP requires are in the IFS directory /QIBM/UserData/OS400/DHCP. The files include:

- **dhcpsd.cfg**: This is the configuration file that DHCP reads when it runs as a regular DHCP server (transaction processing server).
- **dhcprd.cfg**: This is the configuration file that DHCP reads when it runs as a BOOTP/DHCP Relay Agent server.
- **dhcps.ar**: DHCP server non-volatile address records. This file contains up-to-the minute, actual address allocation from the address pools that the DHCP server administers when running in regular DHCP server mode.
- **dhcps.cr**: DHCP server non-volatile client records. This file contains up-to-the minute data on the actual clients that this DHCP server is servicing when running in regular DHCP server mode.
- **dhcps.ar**: DHCP server backup of non-volatile address records. The DHCP server takes an hourly backup of dhcps.ar, the non-volatile address record file.
- **dhcps.cr1**: DHCP backup of server non-volatile client records. The DHCP server takes an hourly backup of dhcps.cr, the non-volatile client records file.
- **dhcp.attrib**: DHCP attributes file. Stores the current value of the CHGDHCPA command parameters, with the exception of the AUTOSTART parameter.

9.7.4 DHCP administration program

After the HCP server is running, and addresses have been assigned, you may want to see which addresses are assigned and obtain other information about the address. A program is available in the technical studio to let you look at this type of information. To find and download the program, go to the Technical Studio Web site at: `http://www.as400.ibm.com/tstudio/`

In the search field, type `DHCP Administration` and click **Search**. A page of matches should be returned. Look for the title “DHCP Server Administration Program for Windows 95/NT”. Click on the link, and you are taken to the page that allows you to download the code.

9.8 DHCP jobs running in QSYSWRK subsystem

The DHCP server jobs will run in the QSYSWRK subsystem. The jobs are described in this section:

- **QTODDHCPS**
  - This job runs if the DHCP Mode attribute is set to *SERVER.
  - The DHCP server will run as a regular DHCP transaction-processing server.
  - The DHCP server will use well-known ports 67 and 68.
  - This job starts with job description QTODDJDS.

> **Note**
>
> You can use Work with Spooled File (WRKSPLF) command for User QTCP to browse the DHCP server job log.

- **QTODDHCPR**
  - This job runs if the DHCP Mode attribute is set to *RELAY.
  - The DHCP server runs only as a BOOTP/DHCP Relay Agent.
  - The BOOTP/DHCP Relay Agent will only use runs on well-known port 67.
  - This job starts with the job description QTODDJDR.

9.9 Changing the DHCP attributes

You can use the Change DHCP Attributes (CHGDHCPA) command to set the AUTOSTART attribute. This determines whether the DHCP server starts automatically when TCP/IP is started using the STRTCP command. This attribute is ignored by the STRTCPSVR command. STRTCPSVR *DHCP starts the DHCP server regardless of the value of the AUTOSTART attribute.

It is also possible to set this attribute from the Operations Navigator interface (Figure 432).

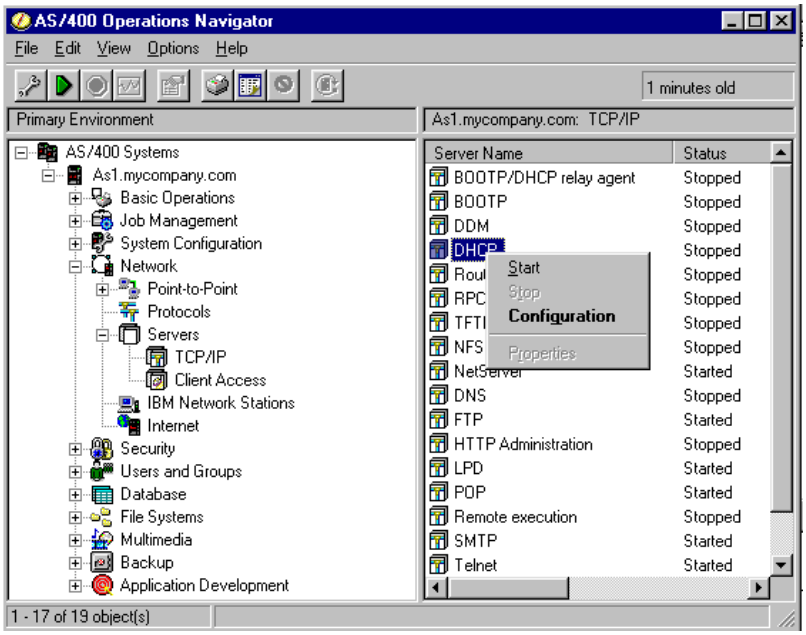


Figure 432. DHCP configuration through Operations Navigator

Use the CHGDHCPA command to set the MODE attribute that determines the DHCP server behavior:

- Set the MODE attribute to *SERVER if you want the DHCP server to automatically assign reusable IP addresses to DHCP clients in response to DHCP requests.
- Set the MODE attribute to *RELAY if you want the DHCP server to function only as a BOOTP/DHCP Relay Agent. A BOOTP/DHCP Relay Agent forwards BOOTP or DHCP packets from hosts to active BOOTP or DHCP servers and from the servers back to the hosts. It performs no BOOTP or DHCP server functions.

The attributes file /QIBM/UserData/OS400/DHCP/dhcp.attrib is updated with the values that you have specified by using the CHGDHCPA command.

9.10 Starting and stopping the DHCP server

You can use the `STRTCPSVR SERVER(*DHCP)` command to start the DHCP server and the `ENDTCPSVR SERVER(*DHCP)` command to stop it.

Startin and stopping the DHCP server can also be done by using Operations Navigator (Figure 432).

9.11 Configuring the DHCP server using Operations Navigator

The installation and configuration of the AS/400 DHCP server is done through Operations Navigator. This is the only configuration interface for the AS/400 DHCP server. The Operations Navigator DHCP configuration wizard provides a simple process for quickly configuring the DHCP server.

To start the DHCP server configuration from Operations Navigator, select **AS400sytem name->Network->Server->TCP/IP**. The window shown in Figure 432 on page 347 is displayed.

To use the Operations Navigator interface, you need to install Client Access/400 for Windows 95/NT V3R1M3 on your administrator's PC. You also need to verify whether the Host servers are started on the AS/400 system. You can use the Start Host Server (STRHOSTSVR) command to start them if they weren't already started.

9.12 DHCP implementation in a simple AS/400 network

Now we discuss the setup of the DHCP server running on the AS/400 system in a simple TCP/IP network. We also explain how to configure both a Windows 95 client and the IBM Network Station as DHCP client.

Let's consider a local area network that is physically complete: all systems and clients are cabled to the network and are able to attach. Let's assume that the network in the following figure is a new network. We are free to choose any possible TCP/IP addressing scheme. We won't cover the complexities that arise with an existing network and hardcoded TCP/IP addresses (Figure 433).

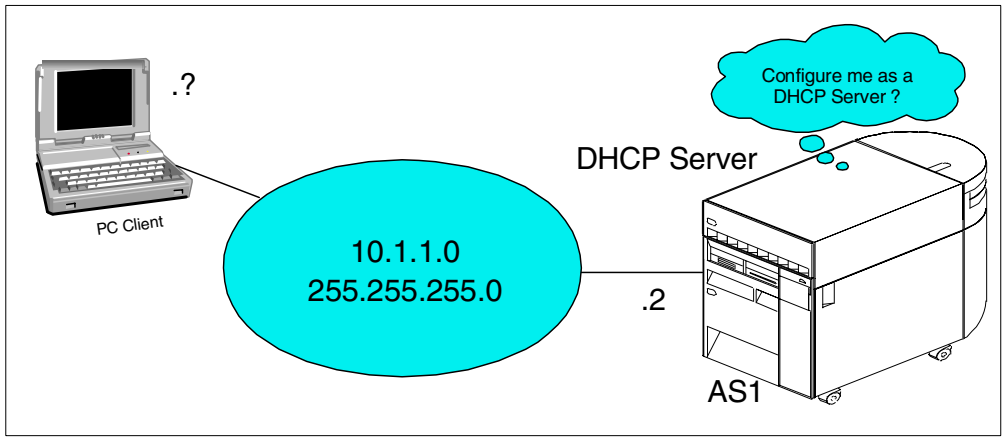


Figure 433. Simple TCP/IP network with AS/400 DHCP server

In our example, we have a single AS/400 DHCP server with a class A TCP/IP address and a single subnet. The subnet mask 255.255.255.0 allows the AS/400 DHCP server to service 253 clients because the AS/400 host address remains constant and will be removed from the address pool.

A discussion of a possible migration from BOOTP to DHCP, or scenarios dealing with networks with routers and bridges or complex subnetting issues, are fully covered in *AS/400 TCP/IP Autoconfiguration: DNS and DHCP Support*, SG24-5147.

9.12.1 Summary of tasks to be performed

You should perform the following steps to configure the DHCP server, as well as the DHCP clients:

1. Verify the hardware, software, and configuration prerequisites.
2. Configure a network interface on the AS/400 system.

3. Configure and start a TCP/IP interface.
4. Gather information to configure the DHCP server.
5. Configure the DHCP server.
6. Start the DHCP server.
7. Configure the Windows 95 DHCP client.

9.12.2 Verifying hardware, software, and configuration prerequisites

Be sure the following prerequisites are met before configuring your AS/400 system to act as a DHCP server:

- Hardware prerequisites:
  1. Verify that your AS/400 system has a LAN adapter installed and is cabled to the network.
  2. All clients within your network should have the correct network interface card installed. Verify also that you have installed all the required drivers.
- Software prerequisites:
  1. The DHCP support is part of 5769-SS1, base option 3, OS/400 - Extended Base Directory Support.
  2. The licensed program product 5769-SS1, option 12 (OS/400 - Host Servers) should be installed.
  3. You need to ensure that the AS/400 Operations Navigator is installed and configured on the administrator's PC. The system administrator will need it to configure DHCP on the AS/400 system.

     Note

     With V4R2, the Client Access code for the client requires no license. Effectively, the base Client Access code is free.

  4. For PC clients that connect to the network, DHCP support is included in Windows 95.
  5. To use and connect an IBM Network Station, you need to ensure that IBM Network Station Manager for AS/400 code is installed.
- Configuration prerequisites:

  A line description for your AS/400 LAN interface card must be added if it does not exist. Refer to Chapter 2, "TCP/IP basic installation and configuration" on page 7, for instructions about setting up your AS/400 system to support TCP/IP.

9.12.3 Configuration overview

Once the AS/400 system is configured to support the basic TCP/IP functions, you can set it up to act as a DHCP server. The following steps must be accomplished to successfully serve DHCP information:

1. Configure the DHCP server support through Operations Navigator.
2. Change some DHCP attributes.
3. Configure the clients to use DHCP.

9.12.3.1 Gathering information to configure the DHCP server

To use the Operations Navigator DHCP configuration effectively, you need to know how you want to set up and manage your networks and subnets with DHCP. You also need to know what address range or ranges you want to use for leasing. You must decide which system is the DHCP server, which ones are the BOOTP/DHCP Relay Agents, and which one performs DHCP backup functions. You must also know the IP addresses that must be reserved for special hosts, such as routers, DNS servers, and firewalls. It is useful to refer to a network diagram that shows the subnet masks and IP addresses for your networks, routers, and clients while you are configuring DHCP.

Our starting point for the DHCP environment explained from here on is the network diagram shown in Figure 433 on page 348. The information shown in the following tables is based on the network picture and other network data.

Table 13 shows general information about AS1 as a TCP/IP host. Table 14 provides more specific information about AS1 as a DHCP server.

Table 13. Planning the DHCP server: AS1 TCP/IP information

Host Name	As1
Description	DHCP server
Domain Name	mycompany.com
IP Address	10 . 1 . 1 . 2
Mask	255.255.254.0
Line Description	TRNLINE1

Note: The *Configuration Reference* column in Table 14 points to the place in Operations Navigator DHCP server configuration where you can configure the particular parameter. You can specify many of these configuration options through the DHCP configuration wizard the first time you configure DHCP.

Table 14. Planning the DHCP server AS1: DHCP server overview

#	Question	Answer	Configuration reference
1	Is the BOOTP server already configured on your system?	No	DHCP configuration wizard
2	Do you want to migrate the BOOTP configuration to DHCP?	N/A	File->Migrate BOOTP
3	What is the default lease time for this server?	24 hours	Global->Properties->Leases
4	Start the DHCP server when TCP/IP starts?	Yes	Server Properties->General
5	List the DHCP server IP interfaces that will be serving DHCP clients.	10.1.1.2	See the network diagram (Figure 433 on page 348).
6	List the subnets that will be administered by this DHCP server.	10.1.1.0	See the subnet planning table (Table 15).
7	Do you want to add a new subnet to be administered by this server?	Yes	Global->New Subnet - Basic Global->New Subnet - Advanced See the subnet planning table (Table 15).

#	Question	Answer	Configuration reference
8	Do you want to log DHCP server activity?	Yes	Server Properties->Logging
9	Do you want the DHCP server to support any client from any subnet?	Yes	Server Properties->Client Support
10	Do you want the DHCP server to support BOOTP clients?	No	Server Properties->Client Support
11	Do you want the DHCP server to reject requests from specific clients (for example, for security reasons)?	No	Global->Properties->Exclude Client
11	Can your DHCP clients (other than IBM Network Stations) identify the class they belong to?	No	
12	If answer to 11 is Yes, do you want to add a new class to serve the DHCP clients that belong to that class?	N/A	Global->New Class

Table 15 provides information about subnet 10.1.0.0 being administered by the DHCP server AS1.

Table 15. Planning the subnet 10.1.1.0 administered by AS1

#	Question	Answer	Configuration Reference
1	Subnet name	10.1.1.0	Subnet Properties->General
2	Subnet description	Our_Company	Subnet Properties->General
3	Subnet address	10.1.1.0	Subnet Properties->Address Pool
4	Subnet mask	255.255.255.0	Subnet Properties->Address Pool
5	Address range	10.1.1.1 10.1.1.254	Subnet Properties->Address Pool
6	Lease time	Inherit from server (12 hours)	Subnet Properties->Leases
7	Exclusions (exclude hosts that required a particular IP address and are manually configured).		Subnet Properties->Address Pool
	Name: Router x Description: Reserved for future router IP address: 10.1.1.1	AS1 DNS/DHCP server 10.1.1.2	
8	Domain Name Server IP address to deliver to clients in this subnet.	10.1.1.2	Subnet Properties->Options-> Option 6 (Domain name server)
9	Gateway IP address to deliver to clients in this subnet.	N/A	Subnet Properties->Options-> Option 3 (Router)
10	Offer options to client in this subnet 01 - Subnet mask 06 - Domain name server	 255.255.254.0 10.1.1.2	Subnet Properties->Options->

9.12.3.2 Configuring the DHCP server through Operations Navigator

If you are configuring DHCP on a system that does not have an existing DHCP configuration, the DHCP configuration wizard is started automatically by Operations Navigator. The wizard helps you create a basic DHCP server configuration.

Note

To reset an existing configuration and start over, perform the following steps:

1. Delete the IFS file dhcpsd.cfg file in /QIBM/UserData/OS400/DHCP.
2. Call `QSYSDIR/QTODDINS` from an AS/400 command entry display. A blank configuration file is created that the Operations Navigator GUI can edit.

Perform the following steps to start the DHCP configuration wizard:

1. Start Operations Navigator.
2. Click **as1.mycompany.com** to select the system name (Figure 434).

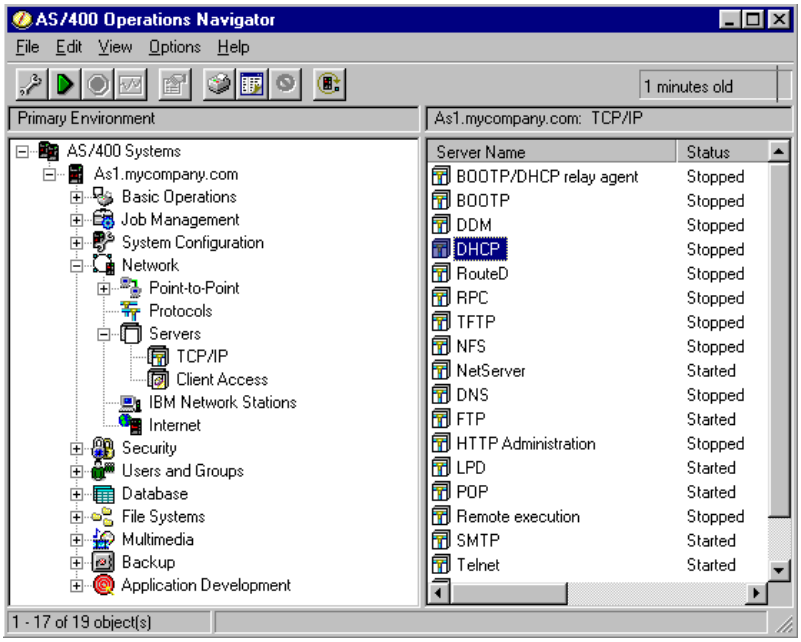


Figure 434. Select the system to configure the DHCP server

3. Double-click **Network**.
4. Double-click **Servers**.
5. Double-click **TCP/IP**.
6. Double-click **DHCP**. Figure 435 shows the result of startup of the DHCP configuration wizard.

Note

If the DHCP configuration wizard does not show up, it is likely that there is already an existing DHCP configuration. To start the wizard and replace the existing configuration, select **File->New Configuration**.

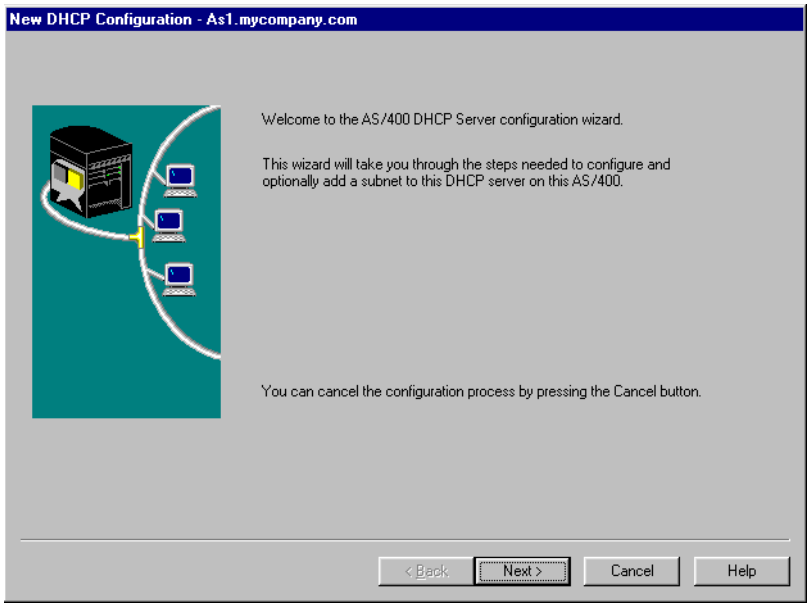


Figure 435. DHCP configuration wizard screen

7. Click **Next**.
8. Select **Yes** to add a new subnet to the DHCP server.
9. Leave the Twinax IP workstation controller address box blank, and click **Next**.
10. Define the range of addresses to use within the subnet.

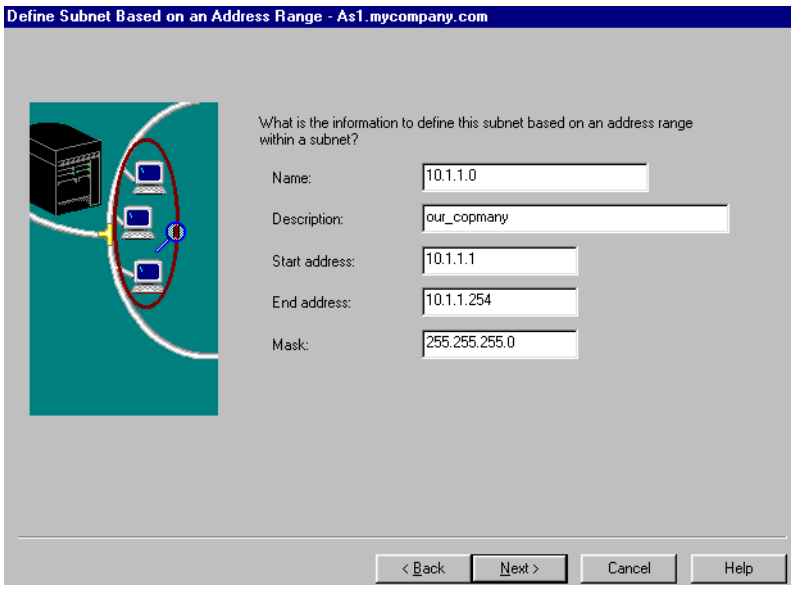


Figure 436. Subnet configuration within the DHCP configuration wizard

11. Define a lease time for the client to keep the address served. Click **Next** to use the default lease time of one day.
12. Specify the IP addresses of the hosts to be excluded. The DHCP server does not deliver these addresses to clients (Figure 437 on page 354).

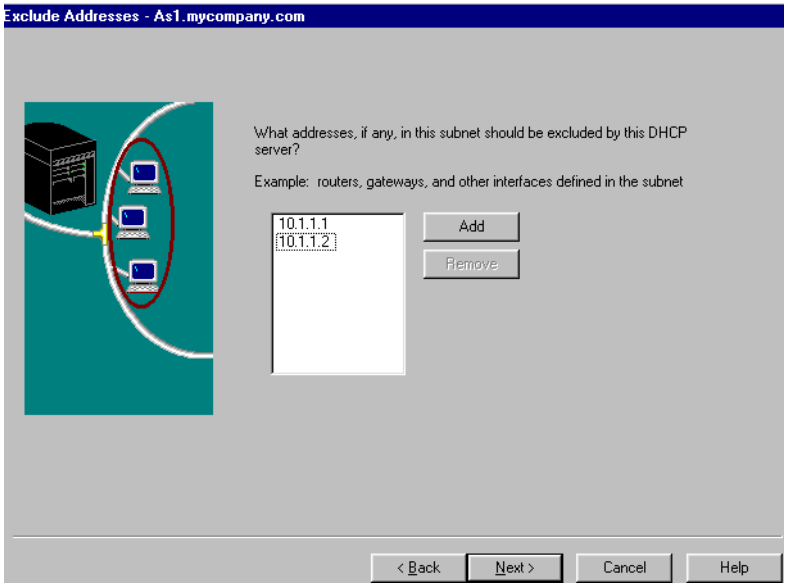


Figure 437. Host IP addresses to be excluded within the subnet

13. Click **Next** to *not* deliver the IP address of a gateway to clients. There is only one subnet in this scenario.
14. Answer **Yes** to the question "Would you like the DHCP server to deliver domain name server address to clients in this subnet?" Specify the DNS IP address (Figure 438). Then, click **Next**.

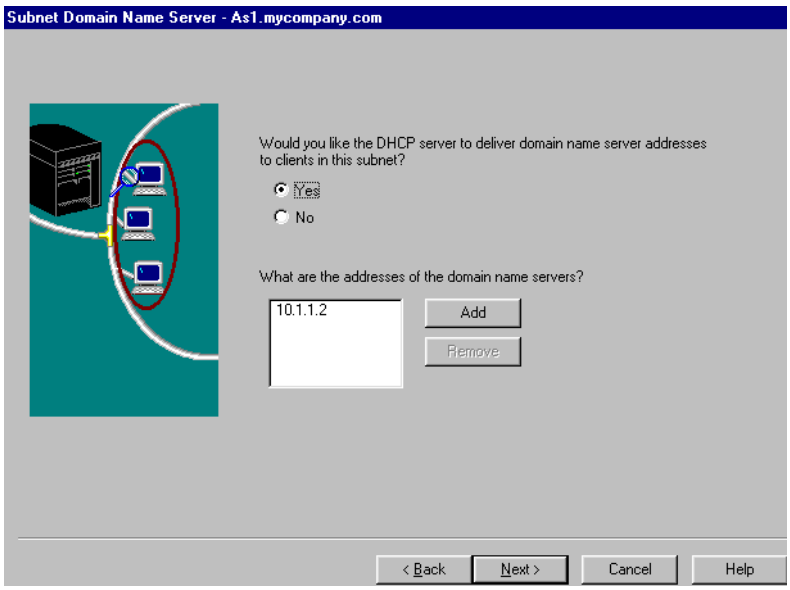


Figure 438. DNS IP address to be delivered to clients within the subnet

15. Answer **No** to the question "Would you like the DHCP server to deliver domain names to clients in this subnet?" Click **Next**.
16. Select Support any clients on this subnet. Click **Next**.
17. Select **Yes** to start the DHCP server when TCP/IP starts, and then select **No** to start the DHCP server now. Click **Next**.

18. Figure 439 shows the DHCP configuration summary window with all the options that you have selected up to this point. Click **Finish**.

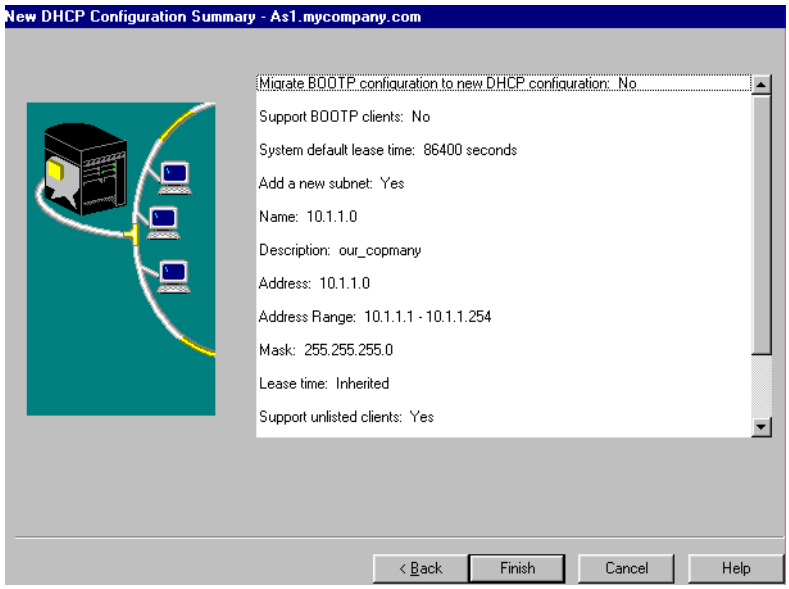


Figure 439. DHCP configuration summary

Figure 440 shows the DHCP server configuration.

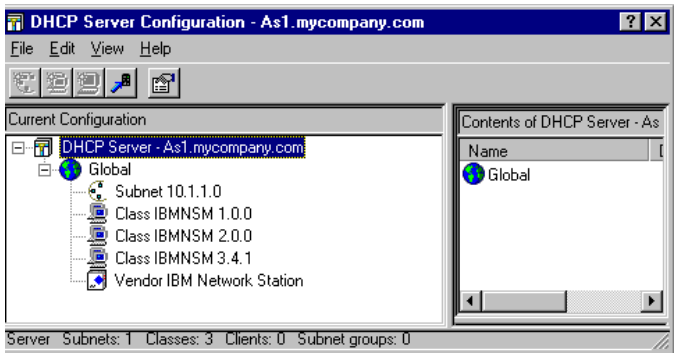


Figure 440. DHCP server configuration

We have now finished the configuration of a simple network to use the AS/400 DHCP server. One subnet from a class A IP address was created using the mask 255.255.255.0. This allows up to 254 IP addresses within the subnet pool to be served to clients. Now follow these steps:

1. From the DHCP Server configuration display (Figure 440), click **Subnet 10.1.1.0**, and select **Properties** from the context menu. The screen shown in Figure 441 on page 356 appears.
2. Click the **Options** tab to add a subnet mask that is served to the clients.
3. Highlight option **1** (subnet mask*)* from the Available options window, and then click **Add**.
4. At the bottom of the display, specify the appropriate subnet mask for the clients to use in the Subnet mask window.

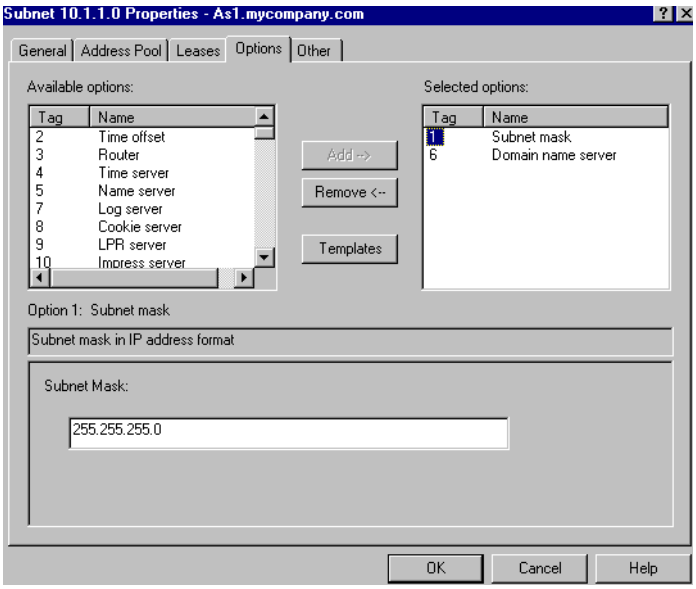


Figure 441. DHCP server options display

5. Notice in Figure 441 that domain name server option 3 is already configured. Specify the IP address of the domain name server when prompted by the configuration wizard.
6. Click **OK**.

Now we are going to assign a longer lease time for a Token-Ring attached IBM Network Station. A longer lease time can be useful for clients that are not mobile. A longer lease reduces the number of lease renewals the client must request which, in turn, reduces some of the network traffic. One way to specify a longer lease time for the Token-Ring IBM Network Station is to specify a lease time for the class they request. Use the documentation that came with the device to determine the class the device will request. In this example, we use a Token-Ring attached Network Station model 100.

1. From the DHCP Server configuration display shown in Figure 440 on page 355, right-click **Class IBMNSM 1.0.0**. to open a context menu. This class is for Token-Ring attached IBM Network Stations.
2. Select **Properties**.
3. Click the **Leases** tab.
4. Click **Duration**. From the pull-down menu, choose **weeks**, and specify `1` to set the lease duration to one week.
5. Click **OK**.

The DHCP server services requests from BOOTP clients. However, this is not the default and must be enabled. To enable the DHCP server to service BOOTP requests, perform these steps:

1. On the DHCP Server configuration display shown in Figure 440 on page 355, right-click **DHCP Server - As1.mycompany.com**, and select **Properties** from the context menu.

2. Click the **Client support** tab, and select both **BOOTP clients** and **Unlisted clients**.
3. Click **OK**.

9.12.4 Configuring DHCP clients

This section covers the DHCP client support, as well as the IBM Network Station DHCP client support.

9.12.4.1 Configuring a Windows 95 client

You should perform the following steps to enable DHCP on your Windows 95 workstation:

1. On your desktop, double-click **My Computer**.
2. Double-click **Control Panel**.
3. Double-click **Network**.
4. Select the **TCP/IP** network component in the **Configuration** tab, and click **Properties** (Figure 442).

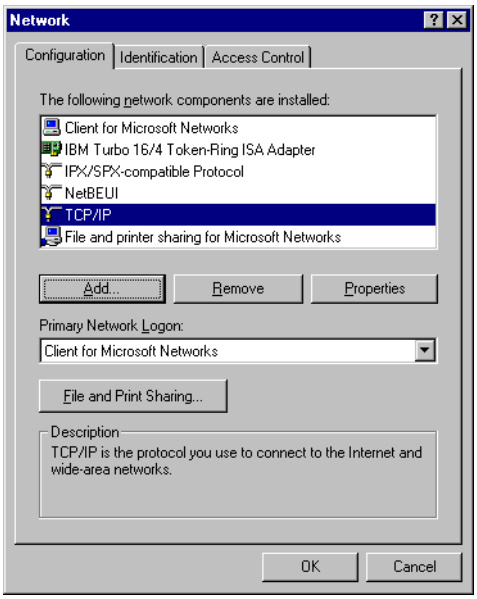


Figure 442. Network Neighborhood Properties display

Figure 443 on page 358 shows the display that appears.

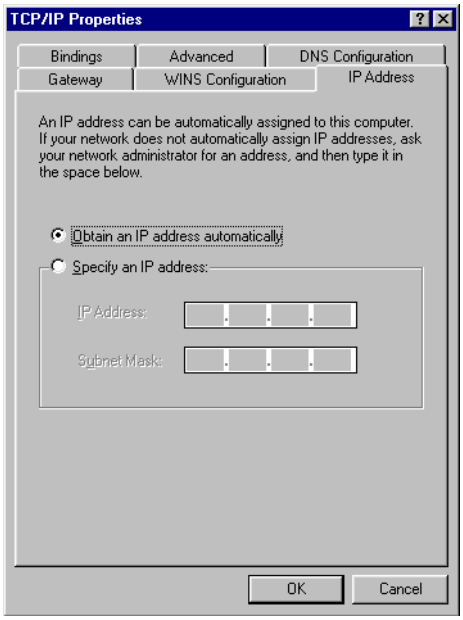


Figure 443. TCP/IP Properties display

5. Select **Obtain an IP address automatically**.
6. Click **OK**
7. Click **OK** again, and restart your computer following the displayed windows prompts.

The Windows 95 client now starts broadcasting the DHCPDISCOVER message.

On the Windows 95 PC, you can use the WINIPCFG.EXE Windows program to verify the current IP configuration of the Windows 95 PC. After you click on the **More Info** button, the display shown in Figure 444 appears.

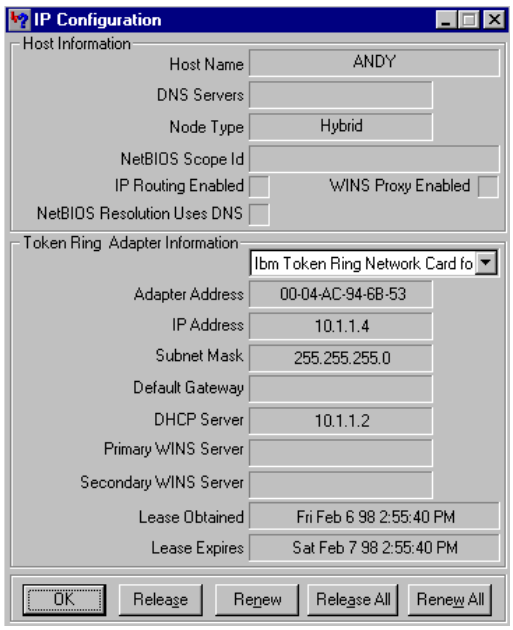


Figure 444. WINIPCFG Windows 95 IP configuration

Chapter 10. Network Address Translation and IP Packet Filtering

This chapter covers the basics of setting up the AS/400 system to perform Network Address Translation (NAT) and IP Packet Filtering. NAT and IP Packet Filtering can be used separately, but are commonly used together because the functions complement each other and enhance the network security of the TCP/IP network.

For more information on NAT and IP Packet Filtering, please refer to *TCP/IP Tutorial and Technical Overview,* GG24-3376, *TCP/IP Configuration and Reference,* SC41-5420, and *AS/400 Internet Security: IBM Firewall for AS/400,* SG24-2162.

This chapter includes background information on NAT and IP Packet Filtering and shows how to configure and manage the functions on the AS/400 system. Different scenarios using NAT and IP Packet Filtering are included to assist you in your network design.

10.1 Introduction to NAT and IP Packet Filtering

The best way to learn about NAT and IP Packet Filtering is by seeing example scenarios. This section introduces common examples of what NAT and IP Packet Filtering can be used to accomplish.

10.1.1 Example scenarios

The use of IP Packet Filtering to provide additional protection can reduce or eliminate the need for a separate firewall product in the cases described.

An AS/400 system running production applications should generally not be directly connected to the Internet even though OS/400 packet filtering is used. This is because AS/400 CPU resources would be necessary to reject undesirable packets, and mistakes in configuration could have serious consequences. A firewall is recommended for protecting a production AS/400 system from the Internet.

10.1.1.1 Protecting a private subnetwork using IP Packet Filtering

Company ABC (Figure 445 on page 360) has an organization that performs top secret research. The research organization has its own private subnetwork and systems on this subnetwork should only be used by research personnel. An AS/400 system is configured as a gateway between the research network and the rest of the corporate network. Since the research personnel only require occasional access to the corporate network, the AS/400 system is acting as a “casual” router. OS/400 IP Packet Filtering is configured to block all connection attempts from the corporate network, while permitting connections from the research network to be routed through to the remainder of the corporate network.

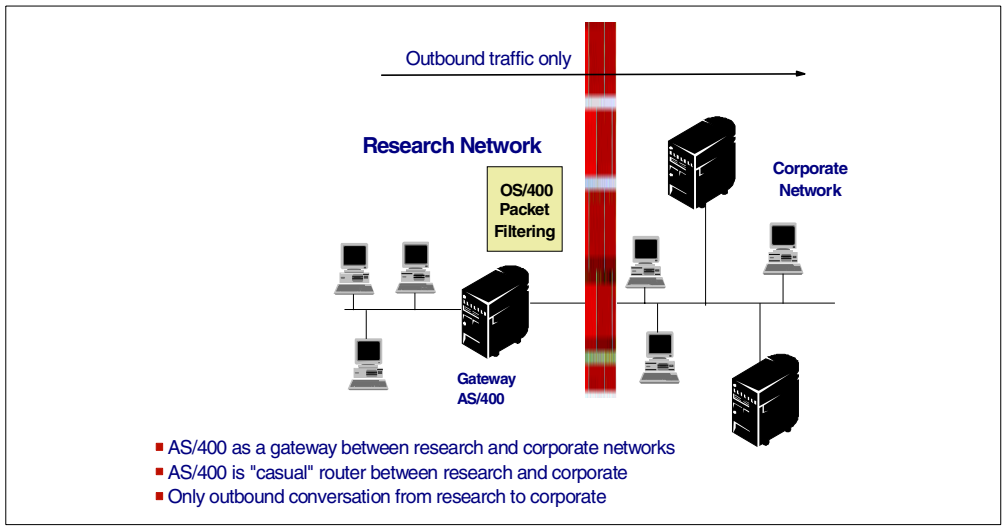


Figure 445. Protecting a private subnet

10.1.1.2 Protecting a public Web server using IP Packet Filtering

Great Deals Inc. (Figure 446) is already connected to the Internet and has a firewall in place to protect their corporate network. They have been using a hosting service to provide information about their company via the World-Wide Web. Great Deals Inc. decides they want to bring their Web server “in-house” so that they can begin to provide Web applications for inventory checking and order status. They want to run these applications on an AS/400 system that is located outside of their corporate firewall (Firewall for AS/400) so that the firewall is not burdened with processing large quantities of HTTP traffic. OS/400 IP Packet Filtering is configured to only allow HTTP traffic (TCP port 80) to reach the AS/400 Web server from the Internet. Other protocols, such as Telnet and FTP, are permitted to reach the Web server system only when the connections are initiated from inside the Great Deals Inc. network.

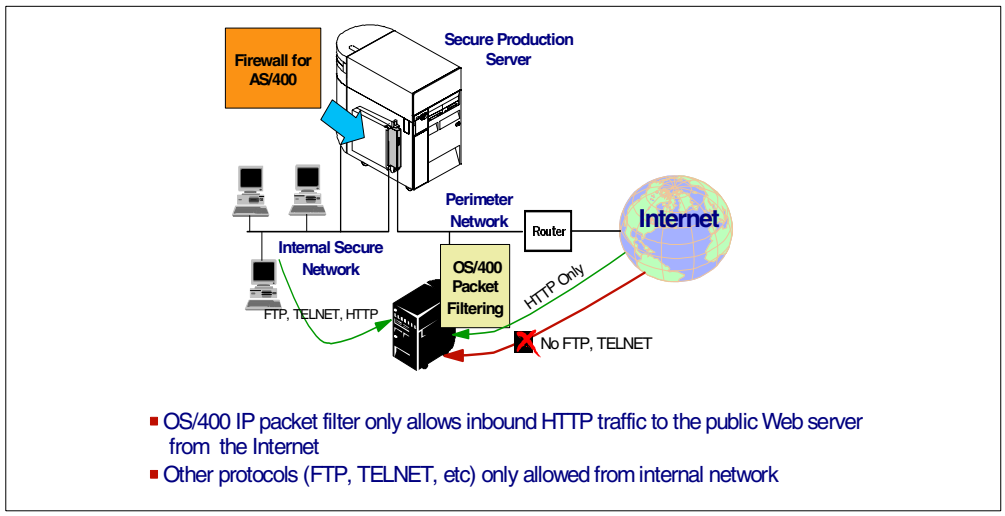


Figure 446. Protecting a public Web server using IP Packet Filtering

10.1.1.3 Connecting partner private networks using IP Packet Filtering

BigMart and BigDistributor (Figure 447) are partners that want to establish a private TCP/IP connection for the exchange of information. Information will be made available via HTTP and FTP servers. BigMart and BigDistributor want to use their existing AS/400 systems as a gateway to the other's network. OS/400 IP Packet Filtering is configured on each gateway system to allow only HTTP and FTP traffic from the other's network to reach selected Web servers.

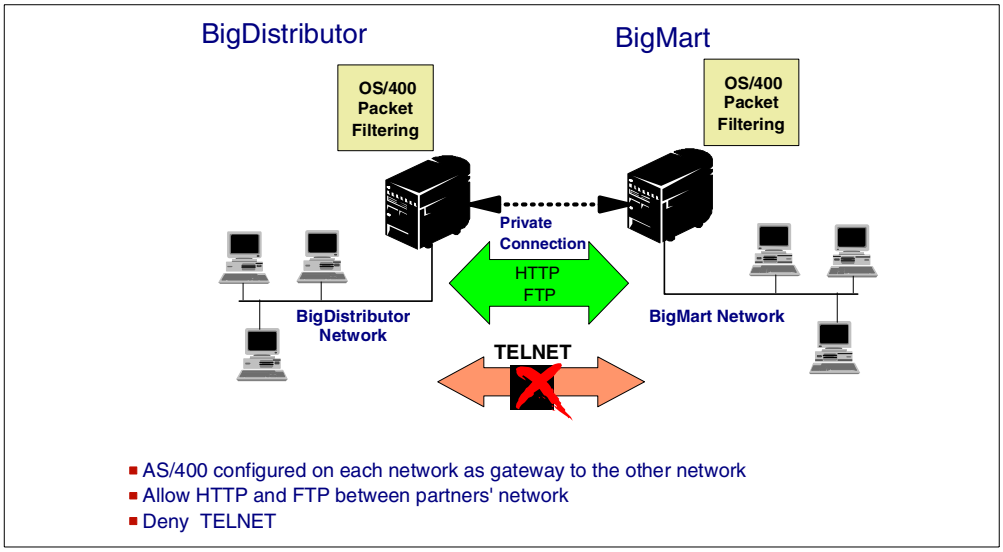


Figure 447. Connecting partners private networks using IP Packet Filtering

10.1.1.4 Connecting networks with duplicate IP addresses using NAT

The Network Address Translation (NAT) functions of OS/400 should be considered when connecting two previously disjointed networks that have inconsistent or incompatible IP addressing structures.

BigMart and BigDistributor (Figure 448 on page 362) are partners that want to establish a private TCP/IP connection for the exchange of information. BigMart must access inventory services provided by a server in the BigDistributor network. Unfortunately, both networks use the 192.168.x.x reserved address range, and there are several addresses that are valid in both networks. NAT is used by BigDistributor to make its server visible to BigMart using an IP address that is convenient to the BigMart network (Figure 448 on page 362).

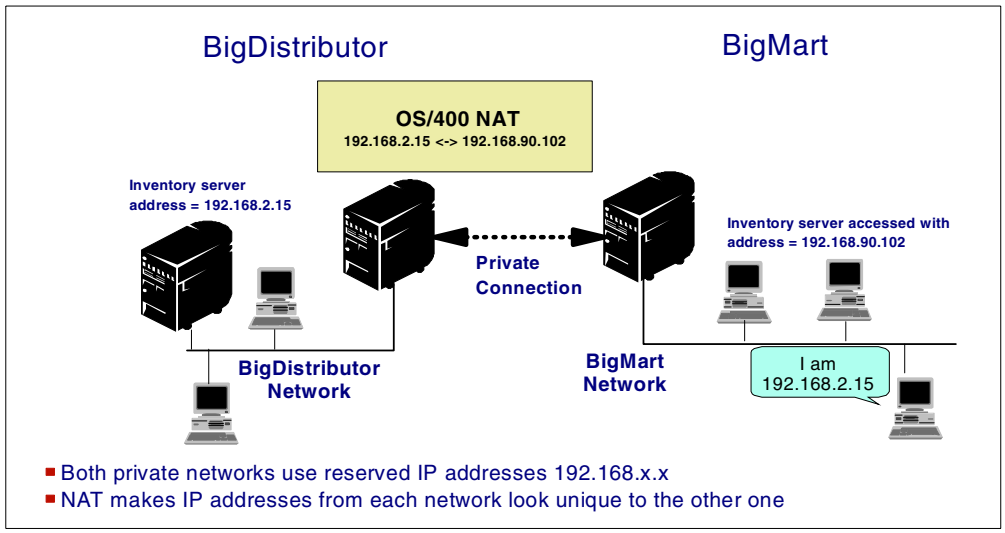


Figure 448. Connecting private networks with duplicate IP addresses using NAT

Another example is if Company ABC and company XYZ merge to form company AX. Company ABC uses IP addresses from the 10.x.x.x reserved range while company XYZ uses registered (for example, real) IP addresses. The two individual networks are to be connected via a private connection to provide clients in the former XYZ company access to ordering and distribution applications in the former ABC company. To avoid major changes in network addressing and routing, NAT is used to give the ordering and distribution systems registered IP addresses in the XYZ network that are mapped to 10.x.x.x. addresses by an AS/400 system acting as a gateway between the two networks.

The Network Address Translation (NAT) functions of OS/400 should also be considered when it is desirable to hide addresses in a subnetwork for which the AS/400 is performing routing.

10.1.1.5 Hiding subnetwork information using NAT

Company ABC (Figure 449) has an internal organization that performs top secret research. The research organization has its own private subnetwork, and systems on this subnetwork should only be used by research personnel. Information about this network is to be kept private from the rest of the corporate network. An AS/400 system is configured as a gateway between the research network and the rest of the corporate network. The OS/400 NAT functions are used to hide the addresses of the client systems in the research network when accessing systems outside the research network. All client requests from the research network are translated to a single IP address when accessing systems in the corporate network.

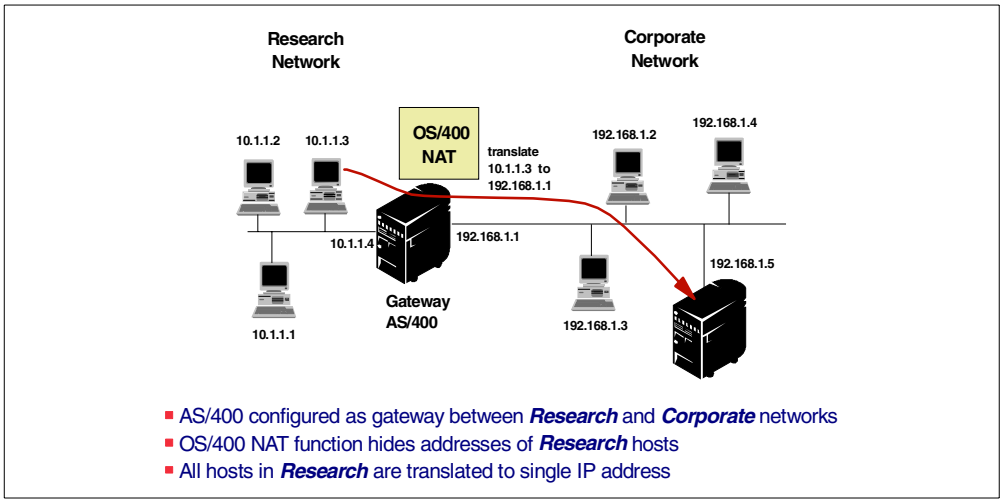


Figure 449. Hiding subnetwork information using NAT

10.1.1.6 Protecting connections using NAT and IP Packet Filtering

LittleCo Inc. (Figure 450) has a small network consisting of five PCs on an Ethernet LAN and an AS/400 system. LittleCo wants to have limited access to the Internet for Web browsing and e-mail retrieval. LittleCo decides to use a 56 Kbps PPP dial connection to an Internet Service Provider (ISP) from their AS/400 system. The ISP provides SMTP/POP3 support, a Web hosting service, and Domain Name Services. LittleCo uses NAT to allow the PCs in their network to access the Internet through their AS/400 system while hiding their IP addresses from servers on the Internet. The use of packet filtering, in conjunction with NAT, provides an effective barrier to incoming connections from the Internet.

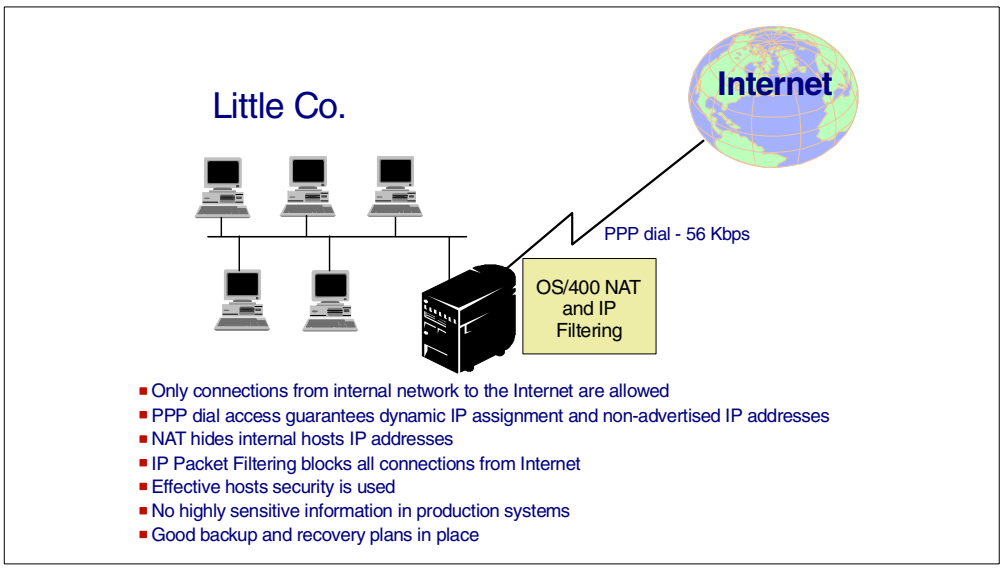


Figure 450. Protecting Internet connections using NAT and IP Packet Filtering

LittleCo understands that there is greater risk associated with their Internet connection because they are not employing a firewall. They feel the risk is acceptable because:

- PPP dial access is used which results in an IP address being dynamically assigned. This IP address is not “advertised” in any Domain Name Server (DNS).
- The ISP is dialed only during normal working hours when the system is monitored.
- Network Address Translation is used to hide all PC addresses.
- IP Packet Filtering is used to block all connections from the Internet.
- The connection speed to the ISP is relatively low, limiting the CPU resources that could potentially be used during packet filtering.
- Effective host security is employed (good passwords, object security...).
- Their AS/400 system does not contain any highly sensitive information.

Note: LittleCo's security requirements are different from that of many other companies. In general, an AS/400 system running production applications should *not* be directly connected to the Internet even though OS/400 packet filtering and NAT are used.

10.2 Network Address Translation (NAT)

Network Address Translation (NAT) lets you translate internal IP addresses to external IP addresses. This section provides an introduction to the NAT function and a technical description of how NAT works.

10.2.1 Introduction

Most of the organizations today have started implementing private networks. These networks may be unregistered networks without proper authentication and permission of any international organization (IANA) and work fine as intranets, as long as they are isolated from the rest of the world. This practice can have serious repercussions. Once we are out on the Internet, we may be using an address range owned and registered by others. Figure 451 shows a similar situation. This organization has implemented an excellent network that works fine while they are restricted to their own networks. Everything seems to be in-place, but it’s not.

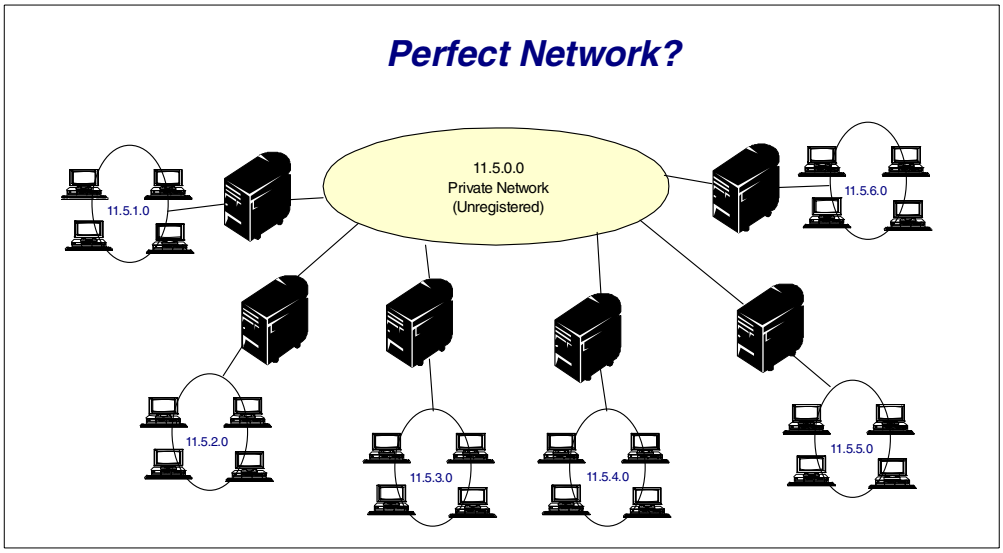


Figure 451. The perfect internal network

Figure 452 shows the problem that may occur once we go out on the Internet. Since we are using an address range that is owned and registered by someone else, we are not able to operate with the current IP setup we have. Does this mean that we have to redesign our network and start from the point when we started to build our company's intranet? That effort involved tremendous work. Is there another way of going to the Internet without redesigning our network and re-working everything that has proven its creditability over several years?

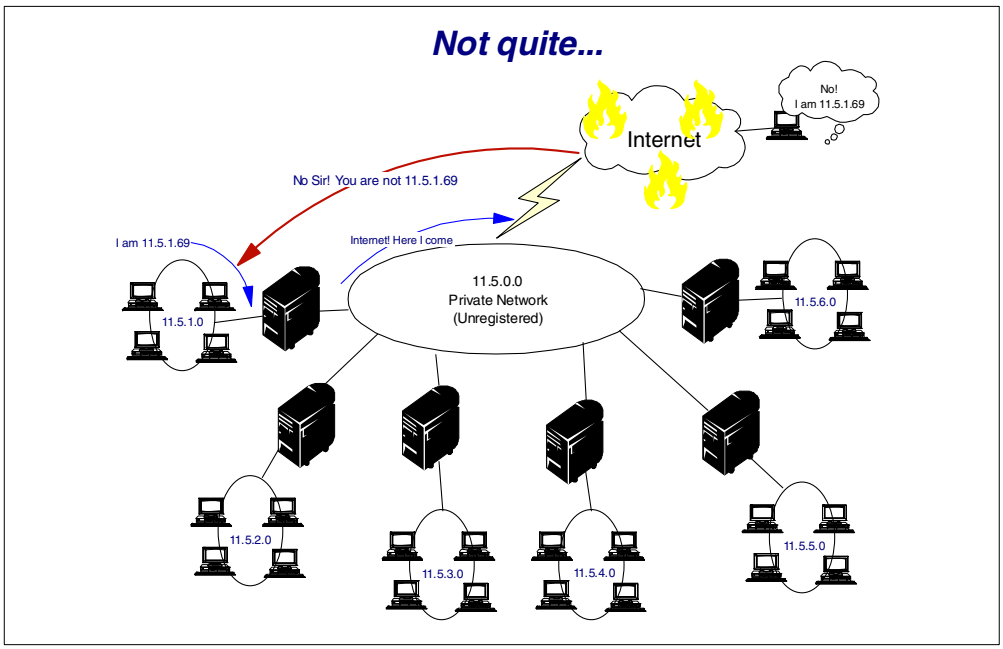


Figure 452. The not-so-perfect-anymore network

OS/400 V4R3 introduces the concept of NAT and IP Packet Filtering. This support will be provided as a part of the base operating system and will serve as an entry level firewall. This means that we will have a native IP translator for

Internet support and added security on the Internet. NAT will take care of the IP exhaustion problem. IP filtering will provide us the added security we need to shield our system from the outside world.

Network Address Translation addresses the problem of IP depletion. NAT allows a private unregistered network to be represented by a small set of valid registered addresses. This function resides on the IP routing device providing the connection between the two networks.

In addition to the IP depletion problem, NAT can also be used to provide a certain level of network security. When connecting to another network, it is highly advisable to isolate the topology of the internal private network. For example, the addressing (IP addresses, subnets and host names) of internal machines should not be known to the outside world. NAT can hide the internal network topology by representing the internal hosts behind a small subset of publicly known addresses.

Network Address Translation is a mechanism that can be used to allow a private network, which is not currently using a registered address, to connect to and communicate with a public, registered network. In all cases, the machine performing the translation function is an IP forwarding device that is connected to at least two different IP networks. It is always a boundary machine.

10.2.2 NAT methods

The NAT function can be used in different ways, depending on the situation and the requirements:

- Masquerading
- Dynamic
- Static
- Round robin

OS/400 V4R3 and V4R4 implements the masquerading and the static translation methods.

Firewall for AS/400 implements dynamic NAT and static NAT.

10.2.2.1 Masquerading

Masquerading is used to allow the private network to hide behind and be represented by the address bound to the public interface of the NAT machine. In most situations, this is the address that has been assigned by an ISP, which may be dynamic in the case of a PPP connection. This type of translation can only be used for connections originating within the private network destined for the outside public network. Each connection out is maintained by using a different source (client) IP port number. Refer to Figure 453.

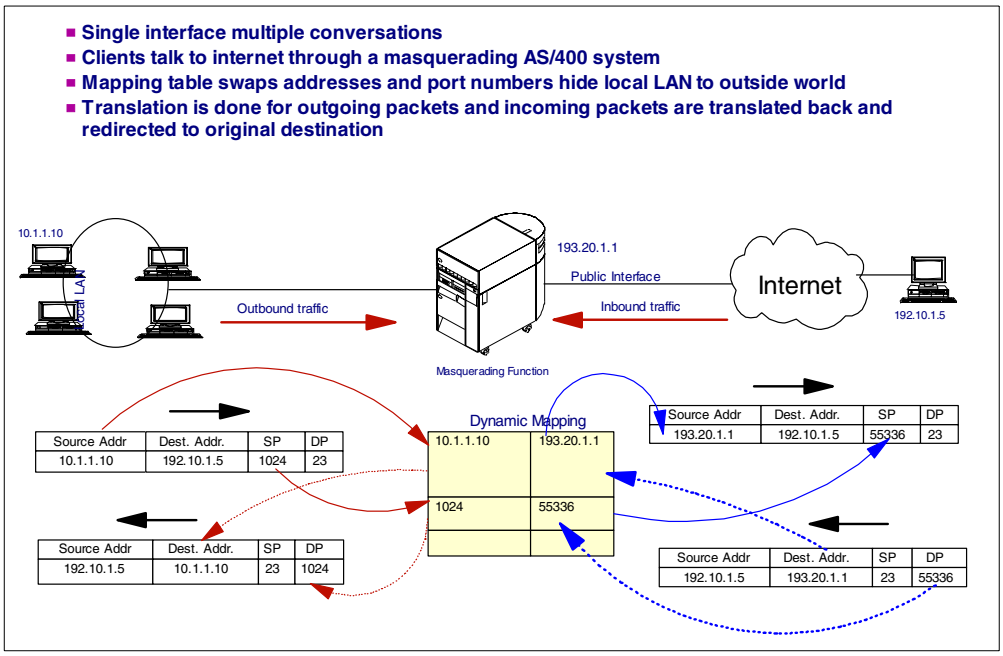


Figure 453. NAT Masquerading operation mechanism

10.2.2.2 Dynamic

Similar to Masquerading, dynamic NAT can only be used to establish connection from within the private network, out to the public network. The difference, however, is that a pool of public addresses is maintained and used when an outbound connection is made. Each connection is assigned a unique public address. The maximum number of simultaneous connections is equal to the number of public addresses in the pool. This is just like a one-to-one correspondence between addresses.

Dynamic NAT is not available in the native NAT implementation of AS/400 TCP/IP. It is available in Firewall for AS/400.

10.2.2.3 Static

Static NAT is a simple one-to-one mapping between private and public addresses. This is required to support inbound connections from the public network into the private network. For each local address defined, there must be an associated globally unique address. Refer to Figure 454 on page 368.

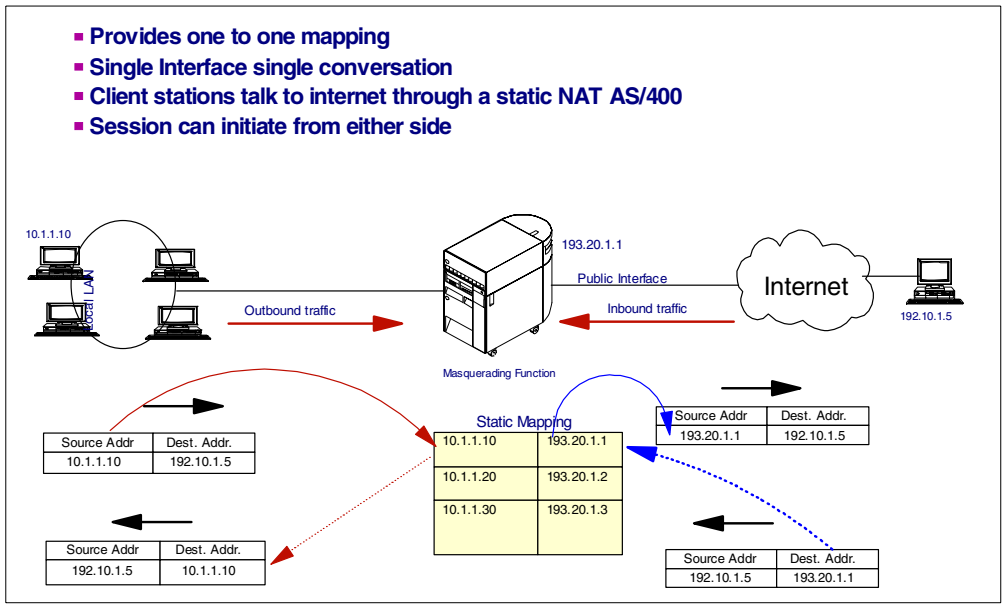


Figure 454. NAT Static operating mechanism

10.2.2.4 Round Robin

Round Robin NAT is used to distribute inbound IP connections for a single IP address to a pool of replicated machines. This allows the NAT machine to act as an IP load balancer for heavily accessed machines. This is similar to Round Robin DNS. However, it works at the IP layer, rather than the application layer.

Round Robin NAT is not available in the native NAT implementation of AS/400 TCP/IP. It is available in some brands of routers.

10.2.3 The technical details

This section gives a detailed description of the NAT masquerading and NAT static principles.

10.2.3.1 NAT masquerading principles

The following list applies to the NAT masquerading technique:

- Supports all protocols above the IP layer, including TCP and UDP
- Provides many-to-one mapping of IP addresses
- Swaps actual local ports with dummy ports and local addresses with the public interface address
- Recomputes checksums for TCP and UP packet headers
- Maintains tables of outgoing and incoming traffic
- Connection initiated by locally attached workstations only:
  - Provides a kind of native firewall, since the external network knows nothing of your existence
  - Hosts at the internal network cannot advertise their addresses
- The range of port numbers for masquerading are dynamically assigned

IP Masquerading is a method by which workstations which do not have Internet Network Information Center (InterNIC) registered IP addresses may communicate with the Internet through a router that has an InterNIC registered IP address. The term “locally attached machine” is used to refer to all machines on an internal network, regardless of the method of attachment (LAN, WAN) or the distance of the connection. The term “external machines” is used to refer to machines located on the Internet. Masquerade supports the TCP, ICMP, and UDP protocols.

To the Internet, all of the workstations appear to be contained within the AS/400 system. That is, there is only one IP address associated with both the AS/400 and the workstations. When the router receives a packet intended for a workstation, it attempts to determine what address on the internal ring should receive the packet and sends it there.

Each workstation must be set up so that the AS/400 system is its gateway, and also its default destination. The correspondence between a particular communication connection (port) and a workstation is set up when one of the workstations sends a packet to the AS/400 system to be sent to the Internet. The Masquerade NAT function saves the port number so that, when it receives responses to the workstations packet over that connection, it can send the response to the correct workstation.

A record of active port connections and the last access time by either end of the connection is created or maintained by Masquerade NAT. These records are periodically purged of all connections idle for a predetermined amount of time based upon the assumption that an idle link is no longer in use.

It should be noted that all communication between the workstation and the Internet must be initiated by locally attached machines. This is an effective security firewall. The Internet knows nothing of the existence of the workstations, and they cannot broadcast their addresses to the Internet.

A key to Masquerade NAT implementation is the use of logical ports, issued by masquerade to distinguish between the various communication streams. Standard transport layer protocols (UDP and TCP) contain a source and a destination port number. To these designations, NAT adds another port number, a logical port number (LPN).

Masquerade NAT outbound processing steps

An outbound message (local to external) contains the source port used by the originating workstation. NAT saves this number and replaces it in the transport header with a unique local port number. For outbound datagrams, the source port number is the local port number. The process involves this series of steps:

1. Routing is done before NAT for outbound traffic. Therefore, outbound masquerade processing assumes all IP packets it receives are bound for external IP addresses and does not check to determine if a packet should be routed locally.
2. The set of in-use LPN entry is searched looking for a match on transport layer and source IP address and source port. If found, the corresponding LPN is substituted for the source port. If no active matching LPN entry is found, a new one is created, a new LPN selected and substituted for the source port.

3. The source IP address is translated.
4. The packet is then processed as usual by the IP, and is sent to the correct external system.

Inbound masquerade processing (response and other)
For inbound datagrams, the destination port number is the local port number. For inbound messages, the source port number is the external port number. For outbound messages, the destination port number is the external port number.

Response messages returning from the Internet bound for a locally attached machine will have a masquerade-assigned logical port number (LPN) as the destination port number in the transport layer header. The Masquerade NAT inbound processing steps are:

1. Masquerade searches its database for this LPN (destination port). If not found, the packet is assumed to be a unsolicited packet, and the packet is *returned* to the caller unchanged. It is then handled as a normal unknown destination.
2. If a matching LPN is found, a further check is made to determine that destination IP address matches the source IP address of the existing LPN table entry. If it is OK, the original local machine's port number replaces the destination port in the IP header. If the check fails, the packets returned unchanged.
3. The local matching IP addresses will be put in the packet IP destination.
4. The packet is then processed as usual by IP, UDP, TCP, etc., and ends up at the correct locally attached machine.

Because masquerade requires an LPN to determine the correct source and destination port addresses, masquerade is incapable of handling unsolicited datagrams from the Internet.

10.2.3.2 NAT masquerading principles: An example

This example shows the working of the NAT masquerading function (Figure 453 on page 367). A client “10.1.1.10” wants to talk to a host on the Internet with the IP address of “192.10.1.5”. We are using a masquerading function AS/400 system, which can do the address translation. The local host is trying to connect to the remote host through local port 1024 onto the remote port 23 (a Telnet session). The request is taken up by the AS/400 system, which translates the IP address of the local host with the public interface address and swaps the local port number with a random number out of the local pool of addresses. An entry is made in its dynamic mapping table for this translation so that, when an inbound request is received, the AS/400 system should be able to route it to the exact destination. The packet arrives at the remote host with these changes, and the remote host sees it as the address of the actual machine trying to communicate with it. It responds to the same port number. The IP address, and the AS/400 system, upon receiving the request, reverse translate it. There will be an entry for every outbound communication request in the table maintained by the AS/400 system for each individual host. This mechanism provides for multiple conversations to multiple systems at the same time through one single interface address. The important point to note here is that only the client stations on the local LAN can initiate a connection to the outside world since the hosts exterior to this local LAN are absolutely unaware of its IP addresses, host names, and so on. This provides an excellent security feature.

In combination with filtering, you can ensure that only intended traffic gets through and the intended traffic will not use any internal network addresses. Generally, NAT is transparent to the communicating systems and their applications. Exceptions are protocols that put IP addresses with the data packet, like FTP and ICMP. These two specific exceptions *are* supported by AS/400 NAT.

10.2.3.3 NAT static principles: An example

Here we discuss the same example as that of Masquerading NAT (Figure 454 on page 368). The only difference is that this function provides one-to-one mapping. A static table entry is present for a pre-determined conversation that is to take place. The internal LAN is hidden behind a dummy address, but each physical interface is capable of carrying out only one conversation at a time. Since it is a static entry and an outside host will always request the same service, a session can be initiated from both ends. For multiple conversations, we need multiple interfaces. For each local address, there is a globally unique public address.

10.3 IP Packet Filtering

IP Packet Filtering support lets you explicitly control what IP traffic is allowed in your network. This section provides an introduction to IP Packet Filtering, as well as a technical description of the function.

10.3.1 Introduction

Figure 455 on page 372 gives a conceptual view of IP filtering. All the outbound and inbound IP packets have to pass through a set of rules. Whenever an outbound connection is requested, the packet is compared against a set of pre-defined rules. If a condition is met and it has permission to go out, it will be allowed to go out onto the Internet. Otherwise, the packet will be discarded. This mechanism is applied for inbound traffic as well. This provides a certain level of security from Internet traffic. IP filtering can also be used in intranets to secure intranets from unauthorized access internally. IP Packet Filtering, RIPv2, and NAT work together to form an entry level firewall.

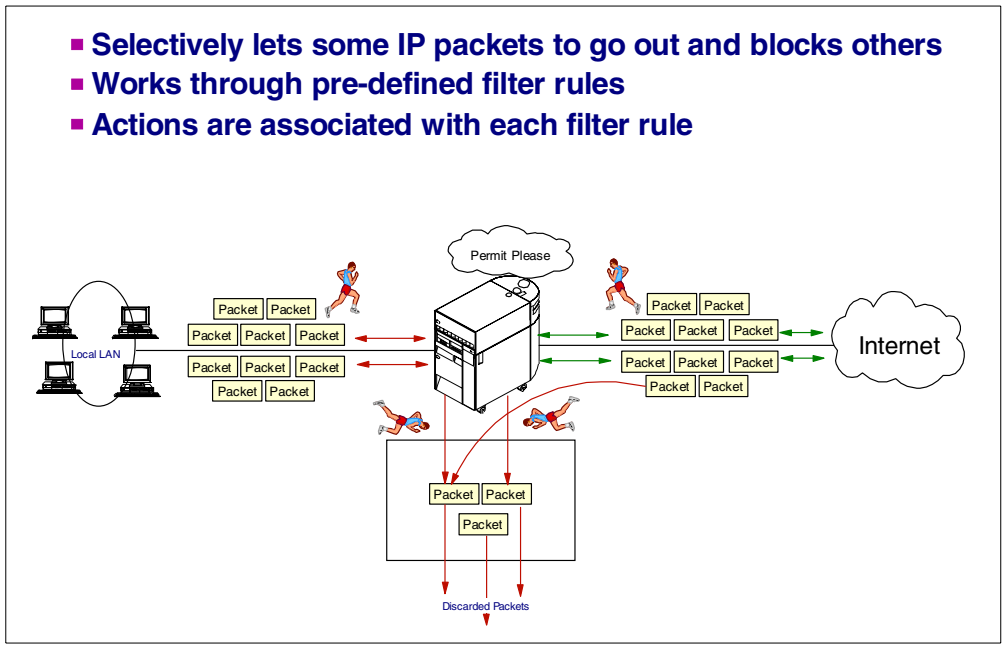


Figure 455. IP Packet Filtering principles

10.3.2 Technical details

The following list of terms is used in conjunction with IP Packet Filtering:

- Criteria
  - Source IP Address
  - Destination IP Address
  - Protocol (TCP, UDP, ICMP, or others)
  - Source Port
  - Destination Port
  - Direction (Inbound, outbound or both)
  - Physical Interfaces
  - Fragmented packets
  - TCP/IP originating or answering (where does the request start)
- Actions
  - Permit
  - Deny
- Scope
  - Specific
  - Universal

The availability of dynamic routing protocols and NAT allows the AS/400 system to easily connect two or more disparate networks to one another. However, these functions alone don't provide adequate security, even for low end, entry level connections. IP Packet Filtering is the core protection mechanism behind security. The combination of Routing, NAT, and IP filtering can be considered an entry-level network firewall.

Packet filters are set rules that limit IP packet flow into or out of a network. We define policies that determine which packets are allowed access into or out of the network. If there is no matching rule, the default rules are used to deny access

and discard the packets. We can filter packets based on the criteria discussed in the following text.

We can limit a specific source address for the outbound traffic to be restricted to the local network only, or we can scan for a restricted destination address. We can also restrict traffic for certain applications using a particular protocol, for example, TELNET using TCP protocol. We can restrict access to specific port numbers and all these rules apply for both inbound and outbound traffic.

We can set up filtering rules for our physical interfaces that will ensure a high level of security. There are two actions associated with the filtering rules. Either you allow or permit someone to enter your system, or you deny them access.

The default action on any physical interfaces that has one or more rules defined is “deny”, so that you must explicitly “permit” packets that you want to accept. This prevents accidental access from unwanted hosts.

The scope of the rules can be specific or it can be universal. You define specific rules for everything that you are aware of and want to prevent. Whenever an IP packet comes in, you check the contents of the IP packet against the rules that you set up for packets filtering. Normally, packets have predictable behavior and you have taken care of it. A rule table is maintained by the system which has all the entries that you define for your system. This table is scanned in the order in which filter rules are contained in the file for all the packets coming in or going out. Whenever a rule is matched, it is applied. If a general rule occurs in the file before a specific rule is found, the action on the general rule is applied. If, after scanning all the rules, there is no exact match, the default deny rule will cause the packet to be discarded.

10.3.3 The Internet Protocol (IP)

The IP suite is the primary means of organizing communications on the Internet. IP functions include:

- Defining the datagram (basic unit of transmission; also called a packet)
- Defining the Internet addressing scheme
- Routing datagrams to remote hosts
- Fragmenting and reassembling packets
- Moving data between the Network Access Layer and the Host-to-Host transport layer

10.3.4 Types of Internet Protocol (IP) communications protocols

The IP suite consists of several lower-level communications protocols:

- Internet Control Message Protocol (ICMP)
- Transmission Control Protocol (TCP)
- User Datagram Protocol (UDP)

10.3.4.1 Internet Control Message Protocol (ICMP)

The ICMP communicates errors and other information between hosts. The PING application makes use of the ICMP Echo and Echo reply functions to provide an easy way to discover whether an address is reachable in the network. ICMP is also used by network components, such as routers, to pass control information between them. ICMP provides information about transport problems, such as whether a host is unreachable or the sender is sending packets too fast.

The ICMP message consists of three control fields and the message data (Figure 456):

- The *Type* field describes the type of message that is contained in the ICMP datagram.
- The *Code* field contains the error code reported by the message.
- The *Checksum* field is generated based on the entire contents of the ICMP message.
- The message data contains the details of the message. In the case of a redirect message (type = 5), the message data contains the address of a new router to use.

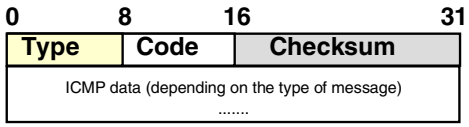


Figure 456. ICMP message format

You need to prevent most ICMP messages from entering your secure network because ICMP messages often provide a means for an attacker to access your network. An attacker can use PING, with its ability to use ICMP messages, to discover addresses in your secure network. An attacker can use re-route messages in an attempt to capture your data by re-routing your network traffic to an untrusted network.

For more information about these and other ICMP functions, see *Assigned Numbers* (RFC 1700).

10.3.4.2 Transmission Control Protocol (TCP)

TCP is the main transport layer protocol of the IP suite. Most IP applications, such as FTP, HTTP, TELNET, and SMTP, use TCP for a reliable end-to-end connection. TCP takes care of re-transmission, duplicate or lost packets, and the reordering of packets. For filtering purposes, the important TCP header information is:

- Source port
- Destination port
- Starting TCP packet flag

10.3.4.3 User Datagram Protocol (UDP)

User Datagram Protocol (UDP) is also a transport layer protocol, although TCP is used more often. Domain name services (DNS) and Simple Network Management Protocol (SNMP) use UDP.

UDP does *not* provide a reliable end-to-end connection. Unlike TCP, UDP does not handle re-transmission of packets, duplicate or lost packets, and re-ordering of packets. Once a packet is sent, the sender receives no confirmation that the packet reached its destination. Since UDP does not provide any acknowledgment information, it is difficult (and sometimes impossible) to tell if the UDP packet is a response to a request generated from the secure network or from the untrusted network.

10.3.4.4 Internet Protocol (IP) packets

An IP packet consists of a formatted header and the payload data. The header consists of fields that contain identifying data about the packet (Figure 457). The payload contains the actual information that is transmitted. The payload data may include an additional header that provides session level protocol information (for example, TCP, UDP, and so forth).

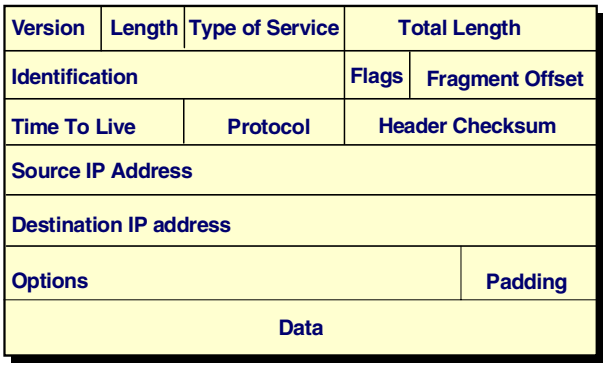


Figure 457. IP packet structure

The important fields for filtering purposes are:

- Source address
- Destination address
- Fragmentation indicator
- Protocol ID

The IP Packet Filtering function uses the source and destination address, together with the protocol ID, to define which packets may access which service.

Different types of networks support different sizes of packets. Consequently, a router sometimes must break a large packet into fragments to pass it from one network to another. The IP Packet Filtering function or receiving router must be aware of the fragmentation because only the first fragment contains the identifying header information for higher layer protocols such as UDP and TCP. Later fragments can override header fields, such as the source and destination address. The packet fragmentation indicator tells the IP Packet Filtering function how to handle fragmented packets. This allows attackers to use this technique as a way to infiltrate a network. Therefore, consider configuring IP Packet Filtering function to allow only non-fragmented packets. Refer to *Security Considerations for IP Fragment Filtering* (RFC 1858) for more information.

10.3.4.5 Transmission Control Protocol (TCP) packets

TCP is a reliable, connection-oriented protocol, which establishes a logical end-to-end connection between two hosts. TCP verifies that data is delivered across the network accurately and in the proper sequence. TCP verifies that a packet arrived at the remote host. If it does not, TCP re-transmits the packet. A TCP packet consists of a formatted header and the application data. The fields in the header contain identifying data about the packet (Figure 458 on page 376). The TCP packet is included in the data portion of the IP packet.

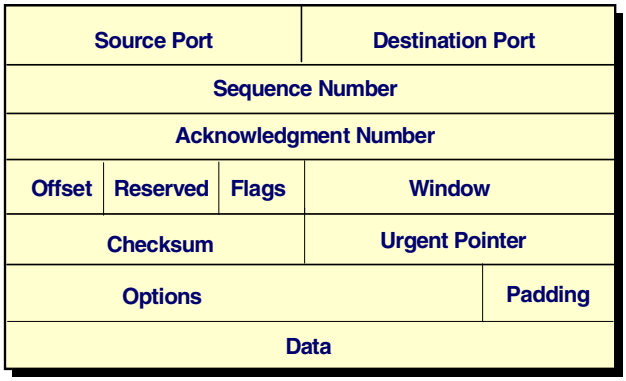


Figure 458. TCP packet structure

A TCP connection is uniquely defined by:

- Source address from the IP portion of the packet
- Source port from the TCP portion of the packet
- Destination address from the IP portion of the packet
- Destination port from the TCP portion of the packet

TCP uses the sequence number and the acknowledgment number (ACK) to keep track of the bytes. The acknowledgment segment performs two functions:

- Positive acknowledgment
- Flow control

The acknowledgment tells the sender how much data has been received and how much more the receiver can accept.

TCP is also responsible for delivering the data received from the IP to the correct application. The application is identified by a 16-bit number called the destination port number. The source and destination port are contained in the first word of the segment header.

The important fields for filtering purposes are:

- Source port
- Destination port
- Acknowledgment (ACK) flag

A TCP session is initiated by a three-way synchronization, which Figure 459 illustrates. Notice that the initial request to start a session does not contain an ACK flag. This feature can be useful for creating filter rules so that start requests from the untrusted network cannot enter your internal secure network. The IP Packet Filtering function uses the term *TCP/STARTING* for defining TCP packets without an ACK flag.

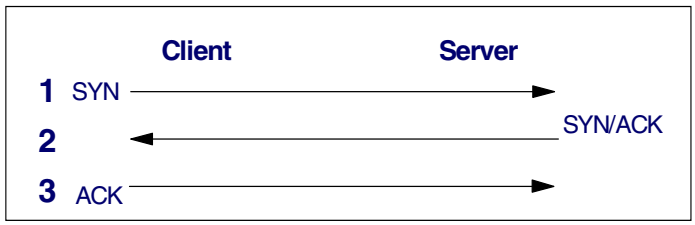


Figure 459. TCP session synchronization

For example, say you want to allow users on the secure network to start an e-mail session with a server on the untrusted network through port 25. You also want to permit your internal users to receive responses from port 25. You can create two filter rules that allow this traffic. However, you do *not* want to permit start requests from port 25 to access your internal network. To block start requests on port 25 from the untrusted network, you must ensure that the filter rules deny inbound packets that do not contain the ACK flag.

10.3.5 Internet Protocol (IP) forwarding

IP forwarding takes packets from one interface to another interface on the same system. The IP Packet Filtering function allows forwarding of only packets that pass the filter rules. You must use IP forwarding when you want the AS/400 system to act as a router. Use IP forwarding with caution. This exposes your internal network to a substantial risk, because an attacker can possibly exploit any holes in your filtering rules to access your internal network.

10.3.6 Well-known ports

Each Internet application (for example, Telnet) uses IP to send communications from a client port to a well-known port on a server (Figure 460). Intruders often try to sneak into a secure network by checking whether they can gain access through obscure, little-used ports. If you configure your Internet applications to use only their associated well-known ports, you can create filter rules to block communications that deviate from this usage.

Service	Port # / Protocol
SMTP	25/tcp
POP v3	110/tcp
Ident Request	113/tcp
TELNET	23/tcp
FTP-data	20/tcp
FTP	21/tcp
DNS	53/tcp 53/udp
Gopher	70/tcp
WWW--HTTP	80/tcp
WWW--HTTPS	443/tcp
IRC	6xxx/tcp
SOCKS	1080/tcp

Figure 460. Well-known ports for common Internet applications

Figure 460 on page 377 contains a list of well-known ports for common Internet applications. For a complete list of well-known ports, refer to *RFC 1700*, *Assigned Numbers*.

10.4 Where and when NAT and IP Packet Filtering is done

The NAT IP Packet Filtering takes place in the IP layer of the TCP/IP protocol as shown in Figure 461. The functions enables the selection of the TCP, UDP, and ICMP protocols. The ICMP protocol is located at the same level as the IP layer and uses the IP datagram to send and receive messages.

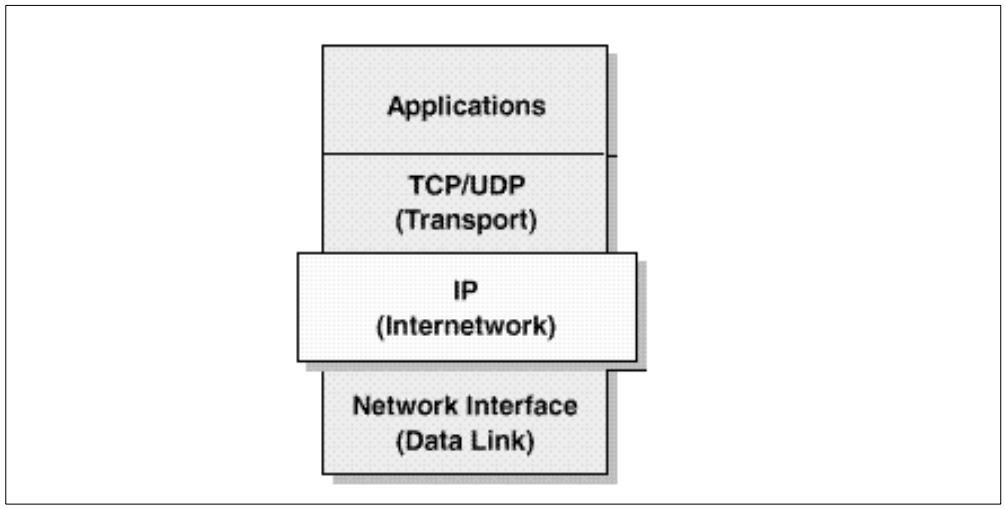


Figure 461. Locating the NAT and IP Packet Filtering functions

The actual order of processing of the NAT and IP Packet Filter functions is shown in Figure 462. IP datagrams arriving at an interface (inbound) are first processed by the NAT function and afterwards by the IP Packet Filter rules. IP datagrams destined for other hosts (outbound) are first processed by the filter rules and finally processed by the NAT function.

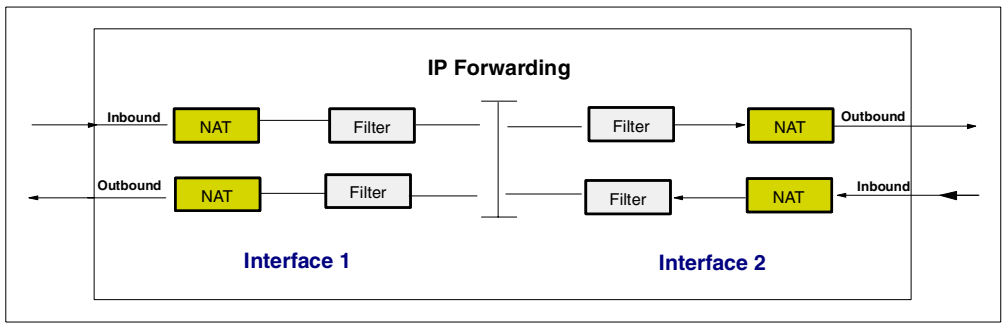


Figure 462. Packet flow through the NAT and IP Packet Filtering functions of OS/400

Table 16 summarizes the process order.

Table 16. Order process summary

Direction	Function
Inbound	1. NAT 2. Filter rules
Outbound	1. Filter rules 2. NAT

10.5 Planning IP Packet Security

The following list is an overview of the planning process you need to go though before implementing IP Packet Security:

1. Get an overview of the network, routers, and IP addresses used in the network. If necessary, make a drawing of the complete network.
2. Identify what IP traffic is allowed in the network.
3. Create filter rules and NAT mappings. Remember to only include wanted traffic. Other traffic is excluded.
4. Create one filter role at a time, for example the Telnet rules, and test if the result is what you expected. This one-step-at-a-time way of implementing the IP Packet Security makes it easier to test your rules.
5. The IP Packet Security function automatically applies a default-deny role on any interface that has at least one filter rule applied to it. This effectively stops any packets not matching a filter rule.
6. NAT and filter rules are applied to specific physical interfaces. Carefully select which lines should have which rules. Some lines can have no rules, while other lines have rules.
7. The specified filter rules must be validated before they are applied to the system. Use the Verify function.
8. Activate the rules on your system.
9. You should make a backup of the filter rules.
10. You should print a copy of the filter rules.
11. If, for some reason, your filter rules are messing up your system (for example, the connection from Operations Navigator is denied), you can use the Remove TCP/IP Table (`RMVTCPTBL`) command to remove all IP Packet Security settings.

10.6 Managing NAT and IP Packet Filtering

The administration and use of the NAT and the IP Packet Filter functions is solely performed by the use of Operations Navigator. This section describes the IP Packet Security functions of Operations Navigator, save/restore considerations, and how to monitor the activities related to NAT and IP Packet Filtering.

10.6.1 Using Operations Navigator

This section gives you an overview of the use of Operations Navigator and the terms used in the IP Packet Security function.

10.6.1.1 Starting the IP Packet Security function

The IP Packet Security functions can be started using the Operations Navigator. Follow these steps:

1. Start Operations Navigator by clicking **Start->Programs->IBM AS400 Client Access->AS/400 Operations Navigator**. The AS/400 Operations Navigator window appears (Figure 463).
2. Double-click on the AS/400 Systems icon (A).
3. Double-click the system icon (B) for the AS/400 system that you are configuring. The system components appear.
4. Double-click the **Network** icon (C). The network components appear.
5. Double-click the **Protocols** icon (D). The available protocols appear.
6. Right-click the **TCP/IP** icon. The pop-up menu appears.
7. Select the **IP Packet Security** menu item.

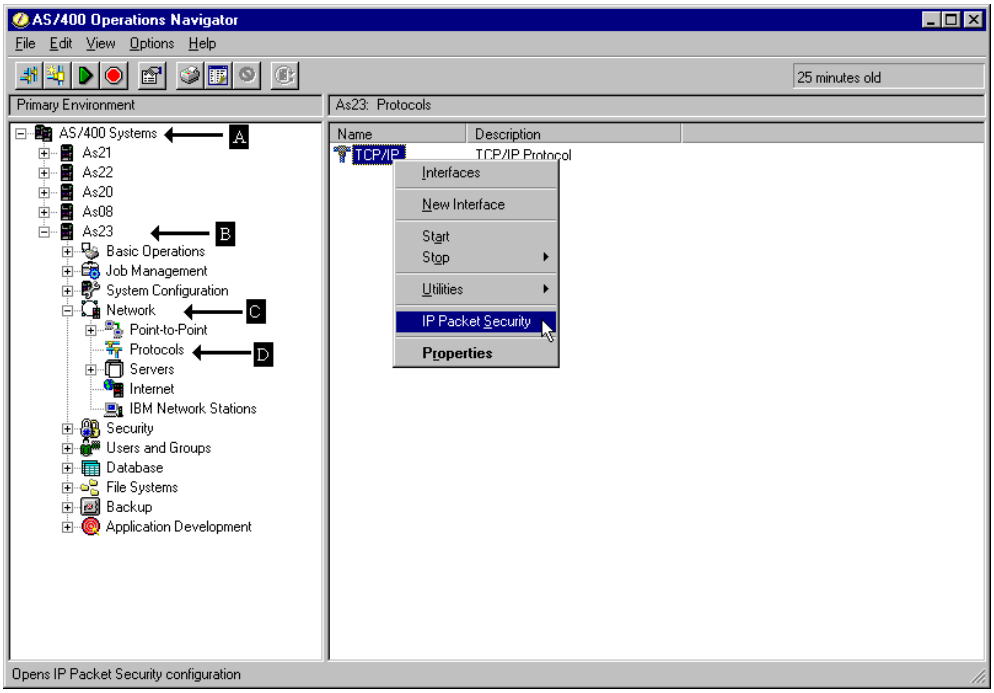


Figure 463. Starting the IP Packet Security function from Operations Navigator

The window in Figure 464 appears. This is the main window that offers an overview of the IP Packet Security functions.

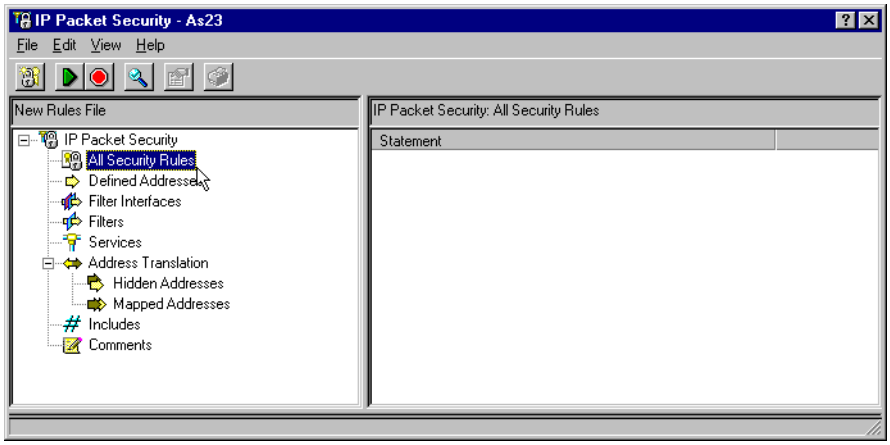


Figure 464. The IP Packet Security window

From here, you define the security rules. The terms used in the left part of the window are described in the next section.

When you create IP Packet Security rules, they are stored in a file in the OS/400 IFS. You can create many different IP Packet Security rules. Use a new file name for the rules.

10.6.1.2 IP Packet Security terms

The IP Packet Security function uses many different terms. These are shown in the left part of the window of Figure 464. Unfortunately, the window does not show the relationship between the different terms. Figure 465 helps you to understand the different terms and their relationship. A description of the different terms used follows the figure.

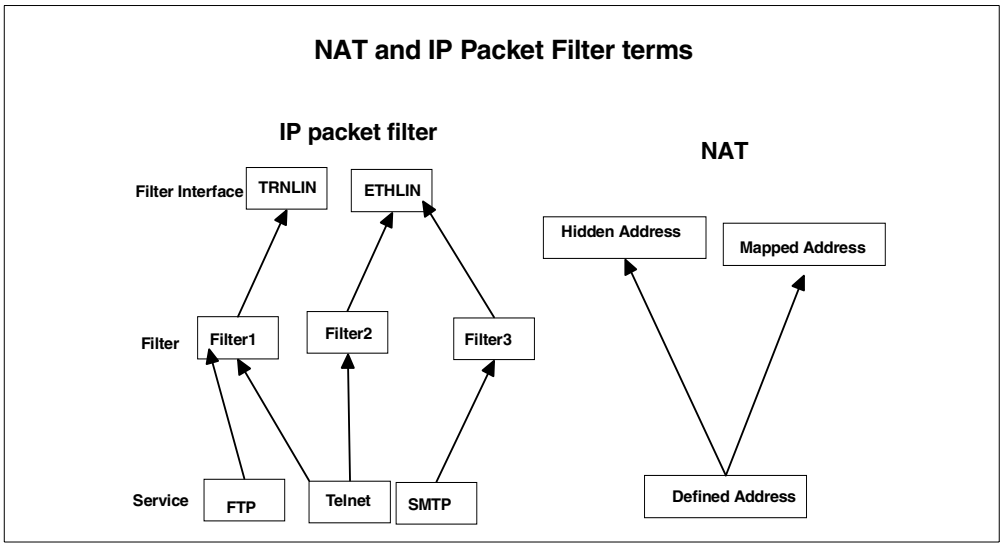


Figure 465. Overview of the IP Packet Security terms

All Security Rules

This item is used to show any of the entries in the Rules File. You can use this to determine what rules (if any) are currently active on your system. See Figure 466 on page 382.

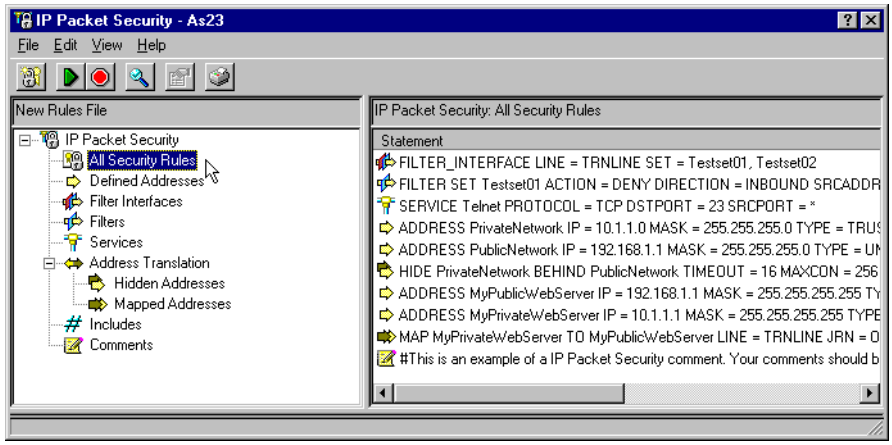


Figure 466. Displaying All Security Rules

Note

The order of the filter rules shown in Figure 466 is the order the rules are processed by the TCP/IP stack. The process starts at the top and continues until the first match is found. You can avoid testing a lot of rules if you put the rules that are hit the most at the top of the list. For example, if this system is used heavily for HTTP serving on port 80 and 443, you may want to put the permit rules for 80 and 443 near the top of the file to avoid testing many other rules first.

Defined addresses

This item is used to provide an alias for an IP address or a range of addresses. The alias is used by the NAT and IP filter rules function. Therefore you need to specify whether this alias is trusted (private) or untrusted (public). See Figure 467 and Figure 468.

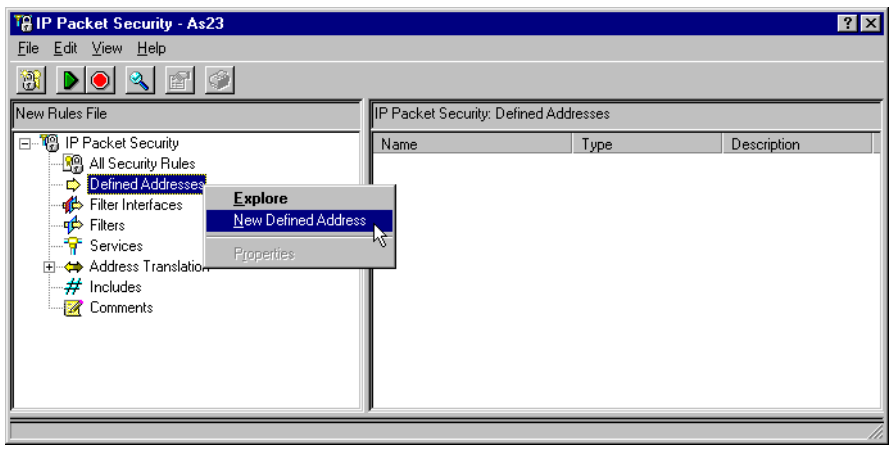


Figure 467. Creating a New Defined Address (Alias) (Part 1)

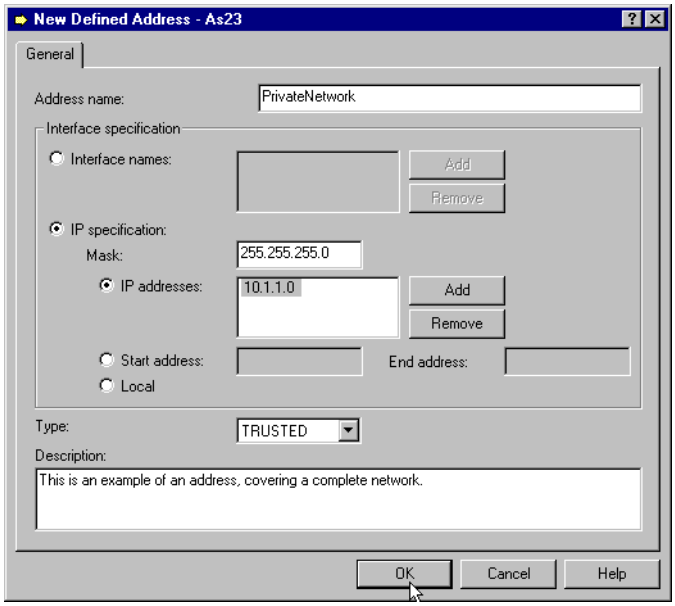


Figure 468. Creating a New Defined Address (Alias) (Part 2)

Filter interfaces

This term describes a link between a set of filter rules and a particular physical interface. The filters are described in “Filters” on page 384. You can associate several filter sets to one physical interface. See Figure 469 and Figure 470 on page 384.

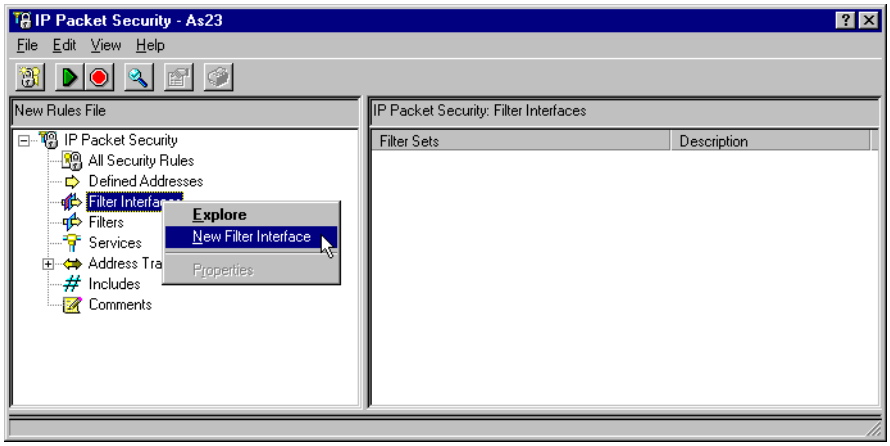


Figure 469. Creating filter interfaces (Part 1)

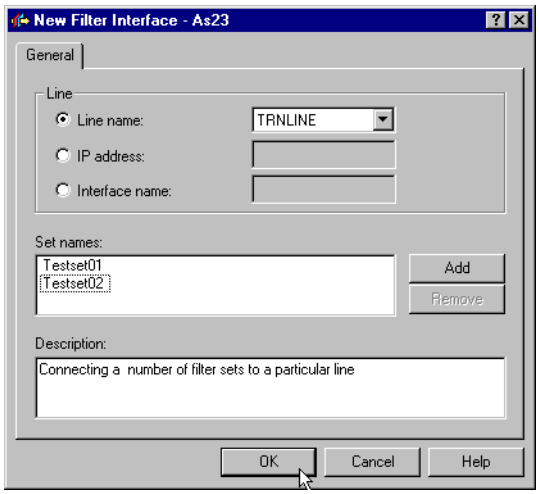


Figure 470. Creating filter interfaces (Part 2)

Filters

Filters let you define the actual selection criteria. You can choose if you want to permit or deny the particular criteria. You can also choose which IP addresses should be investigated, in what direction, whether you want to log a request or not etc. See Figure 471, Figure 472, and Figure 473. You supply the filter with a name. This name is linked to a particular interface, as shown in “Filter interfaces” on page 383. The filter is associated to a service. This service defines the TCP, UDP, and ICMP criteria. In the filter, you can refer to a service, or explicitly enter the criteria. See “Services” on page 385 for more information.

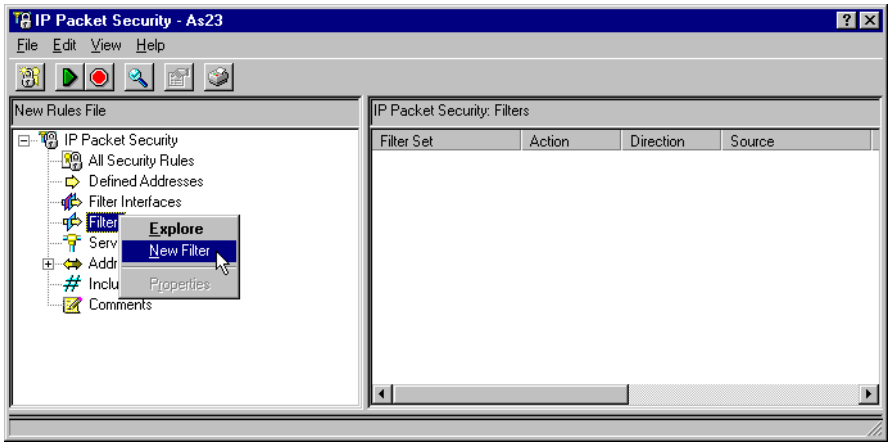


Figure 471. Creating filters (Part 1)

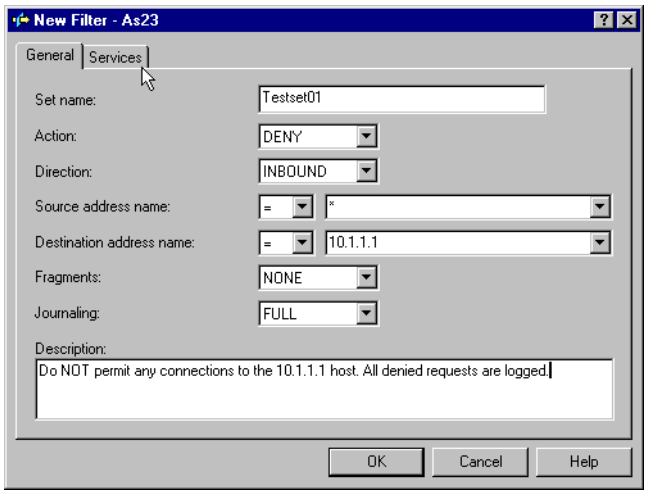


Figure 472. Creating filters (Part 2)

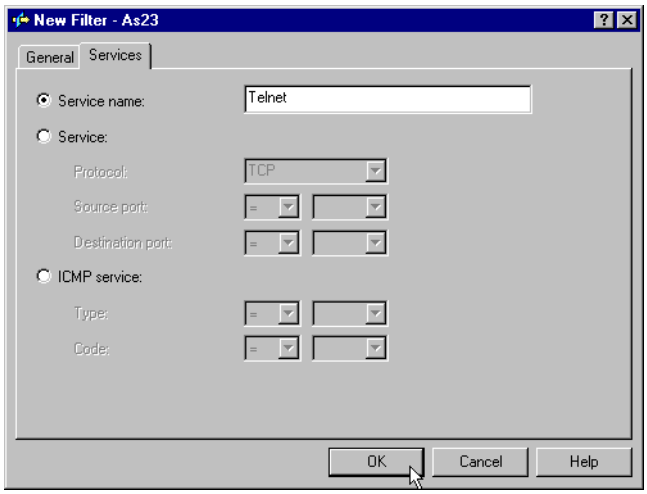


Figure 473. Creating filters (Part 3)

Services

Services define what TCP, UDP, and ICMP criteria you want to select. You specify the source and destination ports you want.

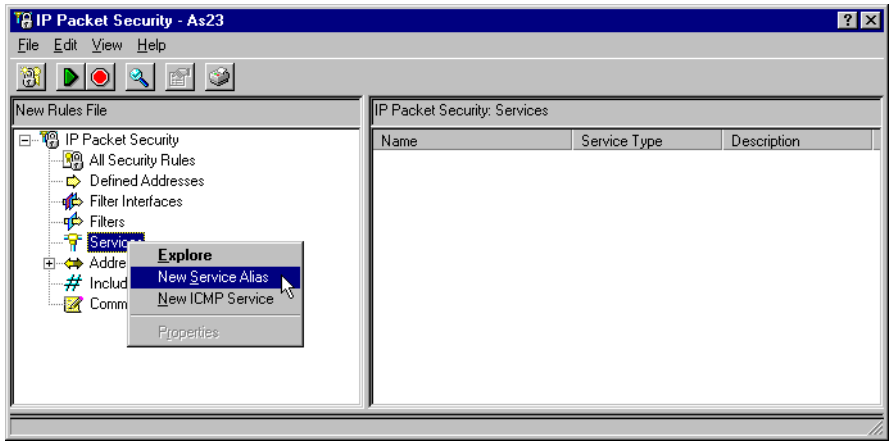


Figure 474. Creating services (Part 1)

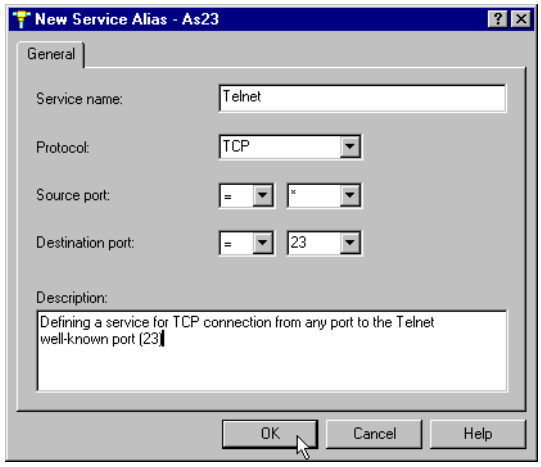


Figure 475. Creating services (Part 2)

Address Translation

Address Translation is a NAT function. Masquerading NAT is implemented using Hidden Addresses, and Static NAT is implemented using Mapped Addresses.

Both methods of NAT relate to a Defined Address (alias). See “Defined addresses” on page 382 for more information on Defined Addresses.

- **Hidden Addresses**

  Masquerading NAT is configured using the Hidden Addresses item. To create a new Hidden Address, you specify the name of the hidden address and the name of the public address. You also specify the maximum number of seconds NAT mapping is allowed to be idle before the mapping is removed. See Figure 476 and Figure 477.

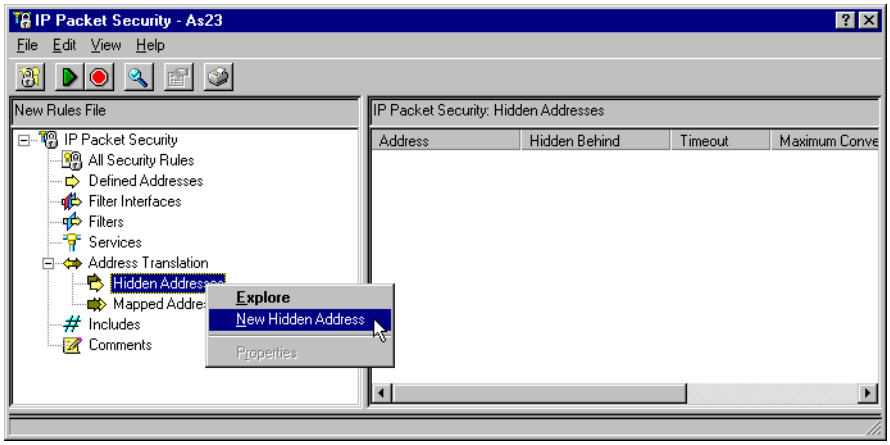


Figure 476. Creating a New Hidden Address (Part 1)

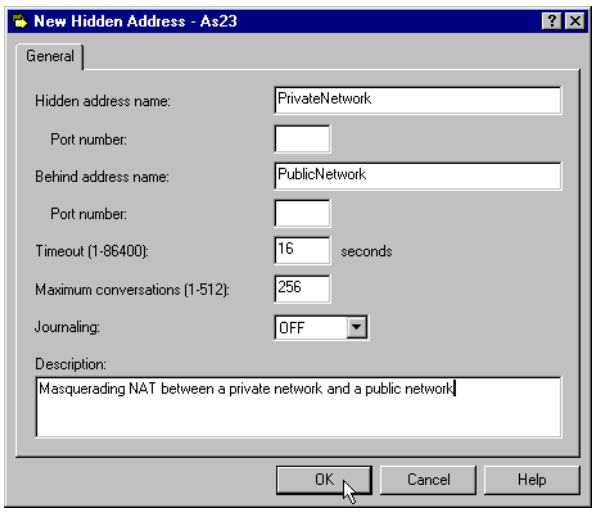


Figure 477. Creating a New Hidden Address (Part 2)

- **Mapped Addresses**

  Static NAT is configured using the Mapped Addresses item. To create a new Mapped Address, you specify the name of the internal address and the name of the public address. You specify the physical line with which this mapping is associated. Please note that you do not specify any port numbers. See Figure 478 and Figure 479 on page 388.

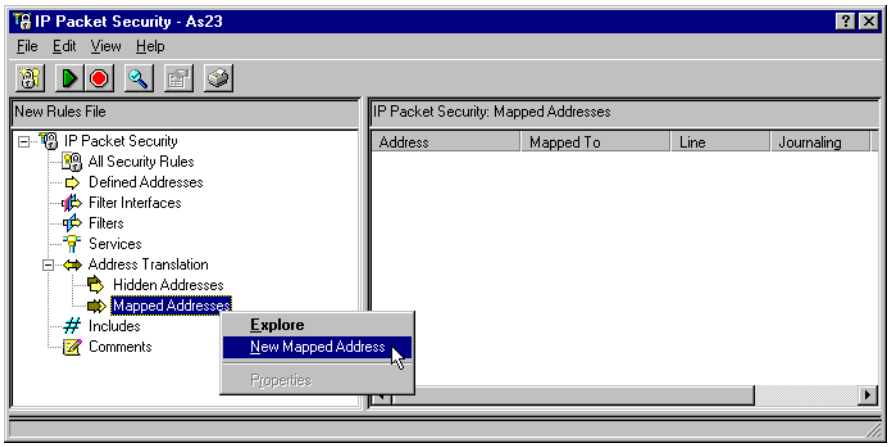


Figure 478. Creating a New Mapped Address (Part 1)

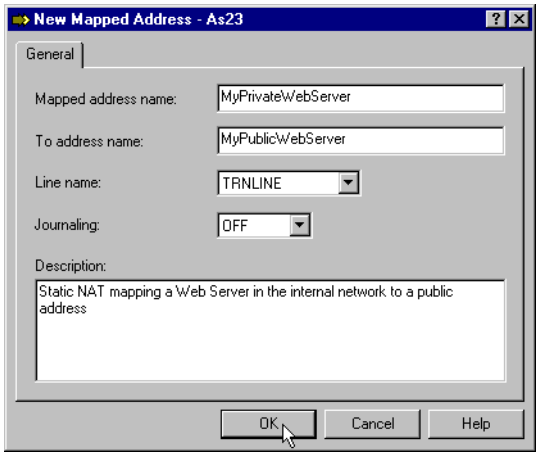


Figure 479. Creating a New Mapped Address (Part 2)

Includes

This feature enables you to save lots of time. Includes work like includes do in programing languages. You can specify the name of any IP Packet Security file (*.I3P) to include. This way, you can include standard rules for Services, Defined Addresses, etc. You create the rules once, test them, and reuse them in later configurations.

One use of includes is in a multiple system environment. You define system specific information like IP addresses in one file and define the set of rules to allow functions in another set of files. You use the same alias names on all the systems in the network with different IP addresses in the alias. You can copy the general rules files to all the systems. You include the general rules file and the system specific file. This allows you to move the same rules around to all the systems.

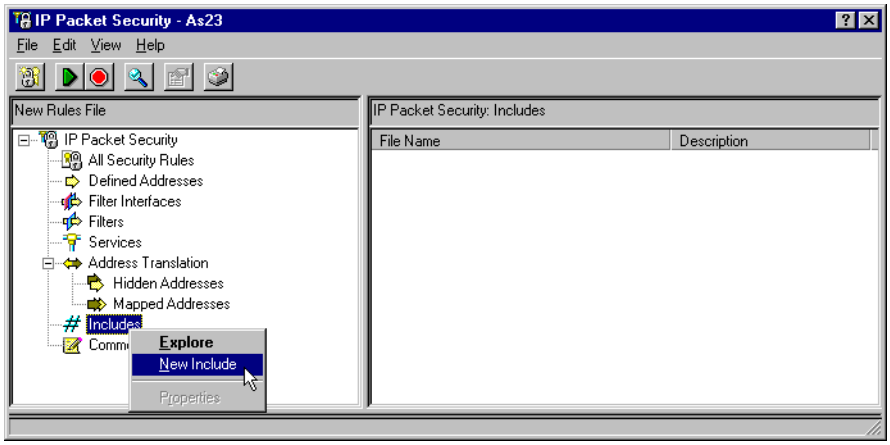


Figure 480. Creating a New Include (Part 1)

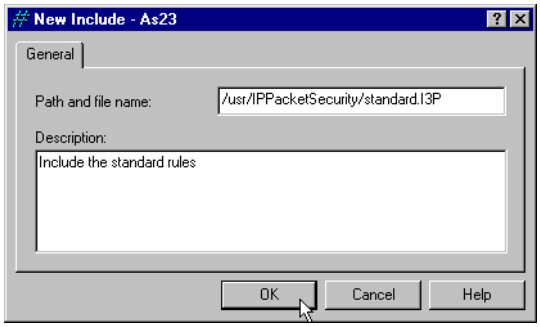


Figure 481. Creating a New Include (Part 2)

Comments

You can add comments to your rules file. This is quite useful, especially if you need to reconfigure rules you created eight months ago and have problems remembering the exact configuration.

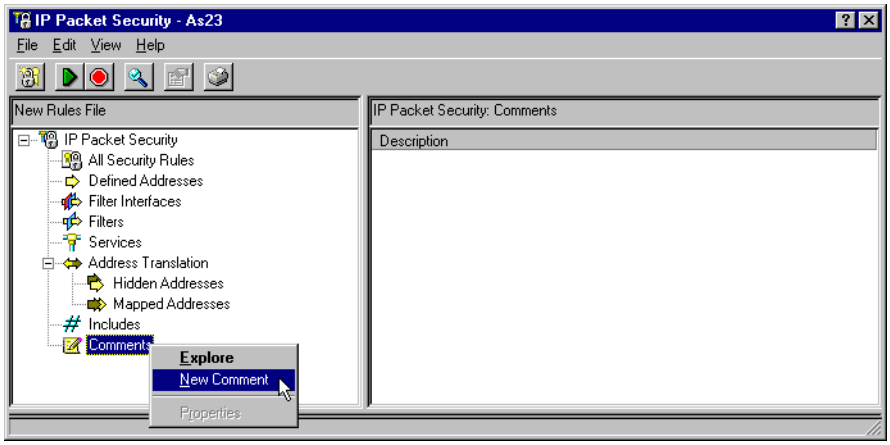


Figure 482. Creating a New Comment (Part 1)

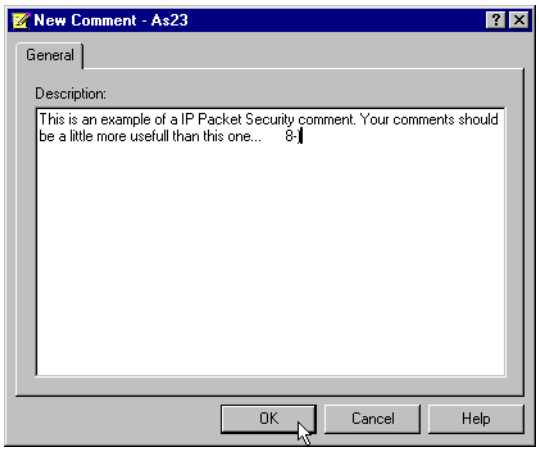


Figure 483. Creating a New Comment (Part 2)

10.6.1.3 Saving a rules file

After all the rules have been created, the file should be saved. Using the Save As... item of the File menu allows you to save the rules file to the AS/400 IFS.

10.6.1.4 Activating and deactivating a rules file

To activate the current rules file, select the **Activate** item from the File menu. The rules are verified and, if found valid, activated on the system.

If the rules contain any errors, they can be viewed on the bottom of the window (Figure 484).

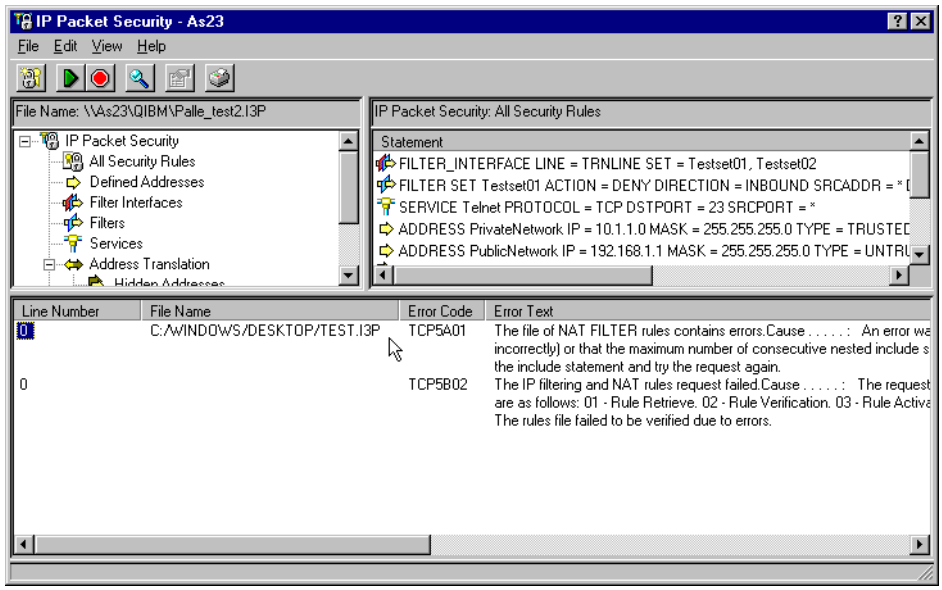


Figure 484. Viewing the IP Packet Security rules errors

An active IP Packet Security Rules File can be deactivated by using the Deactivate item in the File menu.

10.6.2 Backup and restore considerations

Usually, you save the IP Packet Security rules files to the OS/400 IFS file system. You should, at regular intervals, make sure that you back up the IFS directory containing the rules files. Use the Save Object (SAV) and Restore Object (RST) commands to save and restore the files from the IFS.

Before you change any Rules File, you should create a backup of the currently used file. You could develop a naming standard, for example *SYSTEMNAME*nnnn, where *SYSTEMNAME* is the name of the system this file applies to and *nnnn* is an integer. Each time you are about to make changes to the current file, save the new file, increasing the number. This way you have a history of your changes.

You can also use the Comments section of the IP Packet Security function to document your changes.

10.6.3 Monitoring NAT and IP Packet Filtering

The activities performed by the NAT and IP Packet Filtering function can be monitored using standard OS/400 journal support. The following two journals are used:

- QUSRSYS/QIPFILTER for IP Packet Filter journaling
- QUSRSYS/QIPNAT for NAT journaling

Note

Make sure that you have the SF49088 PTF installed on your system before using any of the journalling functions.

The two journals are created automatically the first time you activate IP Packet Filtering or NAT rules that request journaling.

The level of logging depends of the configuration of the IP Packet Security functions. Logging is typically used during testing and initial configuration. After this period, you can define the logging functions only to include requests to access the system that are denied.

To view the entry-specific details in the journals, you can display the raw journal entries on the screen, or you can use the following two system supplied out files:

- QSYS/QATOFIPF for the IP Packet Filter entries
- QSYS/QATOFNAT for the NAT entries

Figure 485 on page 392 and Figure 486 on page 393 show examples from the QIPFILTER journal of a packet being denied.

By copying the journal entries to the out files, you can easily view the entries using query utilities, such as Query/400 or SQL. You can also write your own HLL programs to process the entries in the out files.

The following example shows the Display Journal (DSPJRN) command:

```
DSPJRN JRN(QIPFILTER) JRNCDE((M)) ENTTYP((TF) OUTPUT(*OUTFILE)
OUTFILFMT(*TYPE4) OUTFILE(mylib/myfile) ENTDTALEN(*VARLEN *CALC)
```

Use the following procedure to copy the IP Packet Filtering journal entries to the out files:

1. Create a copy of the system supplied outputfile QSYS/QATOFIPF using the Create Duplicate Object (CRTDUPOBJ) command. Place the file in a user library.
2. Use the Display Journal (DSPJRN) command to display the entries from the QUSRSYS/QIPFILTER journal to the outputfile created in step 1.

Use the following procedure to copy the NAT journal entries to the out files:

1. Create a copy of the system supplied outputfile QSYS/QATOFNAT using the Create Duplicate Object (CRTDUPOBJ) command. Place the file in a user library.
2. Use the Display Journal (DSPJRN) command to display the entries from the QUSRSYS/QIPNAT journal to the outputfile created in step 1.

```
                          Display Journal Entries

Journal  . . . . . :   QIPFILTER      Library  . . . . . :   QUSRSYS

Type options, press Enter.
  5=Display entire entry


Opt    Sequence  Code  Type  Object      Library     Job         Time
             1   J     PR                            QTOFJRN     15:58:35
             2   M     TF                            QTOFJRN     16:13:30
             3   M     TF                            QTOFJRN     16:14:18
             4   M     TF                            QTOFJRN     16:14:21
             5   M     TF                            QTOFJRN     16:14:27
             6   M     TF                            QTOFJRN     16:14:40
             7   M     TF                            QTOFJRN     16:15:04
             8   M     TF                            QTOFJRN     16:15:54
             9   M     TF                            QTOFJRN     16:16:03
            10   M     TF                            QTOFJRN     16:16:06
            11   M     TF                            QTOFJRN     16:16:13
            12   M     TF                            QTOFJRN     16:16:25    +

F3=Exit   F12=Cancel
```

Figure 485. Displaying the QIPFILTER journal

```
                          Display Journal Entry

Object . . . . . :                  Library  . . . . . :
Member . . . . . :                  Sequence . . . . . :   3
Code . . . . . :   M  - Network management data
Type . . . . . :   TF - IP filter rules actions

           Entry specific data
Column      *...+....1....+....2....+....3....+....4....+....5
00001      'IPCSTRN02 A O    3DENY     610.1.223.1      102610'
00051      '.1.223.1       512                                '
00101      '                                                  '

                                                                      Bottom
Press Enter to continue.

F3=Exit   F6=Display only entry specific data
F10=Display only entry details   F12=Cancel   F24=More keys
```

Figure 486. Displaying the QIPFILTER journal: Entry specific data

Table 17 and Table 18 on page 394 show the fields in the out files and a description of the fields that not are self-explanatory.

Table 17. File layout for QSYS/QATOFIPF (IP Packet Filter)

Field name	Field length	Numeric	Description	Comments
TFENTL	5	Y	Length of entry	
TFSEQN	10	Y	Sequence number	
TFCODE	1	N	Journal code	Always “M”
TFENTT	2	N	Entry type	Always “TF”
TFTIME	26	N	SAA timestamp	
TFRES	95	N	Reserved area	
TFLINE	10	N	Line description	“*ALL” if TFREVT is “U*”. Blank if TFREVT is “L*”. Line name if TFREVT is “L”.
TFREVT	2	N	Rule Event	“L*” or “L” when rules are loaded. “U” when rules unloaded. “A” when filter action.
TFPDIR	1	N	IP Packet Direction	“O” is outbound. “I” is inbound.
TFRNUM	5	N	Rule Number	Applies to the rule number in the active rules file.
TFFACT	6	N	Filter Action Taken	“PERMIT” or “DENY”

Field name	Field length	Numeric	Description	Comments
TFPROT	4	N	Transport Protocol	1 is ICMP 6 is TCP 17 is UDP
TFSRCA	15	N	Source IP Address	
TFSRCP	5	N	Source Port	Garbage if TFPROT =1 (ICMP)
TFDSTA	15	N	Destination IP Address	
TFDSTP	5	N	Destination Port	Garbage if TFPROT =1 (ICMP)
TFTEXT	76	N	Additional Text	Contains description if TFRVET = “L*” or “L” or “U”

Table 18. File layout for QSYS/QATOFNAT (NAT)

Field name	Field length	Numeric	Field description	Comments
TNENTL	5	5	Length of entry	
TNSEQN	10	10	Sequence number	
TNCODE	1	0	Journal code	Always “M”
TNENTT	2	0	Entry type	Always “TN”
TNTIME	26	0	SAA timestamp	
TNRES	95	0	Reserved area	
TNLINE	10	0	Line Description	“*ALL” if TFREVT is “U*”. Blank if TFREVT is “L*”. Line name if TFREVT is “L”.
TNREVT	2	0	Rule Event	“L*” or “L” when rules are loaded. “U*” when rules are unloaded. “A” when filter action.
TNPDIR	1	0	IP Packet Direction	“O” is outbound. “I” is inbound.
TNRULE	5	0	Rule Indicator	Applies to the rule number in the active rules file.
TNPROT	4	0	Transport Protocol	1 is ICMP 6 is TCP 17 is UDP
TNUTSA	15	0	Untranslated src IP	
TNUTSP	5	0	Untransl. src Port	

Field name	Field length	Numeric	Field description	Comments
TNUTDA	15	0	Untranslated Dest IP	
TNUTDP	5	0	Untranslated Dest Port	
TNTRSA	15	0	Translated src IP	
TNTRSP	5	0	Translated src Port	
TNTRDA	15	0	Translated Dest IP	
TNTRDP	5	0	Translated Dest Port	
TNTEXT	76	0	Text Information	Contains description if TFREVT = “L*” or “L” or “U*”

10.7 Scenarios

This section provides an overview of the different NAT and IP Packet Filtering scenarios in this book. Each scenario is described and the configuration of the scenario is shown in 10.8, “Configuring the scenarios” on page 398.

The diagrams showing the scenarios use the CIDR method of specifying a network mask. For example, the network 10.1.1.0 subnet mask 255.255.255.0 can also be specified as network 10.1.1.0/24. The 24 identifies the number of significant bits in the IP address to use as the network. The 255.255.255.0 subnet mask identifies that the first three bytes of the mask should be used. Since each byte consists of 8 bits, this equals the 24 bits used in the CIDR method.

10.7.1 Static NAT and IP Packet Filtering

This scenario shows the use of static NAT and IP Packet Filtering between two networks. The internal network, 10.1.1.0/24, is hidden behind the AS22 system. The 192.168.1.0/24 network is the external network in this case. This scenario is typical when connecting two networks from different organizations and can be used during a transition period. The users on the AS23 system use the Telnet server running on AS21, and will use the AS22 system as an intermediate system. For security reasons, IP Packet Filter rules at the AS22 system allow access to the Telnet server only (TCP port 23). All other packets are denied. The NAT translation uses static NAT to translate the external address 192.168.1.1(AS22) to the internal address 10.1.1.2 (AS21).

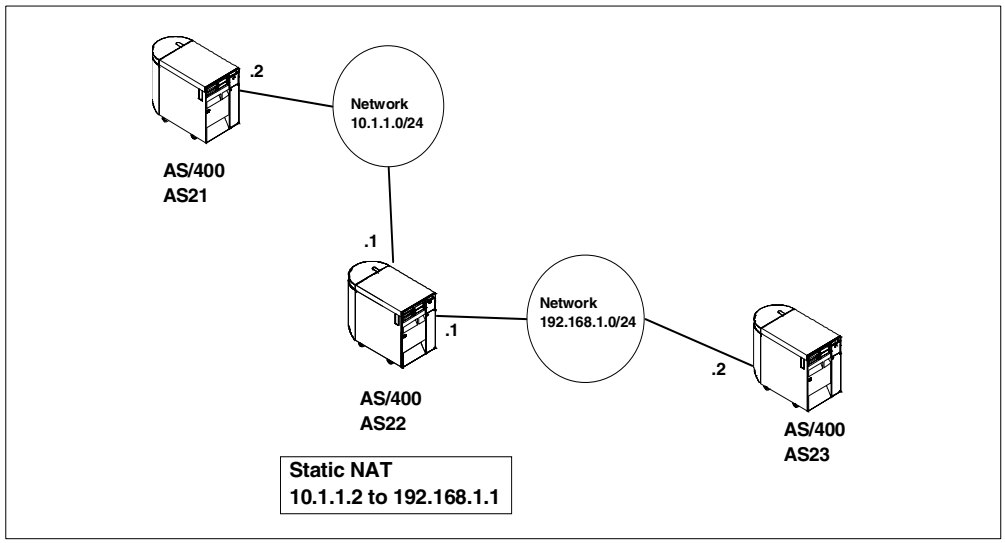


Figure 487. Static NAT between different networks

10.7.2 Masquerading NAT and IP Packet Filtering

This scenario shows the use of Masquerading NAT and IP Packet Filtering. The internal network, 10.1.1.0/24, is hidden behind the AS22 system. The 192.168.1.0/24 network is the external network. The users on the AS21 and the PCSSL systems all need access to the Telnet server on the AS22 system. This is not a problem for the users from the AS21 system, the internal network. The problem is the user at the Windows 9x/NT PCSSL, the external network. This could be users from the Internet. Telnet should not be used across the Internet, since user and password information is sent as clear text. The solution to this problem is to use the AS/400 Telnet SSL Proxy. Please refer to 5.2, “AS/400 Telnet SSL Proxy” on page 224, for more information. By using IP Packet Filtering on the AS22 system, we can limit the access to the AS22 system to the secure port that the Telnet SSL Proxy uses (TCP port 992).

The users at the AS21 system also want to access the Telnet server at the AS23 system. To do this, we configure the AS22 to do Masquerading NAT and create the appropriate IP Packet Filters.

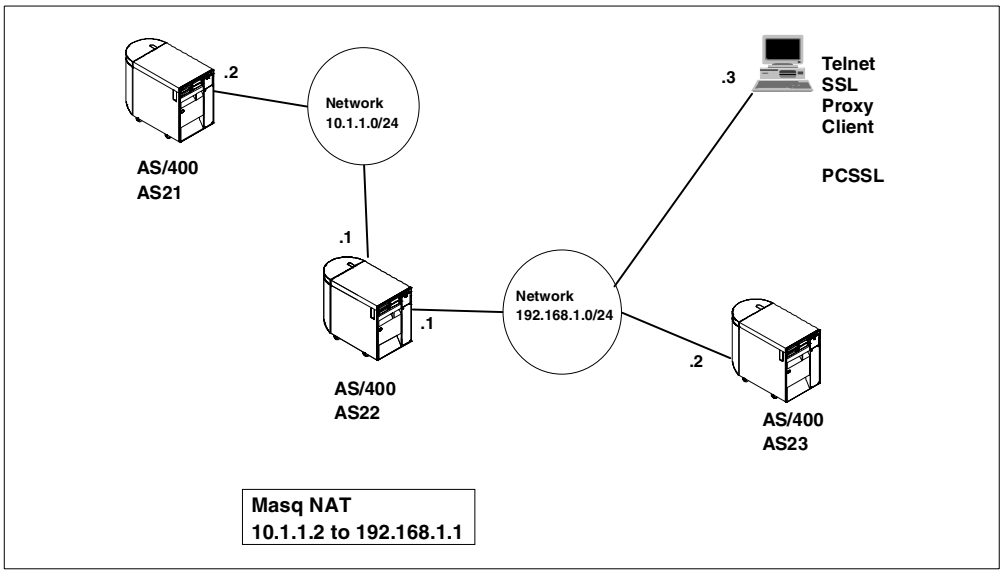


Figure 488. Masquerading NAT between different networks

10.7.3 Using IP Packet Filtering to protect a PPP connection to Domino

This scenario shows the use of IP Packet Filtering in relation to PPP connections. The scenario is based on Scenario 1 in 4.7.1, “Scenario 1: AS/400 answer and Windows PC dial” on page 98, and 4.9, “Scenario 1: AS/400 answer and Windows PC dial” on page 102. The setup and IP addresses are exactly the same. On the AS23 system, a Domino server is running. The Notes client on the PC connecting to the AS23 system using PPP is only allowed to access the Domino server (TCP port 1352) and the WWW port (TCP port 8081). Note that the Domino WWW server is using port 8081, not the well-known TCP port 80. This is not the usual case but, on the AS23 system, the IBM HTTP server is also running, and this server is using TCP port 80.

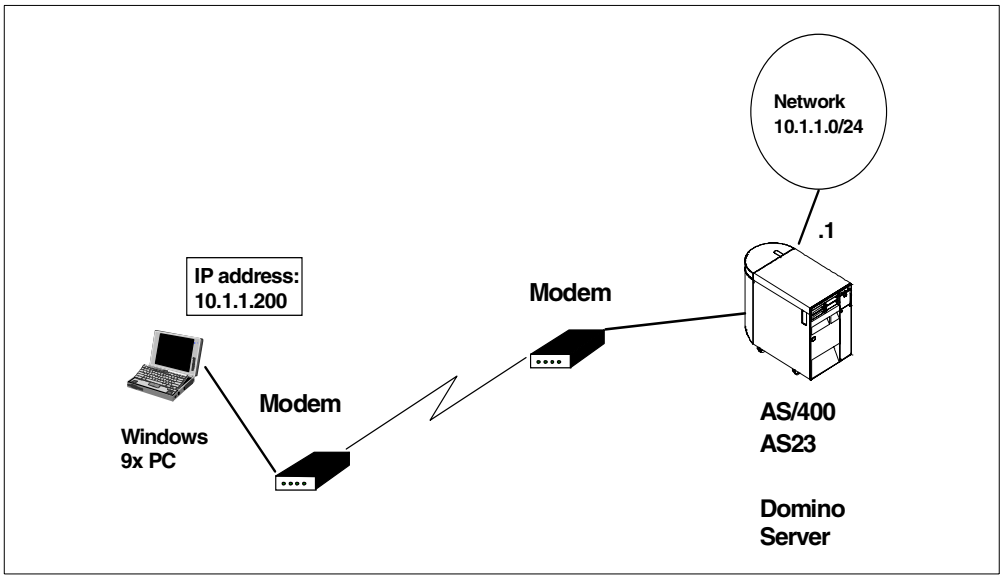


Figure 489. Using IP Packet Filtering to secure a PPP connection to a Domino Server

10.8 Configuring the scenarios

The following section discusses configuring the scenarios.

10.8.1 Using standard filter rules

The use of standard predefined IP Packet Security rules can make the process of configuring the rules much easier for the administrator. This can save time, especially if you are using predefined rules that already have been used, debugged, and tested.

The scenarios in this chapter use standard services and standard filter rules. The rules shown in the scenarios are not complete, but cover the requirements of the scenarios. You can easily add new services and filter rules to the standard files.

The standard files are *not* complete IP Packet Filter files. They only include a subset of the definitions necessary to create a valid IP Packet Security file. In other words, you cannot create a full-functioning rule set, but you can verify the contents of the standard files. The standard files are merely used as includefiles to the real rule files you are creating and should contain generic services and filter rules covering the general known services and filter rules.

In the scenarios, we use two standard files: /QIBM/standard_services.I3P and /QIBM/standard_filters.I3P. You can choose any name you like, but including *standard* in the name is a good idea.

Important

The two files used in these scenarios *are not shipped* or provided by IBM. We chose the names to make identification of the contents easy.

The /QIBM/standard_services.I3P fill includes definitions of well-known TCP, UDP, and ICMP services, such as Telnet, FTP, SMTP, POP3, DNS, PING, etc. These port numbers (TCP and UDP) and type/code (ICMP) never change and are, therefore, good candidates for inclusion in a standard file. The port numbers of the different services are based on the information found in RFC1700.

Figure 490 and Figure 491 show the standard service file used in our scenarios.

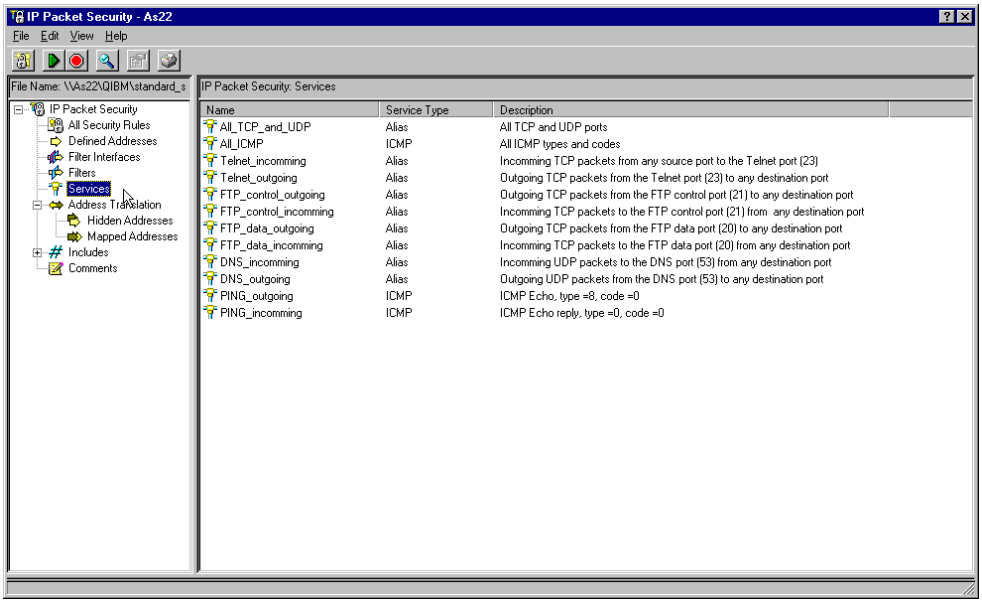


Figure 490. The /QIBM/standard_services.I3P file (Part 1)

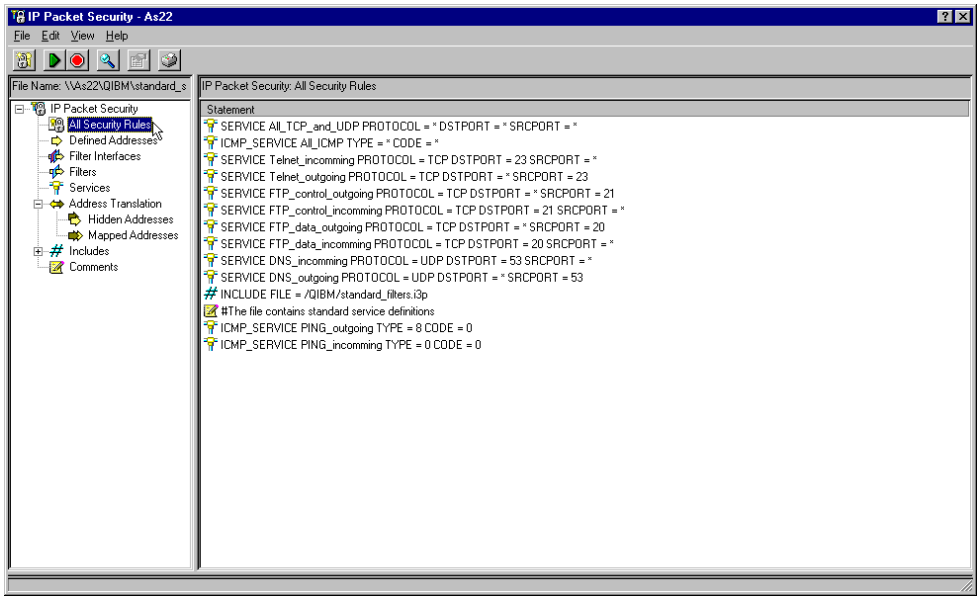


Figure 491. The /QIBM/standard_services.I3P file (Part 2)

The standard filter rules are stored in the */QIBM/standard_filters.I3P* file. You can only create very generic and general filter rules, since the rules usually include detailed information, such as specific IP addresses, direction, etc. The standard file is a good place to define the “catch-all” or “allow-all” filter rules you undoubtedly will use. When defining IP Packet Security, the function requires you to create filters for all interfaces, even if you are only focusing on the security on one of your interfaces. For example, a system has two interfaces, one to the internal network and one for the public network. You are interested in protecting the access from the public network and, therefore, define filter rules on these interfaces. You are not interested in protecting the internal network interface.

To maximize the security and monitoring of the system, you can also add generic *catch-all* filter rules to your filter files. These rules are defined so that they will log all requests that are being denied. For example, a system only allows access to the Telnet function. The specific IP Packet Filter rules file for the scenario has specifically designed filter rules that select packets based on IP addresses. These filter rules are applied to the interface. As the last filter rule, the generic “catch-all” filter rule is added as the last entry. Any packets that do not match the first filter rules (Telnet and so on) are caught by the *catch-all* entry and logs the request to the journal. This effectively provides you with a tool for monitoring any attempts to break the IP Packet Security. Please note that the IP Packet Security automatically adds a deny-all rule to an interface if a filter is defined on that particular interface. This automatically stops unwanted requests, but does not log the attempts. Adding the standard catch-all provides you with the information you need to monitor your security. Refer to 10.8.2, “Scenario 1: Static NAT and IP Packet Filtering” on page 403, for a detailed description of the configuration.

Figure 492 through Figure 499 on page 403 show the standard filter rules used in the scenarios. Notice that the *Do_not_allow...* rules are configured to log all requests and show all attempts to break the filtering rules.

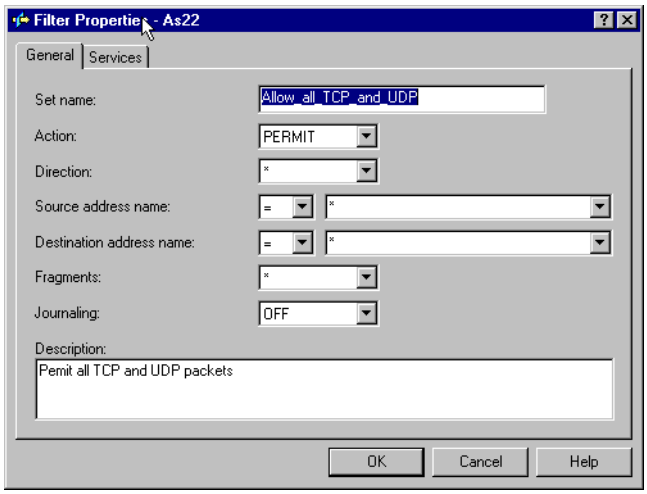


Figure 492. Allow_all_TCP_and_UDP Filter Rule (Part 1)

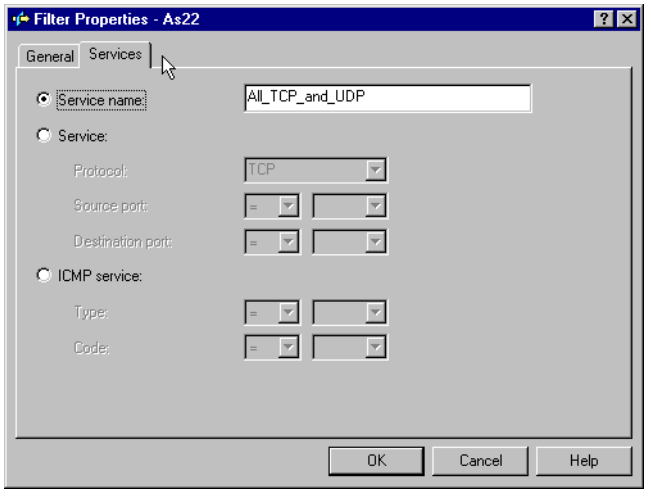


Figure 493. Allow_all_TCP_and_UDP Filter Rule (Part 2)

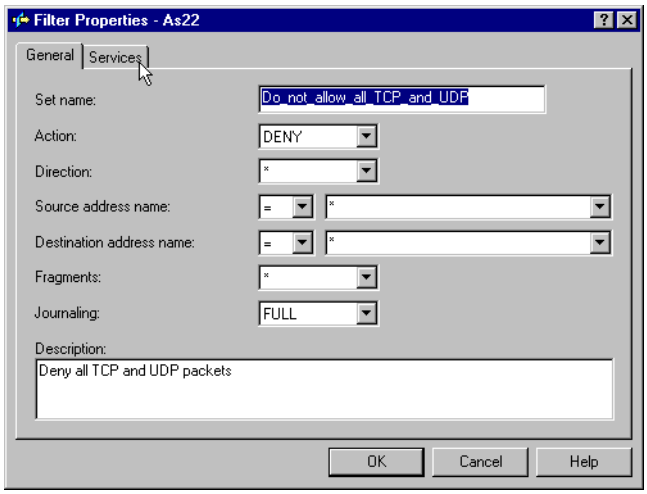


Figure 494. Do_not_allow_all_TCP_and_UDP Filter Rule (Part 1)

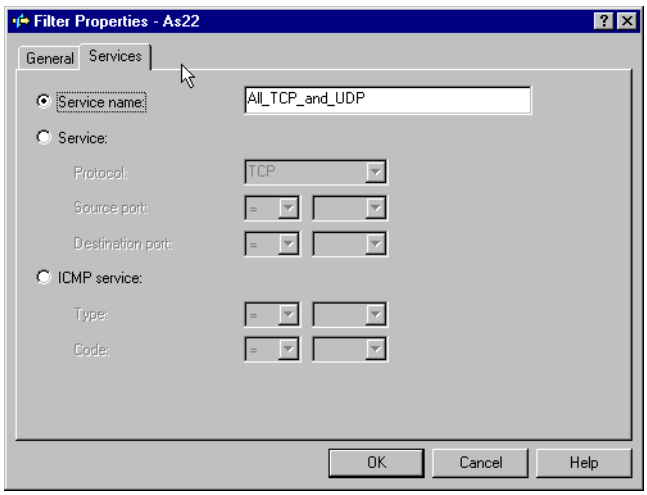


Figure 495. Do_not_allow_all_TCP_and_UDP Filter Rule (Part 2)

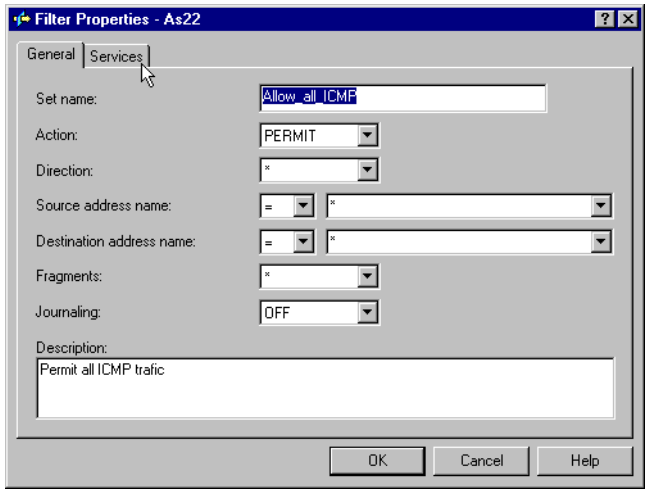


Figure 496. Allow_all_ICMP Filter Rule (Part 1)

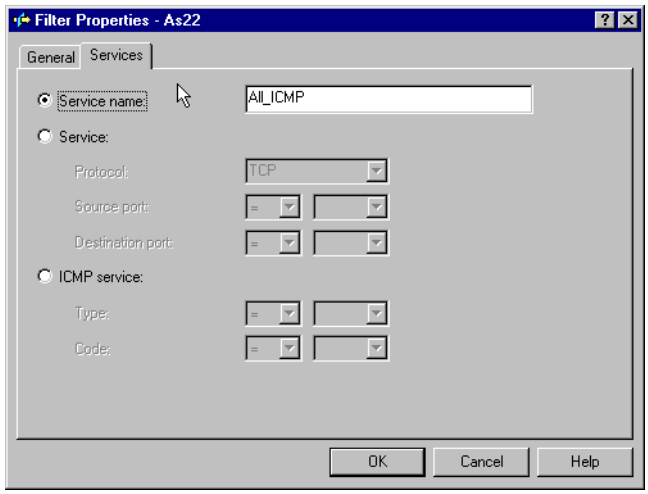


Figure 497. Allow_all_ICMP Filter Rule (Part 2)

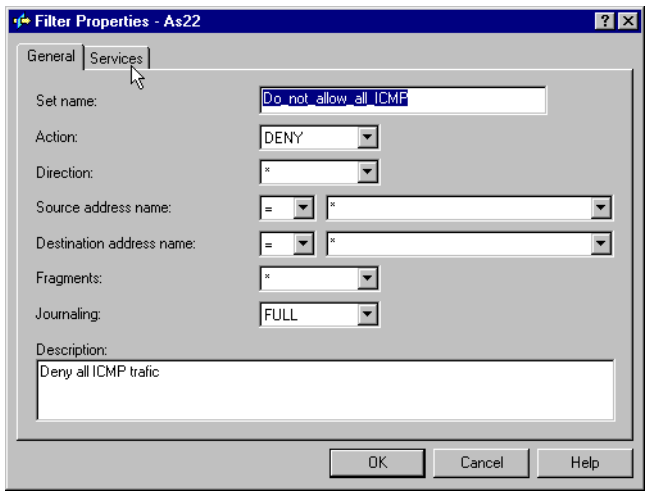


Figure 498. Do_not_allow_all_ICMP Filter Rule (Part 1)

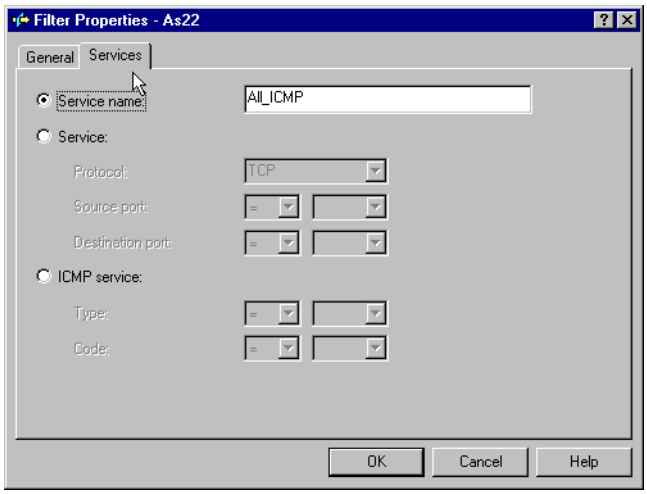


Figure 499. Do_not_allow_all_ICMP Filter Rule (Part 2)

10.8.1.1 Required PTFs

Before using the IP Packet Security functions, make sure that your system has the latest cumulative PTF package applied and that the following PTFs are applied to your system:

- MF20314
- MF20196
- MF20538
- MF20304

In addition, the latest Service Pack for Client Access for Windows 95/NT V3R3 should be applied to your PC. The scenarios in this chapter were configured using Operations Navigator, along with Service Pack SF51617.

10.8.2 Scenario 1: Static NAT and IP Packet Filtering

This scenario shows how to configure the IP Packet Security to perform IP Packet Filtering and static NAT between an internal and an external network.

10.8.2.1 Task overview

In this scenario, you complete the following tasks:

1. Configure the basic IP setup on the three AS/400 systems: AS21, AS22, and AS23.
2. Create default route entries at the AS21 and AS23 systems.
3. Enable IP datagram forwarding on the intermediate AS22 system.
4. Create a new IP Packet Security file on AS22.
5. Define addresses for use in the NAT.
6. Create mapped addresses (static NAT).
7. Include standard services and standard filter rules.
8. Create additional filter rules not found in the standard filter rule file.
9. Apply the filter rules to the interfaces on the system.
10. Verify, save, and activate the IP Packet Security rules.

10.8.2.2 Configuring

Refer to Chapter 2, “TCP/IP basic installation and configuration” on page 7, to create the basic setup of the IP interfaces on the three AS/400 systems. Use the following IP addresses on the AS/400 systems:

- AS21: 10.1.1.2/24 on the 10.1.1.0/24 network
- AS22: 10.1.1.1/24 on the 10.1.1.0/24 network and
  192.168.1.1/24 on the 192.168.1.0/24 network
- AS23: 192.168.1.2/24 on the 192.168.1.0/24 network

Follow this process:

1. Add default route entries on both the AS21 and the AS23 system. Both systems will use the AS22 system as the default route. Create the default route entry using the Add TCP/IP Route (ADDTCPRTE) command. Figure 500 and Figure 501 show how to create the default routing entries.

```
                        Add TCP/IP Route (ADDTCPRTE)

Type choices, press Enter.

Route destination  . . . . . .   *DFTROUTE
Subnet mask  . . . . . . . . . .  *NONE
Type of service  . . . . . . . .  *NORMAL       *MINDELAY, *MAXTHRPUT...
Next hop . . . . . . . . . . . .  10.1.1.1
Preferred binding interface  . .  *NONE
Maximum transmission unit  . . .  576           576-16388, *IFC
Route metric . . . . . . . . . .  1             1-16
Route redistribution . . . . . .  *NO           *NO, *YES
Duplicate route priority . . . .  5             1-10

                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 500. Adding the default routing entry to the AS21 system

```
                        Add TCP/IP Route (ADDTCPRTE)

Type choices, press Enter.

Route destination  . . . . . .   *DFTROUTE
Subnet mask  . . . . . . . . . .  *NONE
Type of service  . . . . . . . .  *NORMAL       *MINDELAY, *MAXTHRPUT...
Next hop . . . . . . . . . . . .  192.168.1.1
Preferred binding interface  . .  *NONE
Maximum transmission unit  . . .  576           576-16388, *IFC
Route metric . . . . . . . . . .  1             1-16
Route redistribution . . . . . .  *NO           *NO, *YES
Duplicate route priority . . . .  5             1-10

                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 501. Adding the default routing entry to the AS23 system

2. To enable the IP datagram forwarding on the AS22 system, use the Change TCP/IP Attributes (CHGTCPA) command:

   ```
   CHGTCPA IPDTGFWD(*YES)
   ```

3. Create a new IP Packet Security file on the AS22 system. Use the following steps to create the new empty file:

   a. Start the IP Packet Security function (Figure 502 on page 406).

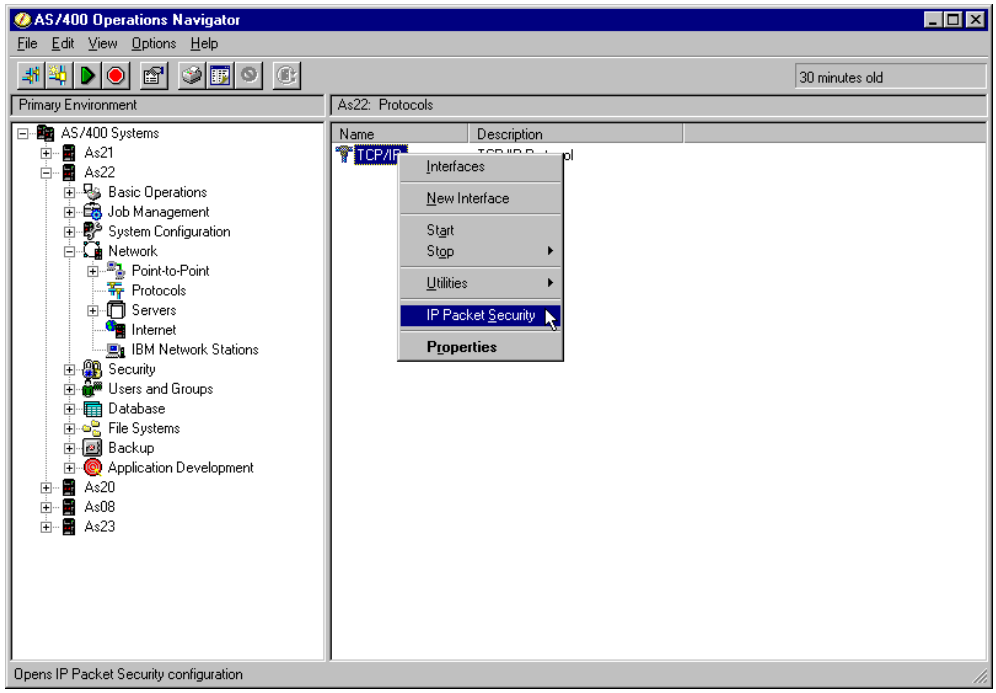


Figure 502. Starting the IP Packet Security function

b. Create a new empty file. Select **File->New File** (Figure 503).

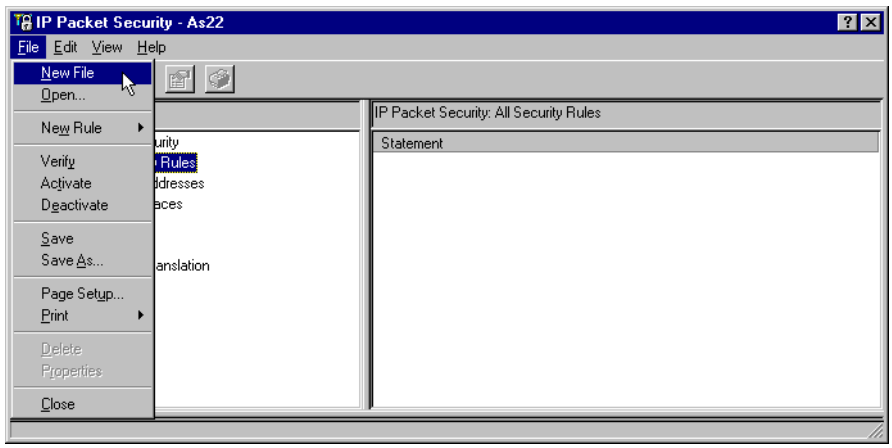


Figure 503. Creating a new IP Packet Security file

4. Create aliases for the IP addresses to include in the NAT definition. Create an alias for the private address and an alias for the public IP address (Figure 504 and Figure 505).

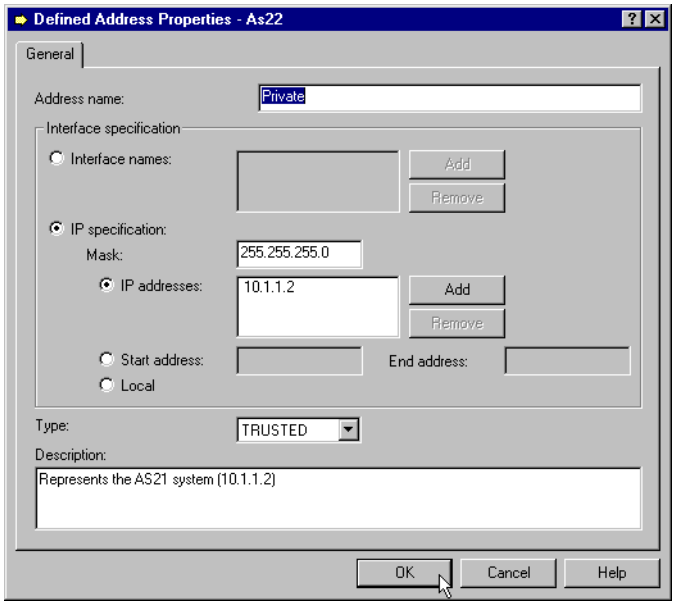


Figure 504. Creating an alias for the AS21 system (Part 1)

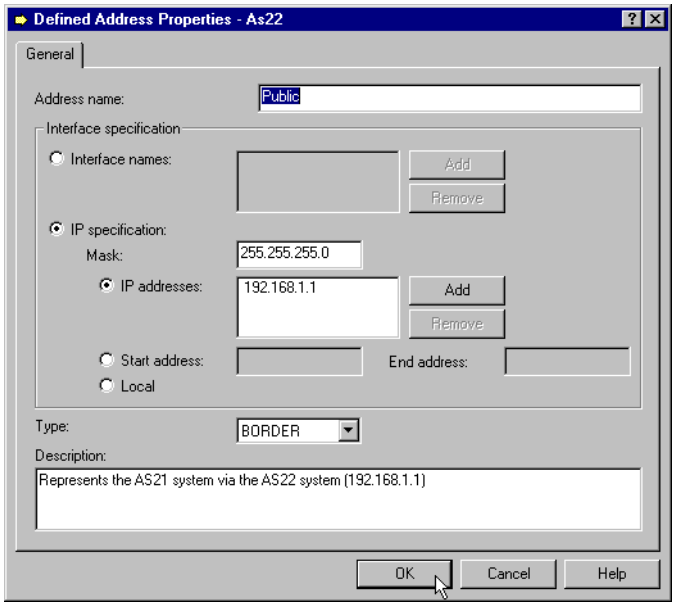


Figure 505. Creating an alias for the AS21 system (Part 2)

5. Create the static NAT entry. Select **Address Translation->Mapped Address**. The line selected is the line connected to the external network (192.168.1.0/24). Usually you select **OFF** for journaling. If you want to log any translations, you can change this to **FULL** or **STARTS**. The former logs every translation, while the latter logs only the first request in each direction.

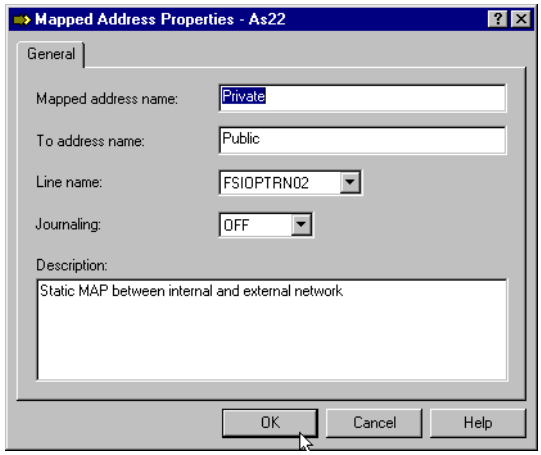


Figure 506. Creating a mapped address (static NAT) at the AS22 system

6. Since we will use some of the standard services and filter rules in this scenario, we include them from the relevant files.

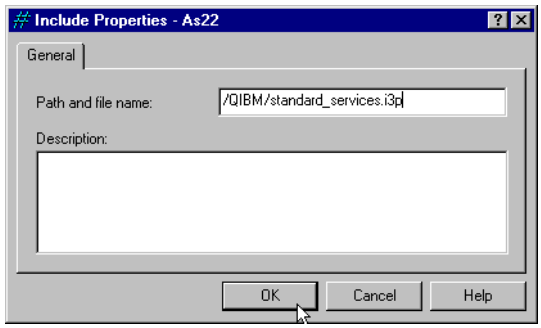


Figure 507. Including the standard services

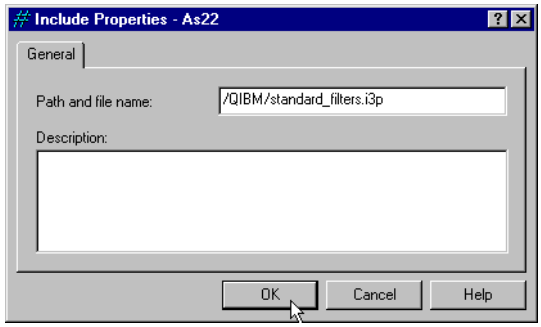


Figure 508. Including the standard filters

7. We now create additional rules that are unique to our requirements. We only allow access to and from the Telnet server at the AS21 system (10.1.1.2). Note that the filter rules deal with the internal IP address of the AS21 system. This IP address is NATed to the 192.168.1.1 address when the datagrams are returned to the external network. See 10.4, “Where and when NAT and IP Packet Filtering is done” on page 378, for more information on the sequence of the NAT and IP Packet Filtering. Note that journaling is turned OFF. In

debugging situations, it can be convenient to turn journaling on by specifying FULL.

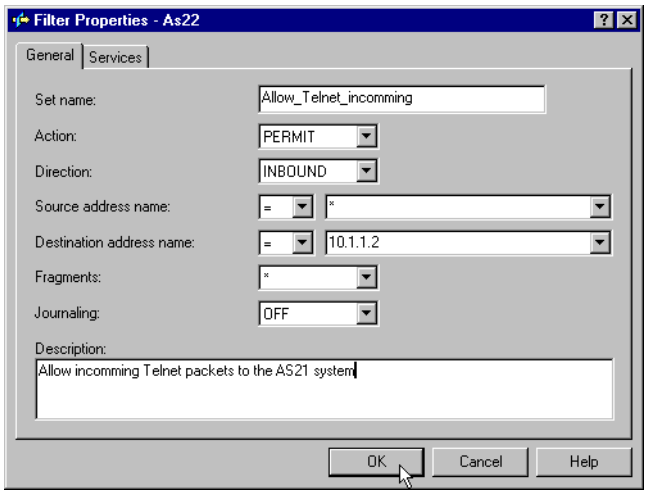


Figure 509. Allowing incomming Telnet packets (Part 1)

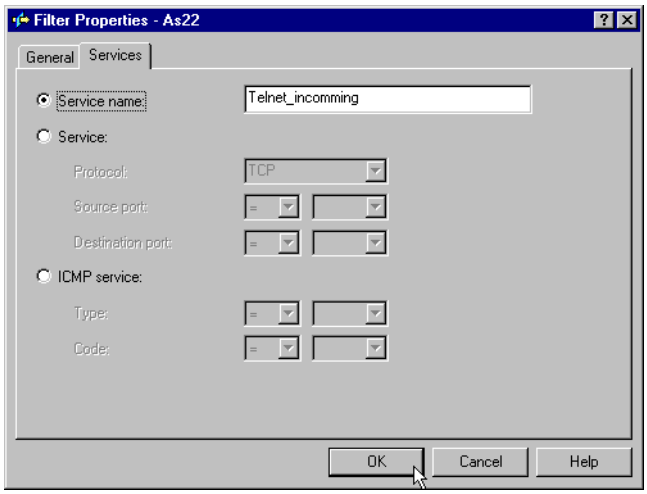


Figure 510. Allowing incomming Telnet packets (Part 2)

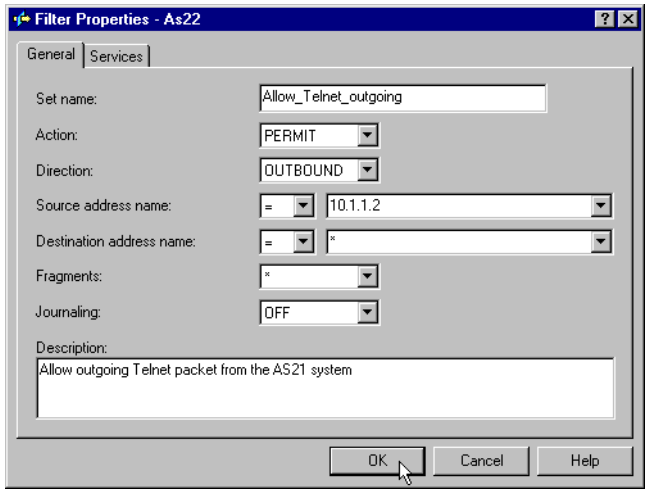


Figure 511. Allowing outgoing Telnet packets (Part 1)

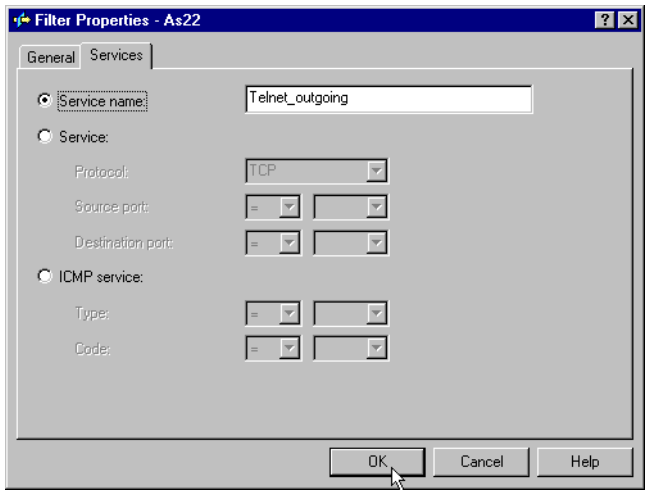


Figure 512. Allowing outgoing Telnet packets (Part 2)

8. We now need to select the interfaces and the corresponding filter rules to be applied. The AS22 system has two interfaces: the private (10.1.1.1) and the public (192.168.1.1). We do not want any restrictions on the private interface, and, therefore, apply the “Allow_all...” filter rules to this interfaces. The public interface should only allow Telnet traffic and should report any attempts to break the system. Therefore, we assign the rules to allow the Telnet traffic and, as the last entry, add the rules that catch any intruders.

   Note: The order of the filter rules is extremely important.

   If a match is found, the processing of the filter rules is skipped and the packet is forwarded or rejected as specified in the rules.

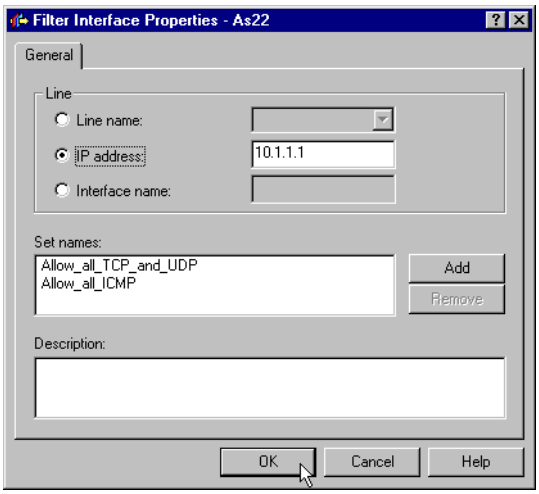


Figure 513. Filter Rules applied to the private interface

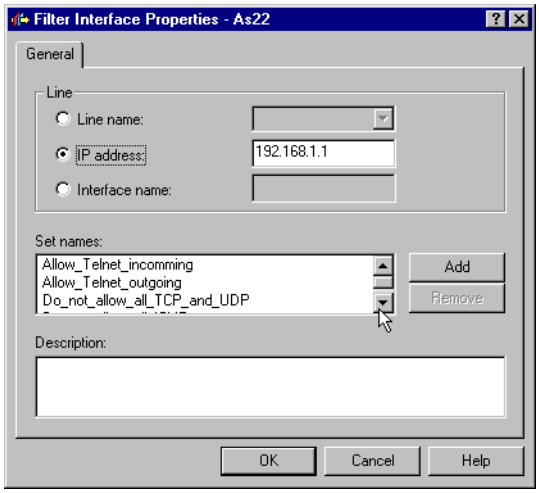


Figure 514. Filter Rules applied to the public interface (Part 1)

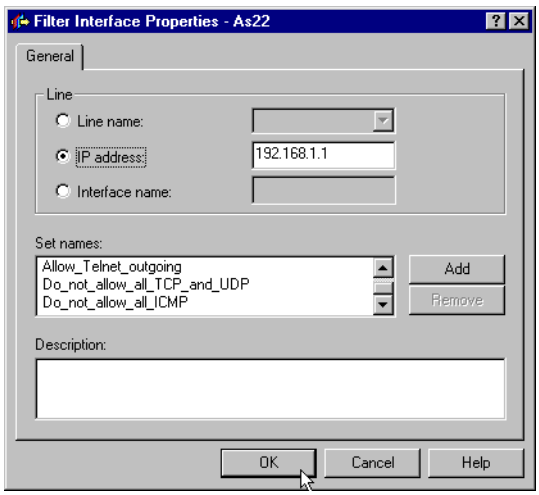


Figure 515. Filter Rules applied to the public interface (Part 2)

9. Now you should verify the IP Packet Security rules. Select **File->Verify**. The Save Rules File As windows appears. Enter the name of the rules file. After the file is saved, the rules are verified on the AS/400 system. If successful, you can activate the rule by selecting **File->Activate**.

10.8.2.3 Testing

Testing the rules is extremely important to make sure that the security settings do not allow any unwanted intruders. The test is performed with the Start TCP/IP TELNET (TELNET) command as shown in Figure 516 and Figure 517.

```
MAIN                              AS/400 Main Menu
                                                          System:   AS23
Select one of the following:

     1. User tasks
     2. Office tasks
     3. General system tasks
     4. Files, libraries, and folders
     5. Programming
     6. Communications
     7. Define or change the system
     8. Problem handling
     9. Display a menu
    10. Information Assistant options
    11. Client Access/400 tasks

    90. Sign off

Selection or command
===> telnet '192.168.1.1'

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=Information Assistant
F23=Set initial menu
```

Figure 516. Using Telnet from AS23 to reach the AS21 system (Part 1)

```
                                  Sign On
                                                 System  . . . . . :   AS21
                                                 Subsystem . . . . :   QINTER
                                                 Display . . . . . :   QPADEV0001

                 User  . . . . . . . . . . . . . .
                 Password  . . . . . . . . . . . .
                 Program/procedure . . . . . . . .
                 Menu  . . . . . . . . . . . . . .
                 Current library . . . . . . . . .









                                          (C) COPYRIGHT IBM CORP. 1980, 1998.
```

Figure 517. Using Telnet from AS23 to reach the AS21 system (Part 2)

To test that no other connections than Telnet are possible, we try to Telnet to another TCP port, the SMTP port (25). See Figure 518 and Figure 519 on page 414.

```
MAIN                              AS/400 Main Menu
                                                                System:   AS23
Select one of the following:

     1. User tasks
     2. Office tasks
     3. General system tasks
     4. Files, libraries, and folders
     5. Programming
     6. Communications
     7. Define or change the system
     8. Problem handling
     9. Display a menu
    10. Information Assistant options
    11. Client Access/400 tasks

    90. Sign off

Selection or command
===> TELNET RMTSYS('192.168.1.1') PORT(25)

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=Information Assistant
F23=Set initial menu
```

Figure 518. Testing the connection to the SMTP port (Part 1)

```
MAIN                          AS/400 Main Menu
                                                          System:   AS23
Select one of the following:

     1. User tasks
     2. Office tasks
     3. General system tasks
     4. Files, libraries, and folders
     5. Programming
     6. Communications
     7. Define or change the system
     8. Problem handling
     9. Display a menu
    10. Information Assistant options
    11. Client Access/400 tasks

    90. Sign off

Selection or command
===> TELNET RMTSYS('192.168.1.1') PORT(25)

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=Information Assistant
F23=Set initial menu
No response from remote host system within open time-out.
```

Figure 519. Testing the connection to the SMTP port (Part 2)

As you can see, there is no access to the SMTP port. To check rejection of the connection, we check the journal at the AS22 system. We configured a “catch-all” entry to catch all non-Telnet connections and log them to the journal. The entry in the journal is shown in Figure 520.

```
Display Journal Entry

Object . . . . . :                 Library  . . . . . :
Member . . . . . :                 Sequence . . . . . :   967
Code . . . . . :   M  - Network management data
Type . . . . . :   TF - IP filter rules actions

           Entry specific data
Column     *...+....1....+....2....+....3....+....4....+....5
00001     'FSIOPTRN02A I   10DENY    6192.168.1.2     164410'
00051     '.1.1.2          25                              '
00101     '                                                '

                                                                    Bottom
Press Enter to continue.

F3=Exit   F6=Display only entry specific data
F10=Display only entry details   F12=Cancel   F24=More keys
```

Figure 520. The rejected Telnet TCP request to the SMTP port

10.8.3 Scenario 2: Masquerading NAT and IP Packet Filtering

This scenario shows how to configure the IP Packet Security to perform IP Packet Filtering and Masquerading NAT between an internal and an external network. In addition to this, the scenario shows the use of the Telnet SSL proxy. Please refer to 5.2, “AS/400 Telnet SSL Proxy” on page 224, for more information on the Telnet SSL Proxy.

10.8.3.1 Task overview

In this scenario, the following tasks are accomplished:

1. Configure the basic IP setup on the three AS/400 systems, AS21, AS22, and AS23, and the Windows system PCSLL.
2. Create default route entries at the AS21, AS23 and PCSSL systems.
3. Enable IP datagram forwarding on the intermediate AS22 system.
4. Create a new IP Packet Security file on AS22.
5. Define addresses for use in the NAT.
6. Create hidden addresses (Masquerading NAT).
7. Include standard services and standard filter rules.
8. Create additional filter rules not found in the standard filter rule file to allow Telnet and Telnet SSL traffic.
9. Apply the filter rules to the interfaces on the system.
10. Verify, save, and activate the IP Packet Security rules.

10.8.3.2 Configuring

Refer to Chapter 2, “TCP/IP basic installation and configuration” on page 7, and the relevant Windows documentation to create the basic setup of the IP interfaces on the systems. Use the following IP addresses:

- AS21: 10.1.1.2/24 on the 10.1.1.0/24 network
- AS22: 10.1.1.1/24 on the 10.1.1.0/24 network and
  192.168.1.1/24 on the 192.168.1.0/24 network
- AS23: 192.168.1.2/24 on the 192.168.1.0/24 network
- PCSSL: 192.168.1.3/24 on the 192.168.1.0/24 network

Follow this process:

1. Add default route entries on both the AS21 and the AS23 system. You should also add the AS22 (192.168.1.1) as the default router for the PCSSL system. Both AS/400 systems will use the AS22 system as the default route. You create the default route entry using the Add TCP/IP Route (ADDTCPRTE) command. Figure 521 and Figure 522 on page 416 show how to create the default routing entries.

```
                        Add TCP/IP Route (ADDTCPRTE)

Type choices, press Enter.

Route destination  . . . . . .   *DFTROUTE
Subnet mask  . . . . . . . . . .  *NONE
Type of service  . . . . . . . .  *NORMAL       *MINDELAY, *MAXTHRPUT...
Next hop . . . . . . . . . . . .  10.1.1.1
Preferred binding interface  . .  *NONE
Maximum transmission unit  . . .  576           576-16388, *IFC
Route metric . . . . . . . . . .  1             1-16
Route redistribution . . . . . .  *NO           *NO, *YES
Duplicate route priority . . . .  5             1-10

                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 521. Adding the default routing entry to the AS21 system

```
                        Add TCP/IP Route (ADDTCPRTE)

Type choices, press Enter.

Route destination  . . . . . .   *DFTROUTE
Subnet mask  . . . . . . . . . .  *NONE
Type of service  . . . . . . . .  *NORMAL       *MINDELAY, *MAXTHRPUT...
Next hop . . . . . . . . . . . .  192.168.1.1
Preferred binding interface  . .  *NONE
Maximum transmission unit  . . .  576           576-16388, *IFC
Route metric . . . . . . . . . .  1             1-16
Route redistribution . . . . . .  *NO           *NO, *YES
Duplicate route priority . . . .  5             1-10

                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 522. Adding the default routing entry to the AS23 system

2. To enable the IP datagram forwarding on the AS22 system, use the Change TCP/IP Attributes (CHGTCPA) command:

   ```
   CHGTCPA IPDTGFWD(*YES)
   ```

3. Create a new IP Packet Security file on the AS22 system. Use the following steps to create the new empty file:

   a. Start the IP Packet Security function (Figure 523).

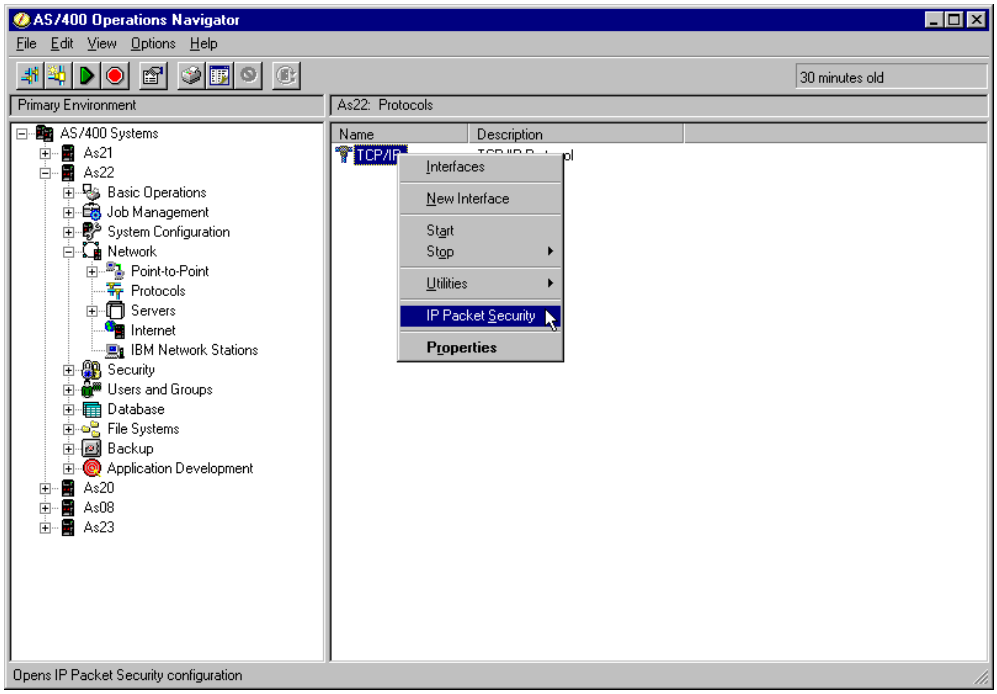


Figure 523. Starting the IP Packet Security function

b. Create a new empty file. Select **File->New File** (Figure 524).

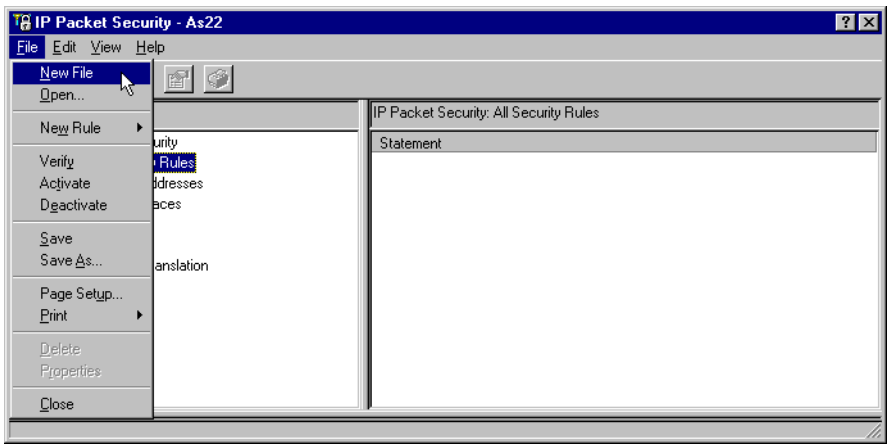


Figure 524. Creating a new IP Packet Security file

4. Create aliases for the IP address to include in the NAT definition. You create an alias for the private IP address and an alias for the public IP address (Figure 525 and Figure 526 on page 418).

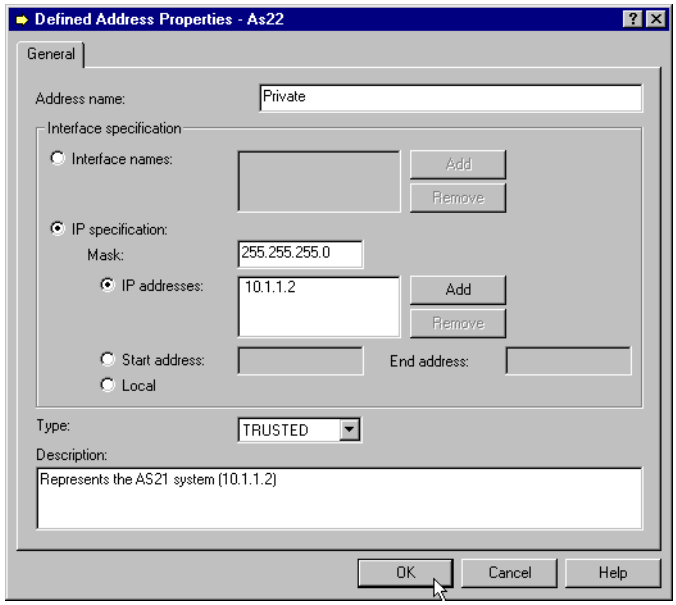


Figure 525. Creating the alias for the hidden system (AS21)

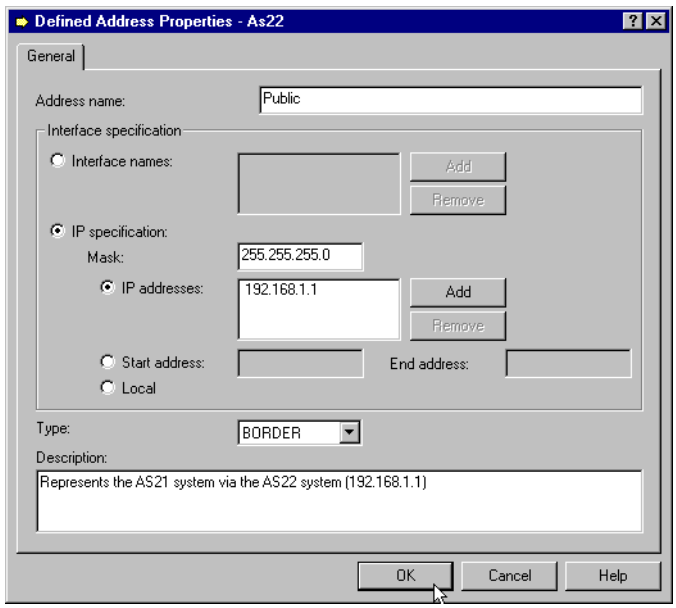


Figure 526. Creating the alias for the public interface of AS21 using AS22

5. Create the Masquerading NAT entry (select **Address Translation->Hidden Address**). We do not specify any port numbers. The NAT function allocates the port numbers. Use the default values for the Timeout and Maximum conversations fields. In the Journaling field, select **STARTS**. This enables you to see any initiations of connections using the Masquerading NAT function in the journal (Figure 527).

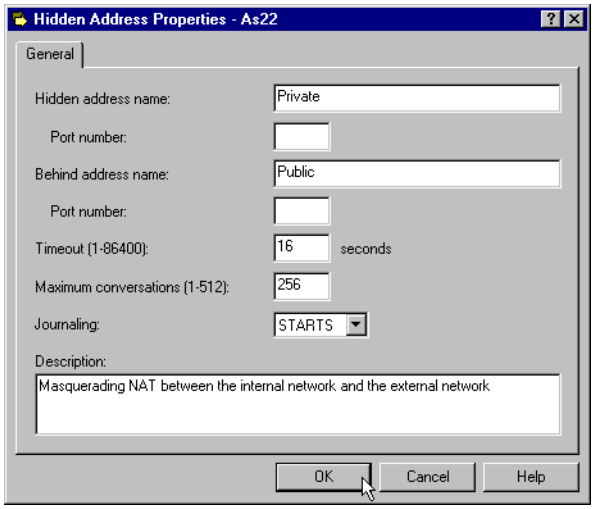


Figure 527. Creating a hidden address entry at the AS22 system

6. Since we will use some of the standard services and filter rules in this scenario, we will include them from the relevant files. This is shown in Figure 528 and Figure 529. In addition to the standard services, we define a new service for the Telnet SSL, using TCP port 992. See Figure 530 and Figure 531 on page 420.

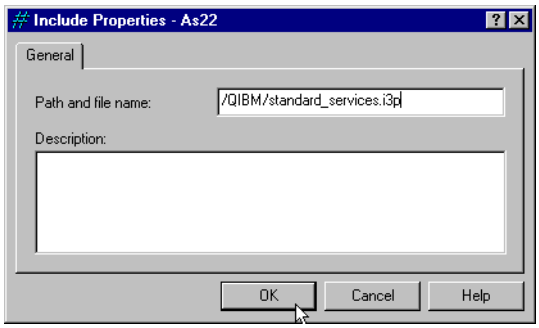


Figure 528. Including the standard services

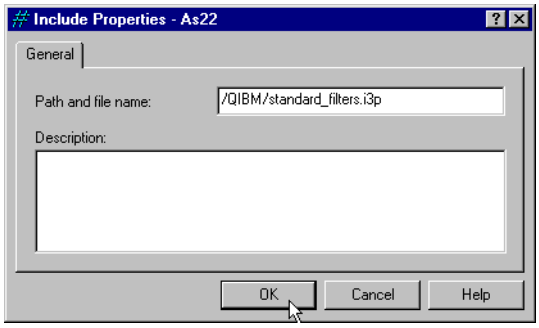


Figure 529. Including the standard filters

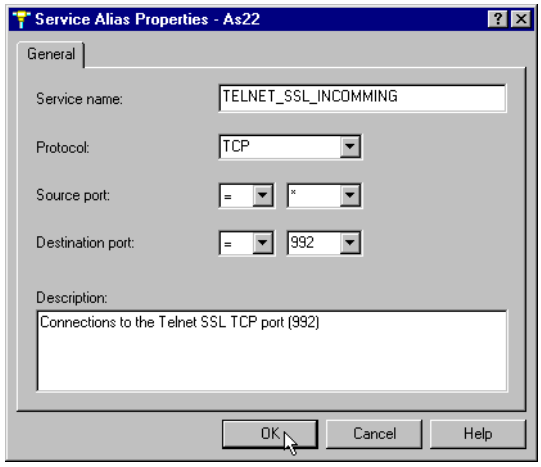


Figure 530. Creating a new service for the Telnet SSL (Part 1)

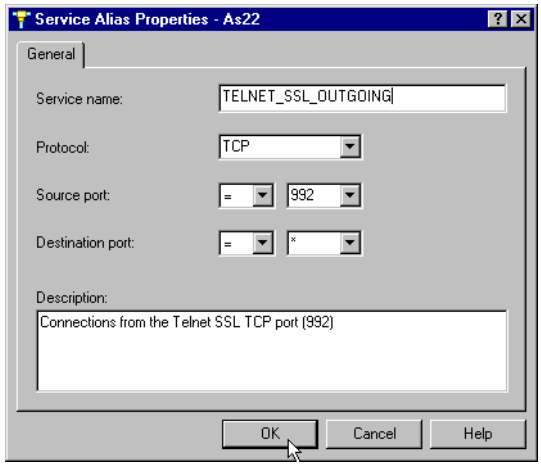


Figure 531. Creating a new service for the Telnet SSL (Part 2)

7. We now create additional rules that are unique to our requirements. We only allow access from the AS21 system (10.1.1.2) to the Telnet server at the AS23 system (192.168.1.2). In addition to this, we only allow access to the Telnet SSL port (TCP port 992) on the public interface on AS22 (192.168.1.1). Note that the journaling is turned OFF. In debugging situations, it can be convenient to turn on the journaling by specifying FULL. The filters are shown in Figure 532 through Figure 539 on page 423. Note that we are using the TELNET_OUTGOING service in the Allow_Telnet_incomming rule and the TELNET_INCOMMING service in the Allow_Telnet_outgoing rule. This is because the Telnet server being used is the Telnet server located on AS23 (192.168.1.2).

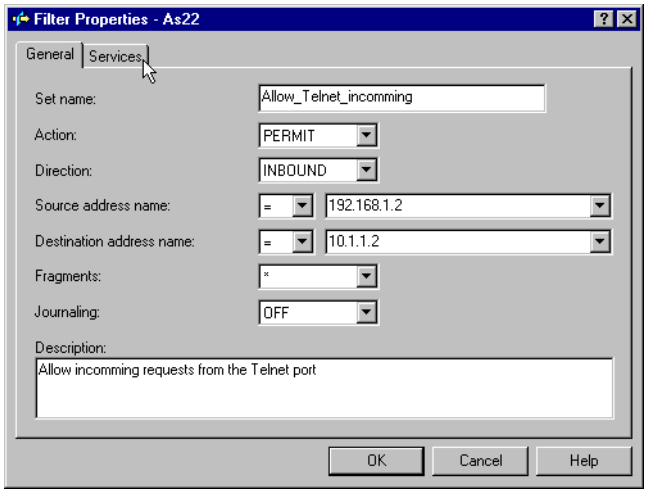


Figure 532. Creating the Allow_Telnet_incomming filter (Part 1)

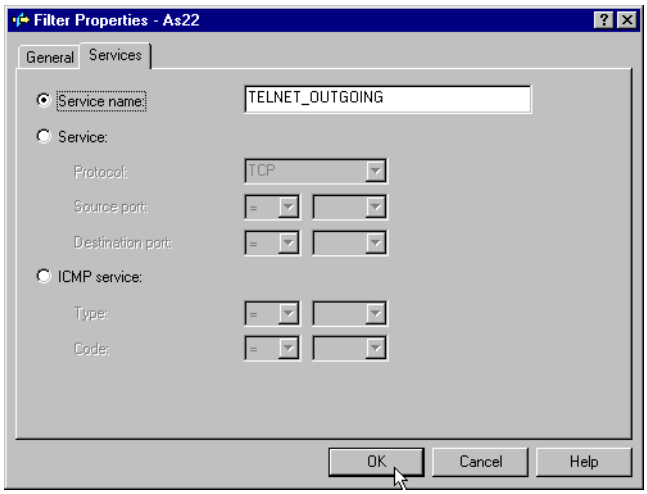


Figure 533. Creating the Allow_Telnet_incomming filter (Part 2)

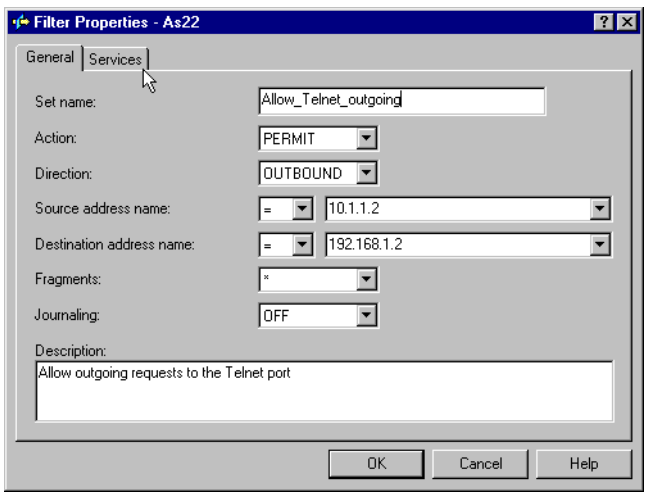


Figure 534. Creating the Allow_Telnet_outgoing filter (Part 1)

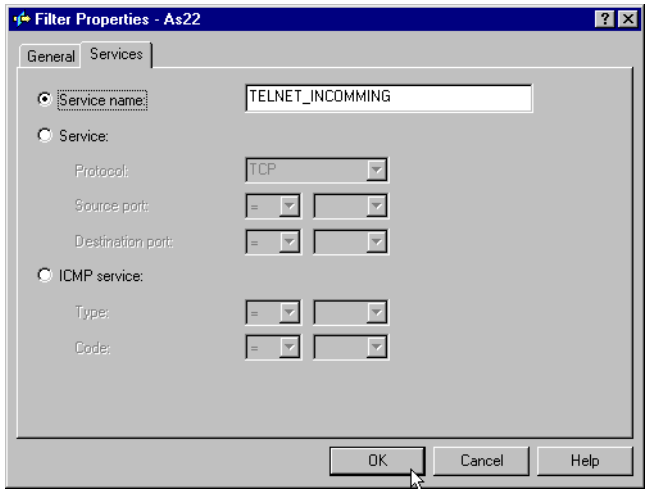


Figure 535. Creating the Allow_Telnet_outgoing filter (Part 2)

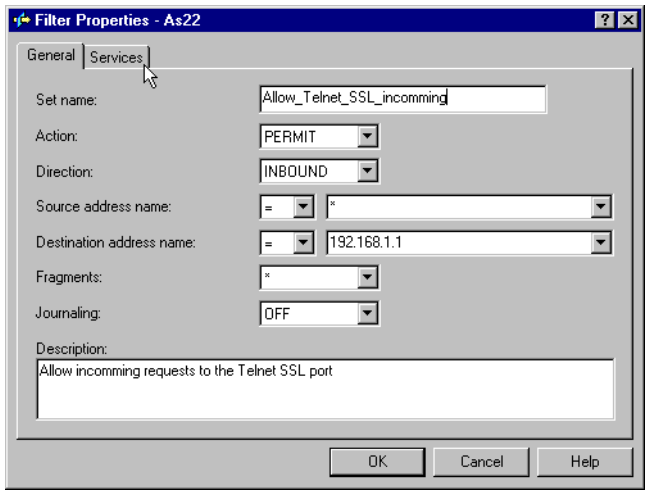


Figure 536. Creating the Allow_Telnet_SSL_incomming filter (Part 1)

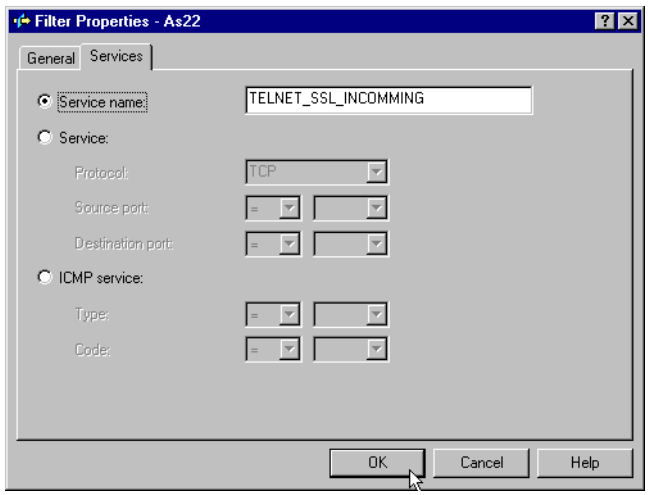


Figure 537. Creating the Allow_Telnet_SSL_incomming filter (Part 2)

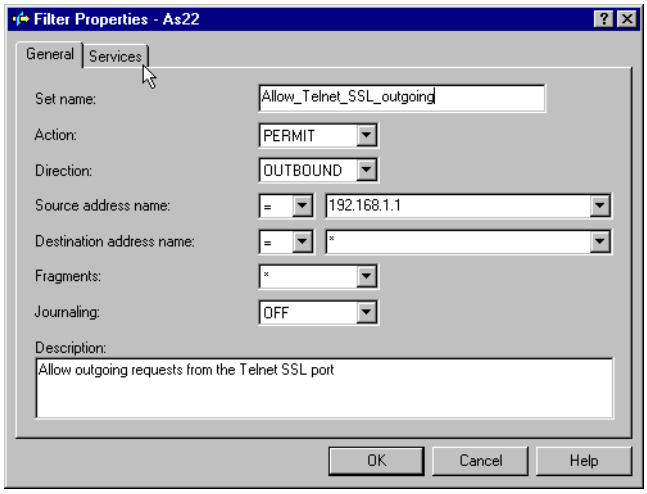


Figure 538. Creating the Allow_Telnet_SSL_outgoing filter (Part 1)

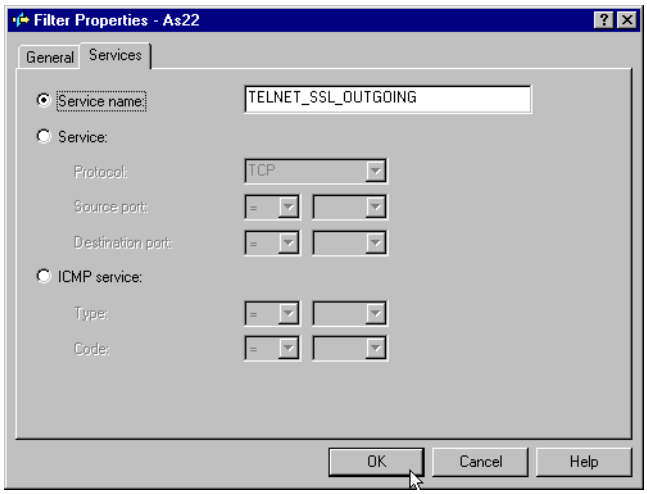


Figure 539. Creating the Allow_Telnet_SSL_outgoing filter (Part 2)

8. We now need to select the interfaces and the corresponding filter rules to be applied (Figure 540 on page 424 through Figure 543 on page 425). The AS22 system has two interfaces, the private (10.1.1.1) and the public (192.168.1.1). We do not want any restrictions on the private interface and, therefore, applies the “Allow_all...” filter rules to this interfaces. The public interface should only allow Telnet and Telnet SSL traffic and should report any attempts to break the system. Therefore, we assign the rules to allow the Telnet and Telnet SSL traffic and as the last entry add the rules that catch any intruders.

   Note: The order of the filter rules is extremely important.

   If a match is found, the processing of the filter rules is skipped, and the packet is forwarded or rejected as specified in the rules.

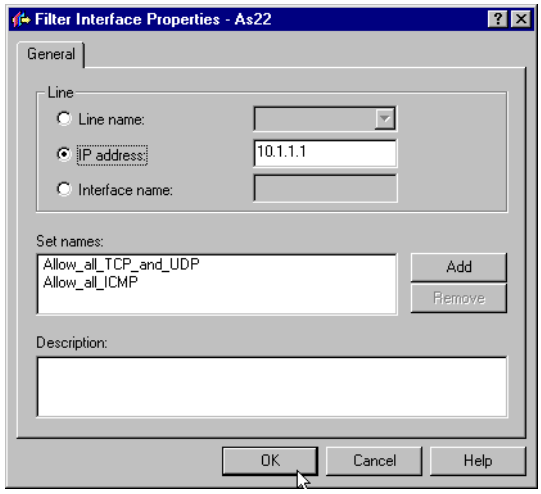


Figure 540. Assigning filters to the 10.1.1.1 interface on AS22

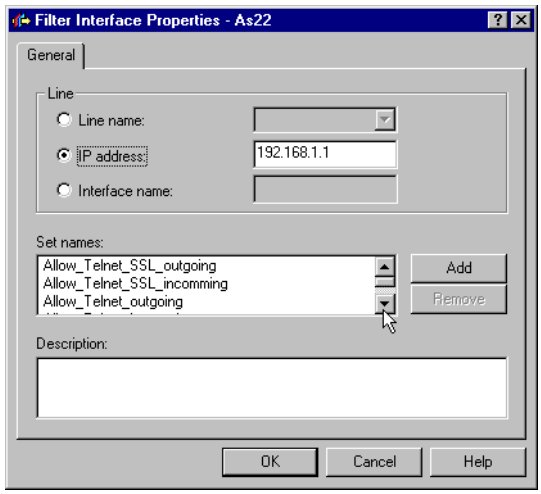


Figure 541. Assigning filters to the 192.168.1.1 interface on AS22 (Part 1)

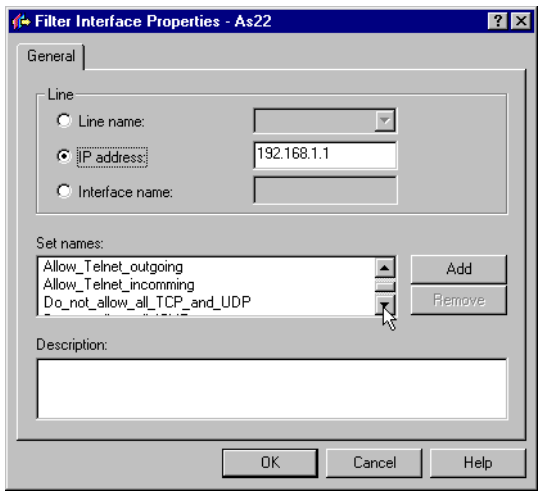


Figure 542. Assigning filters to the 192.168.1.1 interface on AS22 (Part 2)

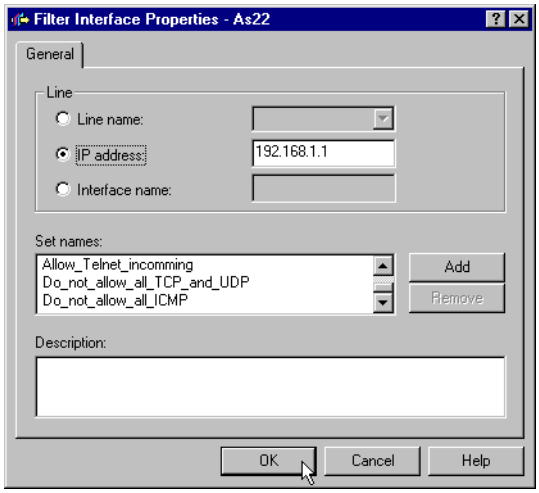


Figure 543. Assigning filters to the 192.168.1.1 interface on AS22 (Part 3)

9. Verify the IP Packet Security rules. Select **File->Verify**. The Save Rules File As window appears. Enter the name of the rules file. After the file is saved, the rules are verified on the AS/400 system. If successful, you can activate the rule by selecting **File->Activate**. In this case, the verification of the rules returns a warning related to the Allow_Telnet_SSL_incomming rule. The warning can be seen in the bottom part of the Operations Navigator IP Packet Security window. The message returned can also be found in the QTCPMSG message file on the AS/400 system. To view the complete message, use the following command:

   ```
   DSPMSGD RANGE(TCP5AFD) MSGF(QTCPMSG)
   ```

   The message details are:

   ```
   A FILTER statement may not have the intended result. Either an INBOUND
   FILTER statement has a destination address that is the second address of a
   MAP or HIDE statement, or an OUTBOUND FILTER statement has a source address
   that is the second address of a MAP or HIDE statement. It is possible that
   the FILTER will not have the intended result due to the order of rules
   processing. For INBOUND IP traffic, filtering occurs after network address
   translation and for OUTBOUND IP traffic, filtering occurs before network
   address translation.
   ```

10. This message can be ignored, since the rule Allow_Telnet_incomming ensures that the Telnet traffic can be relayed. To understand this, consider the following series of events:

    a. When the AS21 system (10.1.1.2) requests a Telnet session to AS23 (192.168.1.2), the AS22 system performing the Masquerading NAT effectively changes the source address from 10.1.1.2 to 192.168.1.1.

    b. When the datagrams return from the AS23 system (192.168.1.2) to the AS21 system (now known as 192.168.1.1), the AS22 system NATs the destination address (192.168.1.1) back to the original address of the AS21 system (10.1.1.2). This address translation occurs before the IP Packet Filters are checked.

    c. The AS22 system now checks the returning datagram from step 2 using the IP Packet Filter rules, and finds a match in the Allow_Telnet_incomming rule. The datagram is delivered to the AS21 system (10.1.1.2).

d. Any Telnet SSL connections from the outside are also successfully processed. Since the source address of the datagrams all belong to the external network, for example, 192.168.1.2, NAT is not performed on any of these packets. The Masquerading NAT is configured to translate addresses from the internal network to the external network, not the opposite.

The following extract is from the NAT journal. See 10.6.3, “Monitoring NAT and IP Packet Filtering” on page 391, for more information of the file layout, etc. The extract shows the translation from the internal AS21 system (10.1.1.2) to the public interface on the AS22 system (192.168.1.1).

```
FSIOPTRN02A O   29   610.1.1.2      1062192.168.1.2     23192.168.1.1   55535192.168.1.2
23
FSIOPTRN02A I   29   6192.168.1.2      23192.168.1.1   55535192.168.1.2      2310.1.1.2
1062
FSIOPTRN02A O   29   610.1.1.2      1062192.168.1.2     23192.168.1.1   55535192.168.1.2
23
FSIOPTRN02A O   29   610.1.1.2      1062192.168.1.2     23192.168.1.1   55535192.168.1.2
23
FSIOPTRN02A I   29   6192.168.1.2      23192.168.1.1   55535192.168.1.2      2310.1.1.2
1062
FSIOPTRN02A I   29   6192.168.1.2      23192.168.1.1   55535192.168.1.2      2310.1.1.2
1062
FSIOPTRN02A O   29   610.1.1.2      1062192.168.1.2     23192.168.1.1   55535192.168.1.2
23
FSIOPTRN02A O   29   610.1.1.2      1062192.168.1.2     23192.168.1.1   55535192.168.1.2
23
FSIOPTRN02A I   29   6192.168.1.2      23192.168.1.1   55535192.168.1.2      2310.1.1.2
1062
FSIOPTRN02A I   29   6192.168.1.2      23192.168.1.1   55535192.168.1.2      2310.1.1.2
1062
FSIOPTRN02A O   29   610.1.1.2      1062192.168.1.2     23192.168.1.1   55535192.168.1.2
23
FSIOPTRN02A I   29   6192.168.1.2      23192.168.1.1   55535192.168.1.2      2310.1.1.2
1062
FSIOPTRN02A O   29   610.1.1.2      1062192.168.1.2     23192.168.1.1   55535192.168.1.2
23
FSIOPTRN02A I   29   6192.168.1.2      23192.168.1.1   55535192.168.1.2      2310.1.1.2
1062
FSIOPTRN02A O   29   610.1.1.2      1062192.168.1.2     23192.168.1.1   55535192.168.1.2
23
FSIOPTRN02A I   29   6192.168.1.2      23192.168.1.1   55535192.168.1.2      2310.1.1.2
1062
FSIOPTRN02A I   29   6192.168.1.2      23192.168.1.1   55535192.168.1.2      2310.1.1.2
1062
FSIOPTRN02A O   29   610.1.1.2      1062192.168.1.2     23192.168.1.1   55535192.168.1.2
23
FSIOPTRN02A O   29   610.1.1.2      1062192.168.1.2     23192.168.1.1   55535192.168.1.2
23
FSIOPTRN02A I   29   6192.168.1.2      23192.168.1.1   55535192.168.1.2      2310.1.1.2
1062
```

10.8.3.3 Testing

Testing the rules is extremely important to make sure that the security settings do not allow any unwanted intruders. The test is performed with the Start TCP/IP TELNET command.

First, the Masquerading NAT is tested by starting a connection from AS21 to AS23. This is shown in Figure 544 and Figure 545.

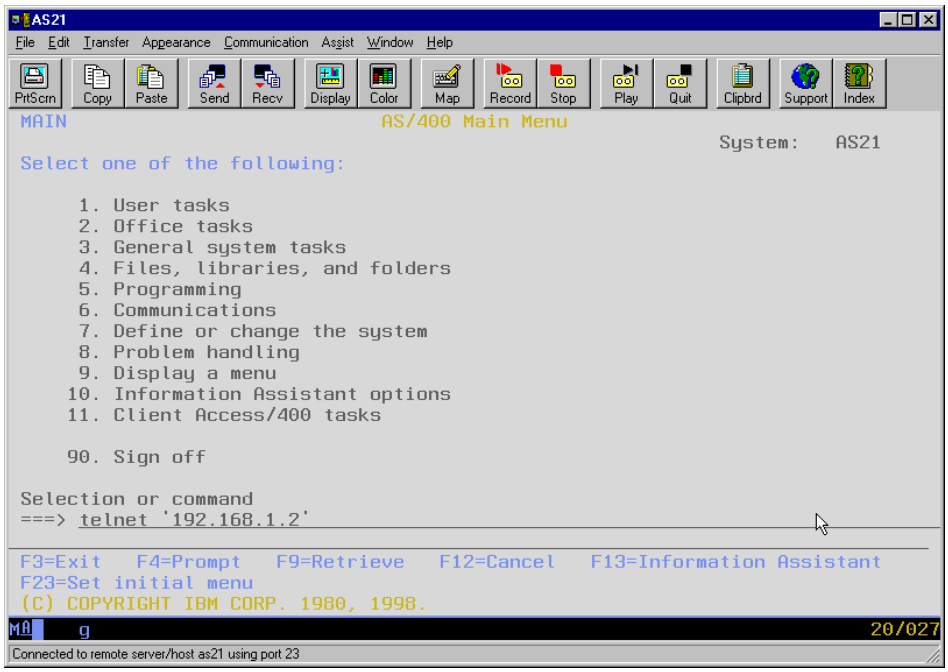


Figure 544. Testing the Telnet Connection to External AS23 (192.168.1.2) (Part 1)

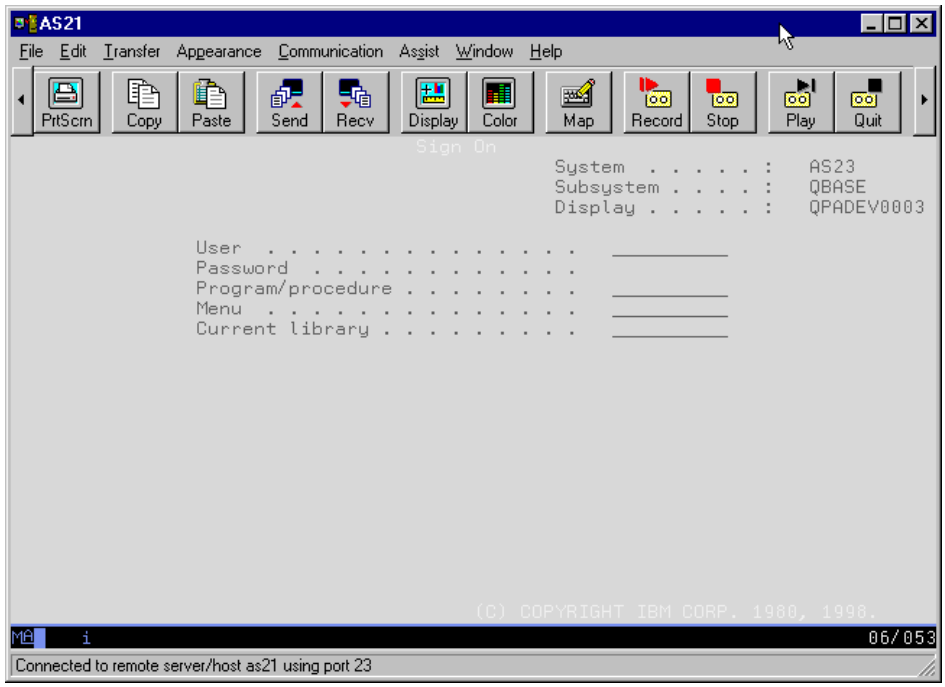


Figure 545. Testing the Telnet Connection to External AS23 (192.168.1.2) (Part 2)

Then, the Telnet from the external system (PCSSL) is tested. We need to test both the native Telnet (TCP port 23) and the Telnet SSL (TCP port 992).

Figure 546 on page 428 shows that the connection to the SSL-enabled Telnet is successful.

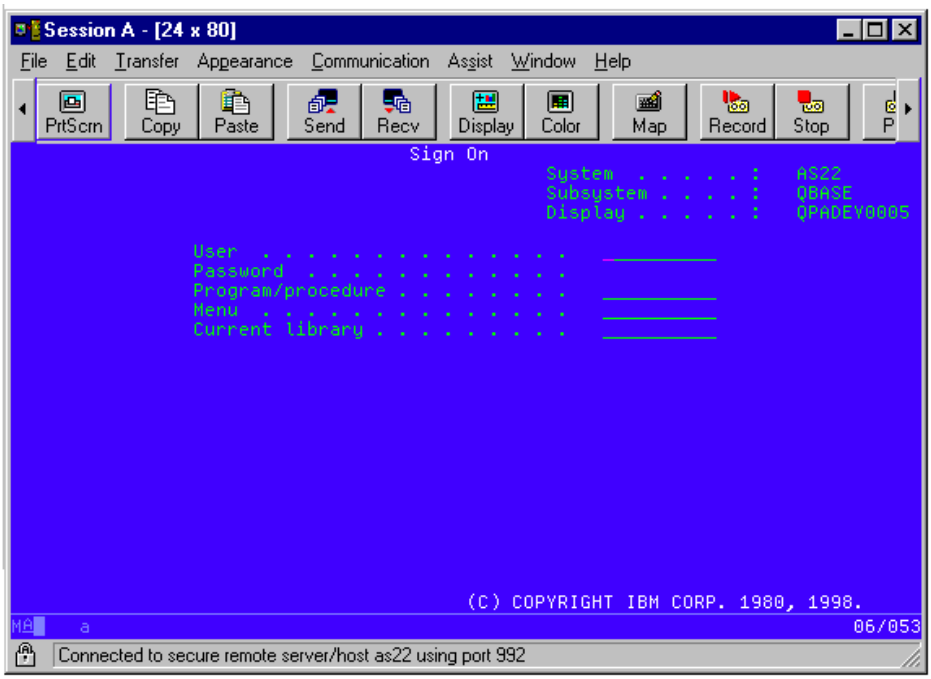


Figure 546. Connected to the AS22 system using the Telnet SLL client

The connection to the native Telnet is unsuccessful when the screen remains blank and you receive the message on the bottom of the screen: `Hostname not specified or not found`. In other words, the filter rules we created work.

The following example is an extract from the IP Packet Filtering journal. This clearly shows the packets from the PCSSL system (192.168.1.3) being denied by the AS22 system. Refer to 10.6.3, “Monitoring NAT and IP Packet Filtering” on page 391, for more information about the file layout.

```
1.  FSIOPTRN02 A I    7 DENY     6 192.168.1.3     1077 192.168.1.1       23
2.  FSIOPTRN02 A I    7 DENY     6 192.168.1.3     1079 192.168.1.1       23
3.  FSIOPTRN02 A I    7 DENY     6 192.168.1.3     1075 192.168.1.1       23
4.  FSIOPTRN02 A I    7 DENY    17 192.168.1.3      138 192.168.1.255    138
5.  FSIOPTRN02 A I    7 DENY     6 192.168.1.3     1080 192.168.1.1       23
6.  FSIOPTRN02 A I    7 DENY     6 192.168.1.3     1078 192.168.1.1       23
7.  FSIOPTRN02 A I    7 DENY     6 192.168.1.3     1103 192.168.1.1       23
8.  FSIOPTRN02 A I    7 DENY     6 192.168.1.3     1099 192.168.1.1       23
9.  FSIOPTRN02 A I    7 DENY     6 192.168.1.2     1659 192.168.1.1       25
```

Lines 1 and 2 show packets from the TN5250 session on PCSSL to the AS22 being denied.

Line 4 shows a packet from the PCSSL to AS22 being denied. The packet is a UDP packet to port 138 (NETBIOS Datagram Service). Apparently, the PCSSL system is broadcasting some UDP datagrams to all systems in the subnet (192.168.1.255)

Lines 5 through 8 are the same as lines 1 through 3.

Line 9 shows a denied packet from the AS23 system (192.168.1.2). We started a Telnet session from the AS23 (192.168.1.2) system to the AS22 (192.168.1.1) system, SMTP port (TCP port 25). The packet is correctly discarded.

10.8.4 Scenario 3: IP filters protecting a PPP connection to Domino

This scenario shows how to configure the IP Packet Security in relation to the PPP function of OS/400. The scenario is based on the PPP Scenario 1 (4.9, “Scenario 1: AS/400 answer and Windows PC dial” on page 102). This section only shows how to configure the IP Packet Security function. Follow the instructions in the PPP scenario to create the basic configuration.

This scenario covers the following tasks:

1. Configure the basic PPP scenario.
2. Create a new IP Packet Security file on AS23.
3. Include standard services and standard filter rules.
4. Create additional filter rules not found in the standard filter rule file.
5. Apply the filter rules to the interfaces on the system.
6. Verify, save, and activate the IP Packet Security rules.

10.8.4.1 Configuring

Refer to 4.9, “Scenario 1: AS/400 answer and Windows PC dial” on page 102, for further instructions on configuring the PPP connection.

1. Create a new IP Packet Security file on the AS22 system. Use the following steps to create the new empty file:

   a. Start the IP Packet Security function (Figure 547).

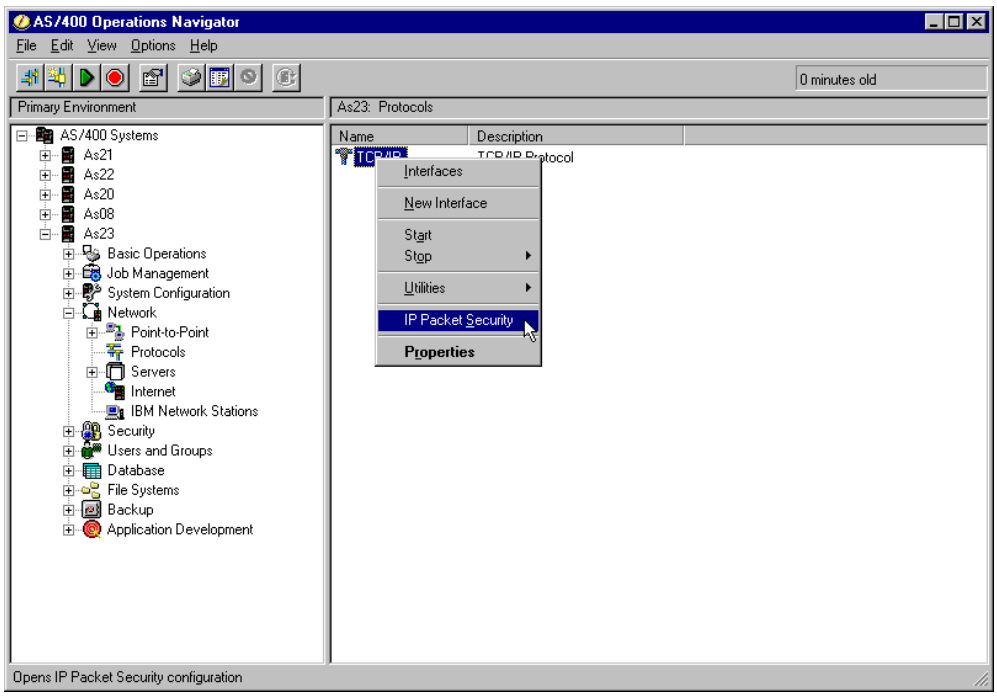


Figure 547. Starting IP Packet Security on the AS23 system

   b. Create a new empty file. Select **File->New** (Figure 548 on page 430).

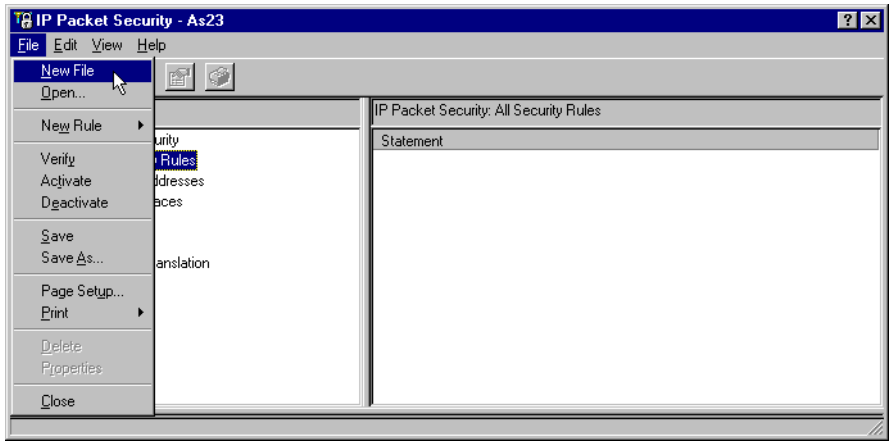


Figure 548. Creating a new IP Packet Security File on the AS23 system

2. In this scenario, we add two new services to the standard include file /QIBM/standard_services.I3P. These are the services required by the Notes client to connect to the Domino server. The service use TCP port 1352 (Figure 549 and Figure 550). In addition to the new services on the standard file, we add new services to the rules file used in this scenario. These services relate to the Domino HTTP server, running on TCP port 8081. Since this is not the usual TCP port for a HTTP server, we choose not to include the service in the standard file, but place it in the file related to this scenario (Figure 551 and Figure 552). Finally, we include the standard files to this scenario file (Figure 553 and Figure 554 on page 432).

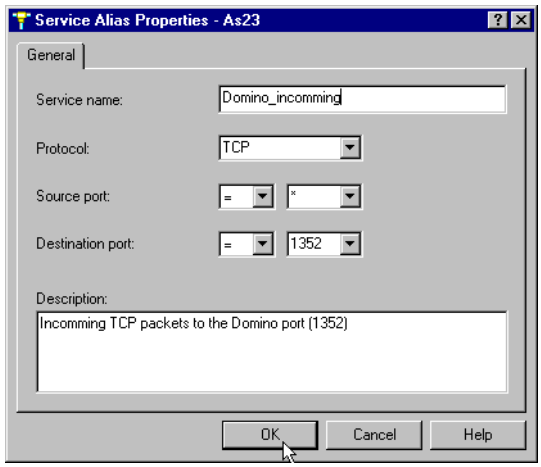


Figure 549. Adding new services to the standard file (Part 1)

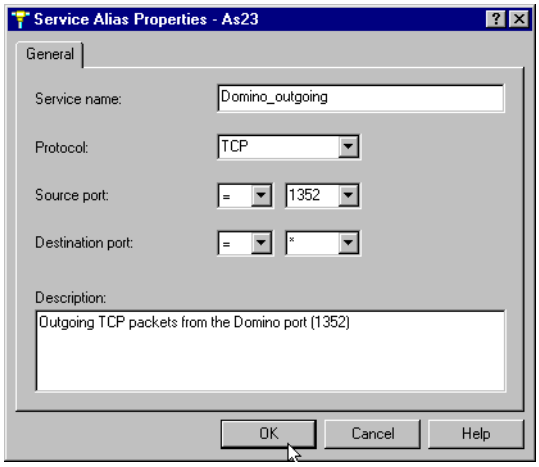


Figure 550. Adding new services to the standard file (Part 2)

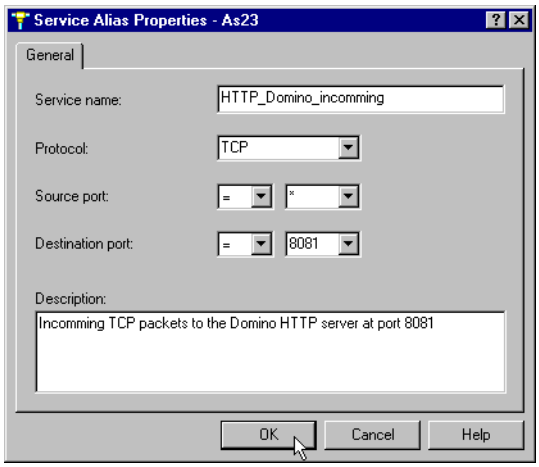


Figure 551. Adding the Domino HTTP Service (Part 1)

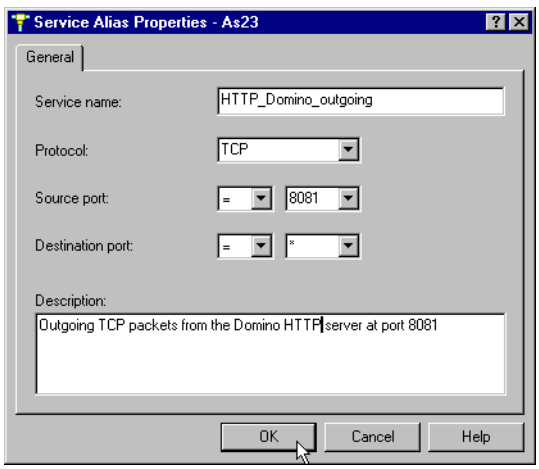


Figure 552. Adding the Domino HTTP Service (Part 2)

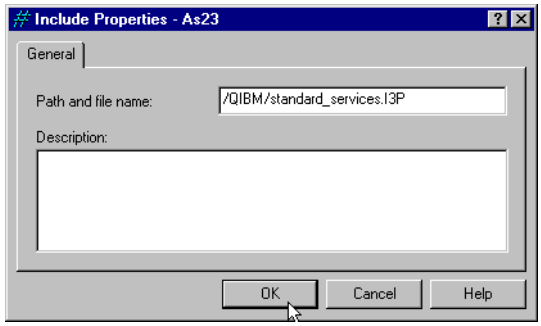


Figure 553. Including the standard services

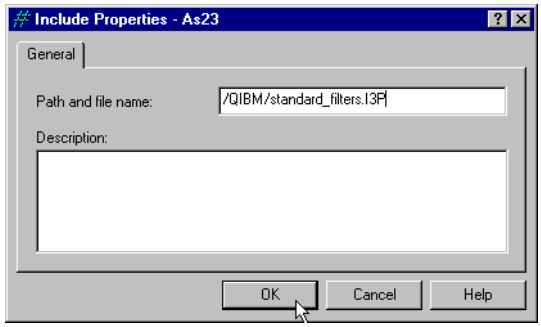


Figure 554. Including the standard filters

3. We now need to add specific filters for our scenario. Only traffic to the Domino server and the Domino HTTP server is allowed. The filters are shown in Figure 555 through Figure 562 on page 435.

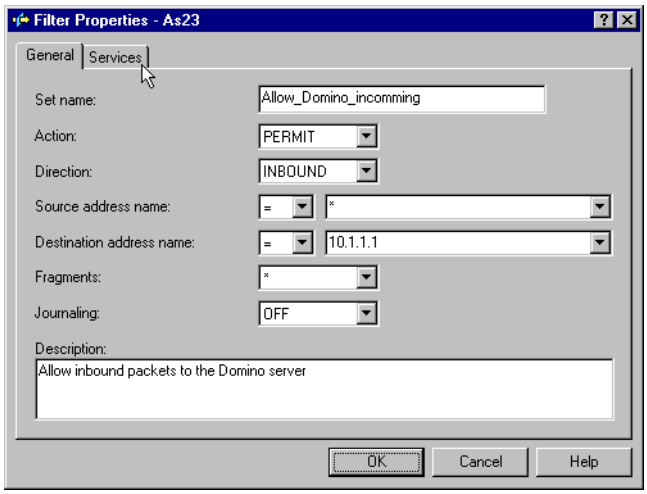


Figure 555. Configuring the Allow_Domino_incomming rule (Part 1)

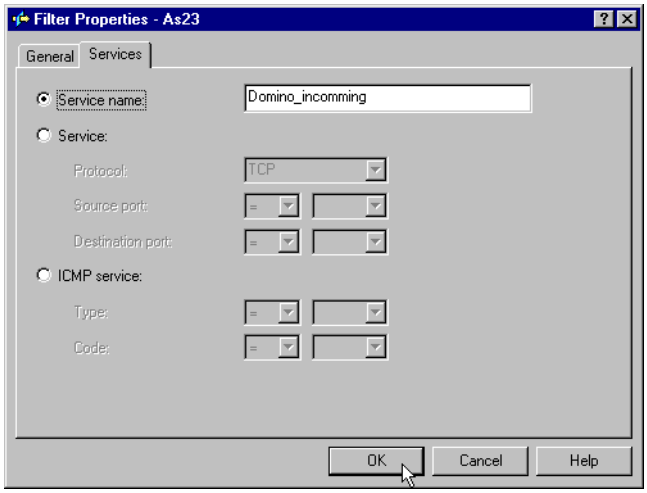


Figure 556. Configuring the Allow_Domino_incomming rule (Part 2)

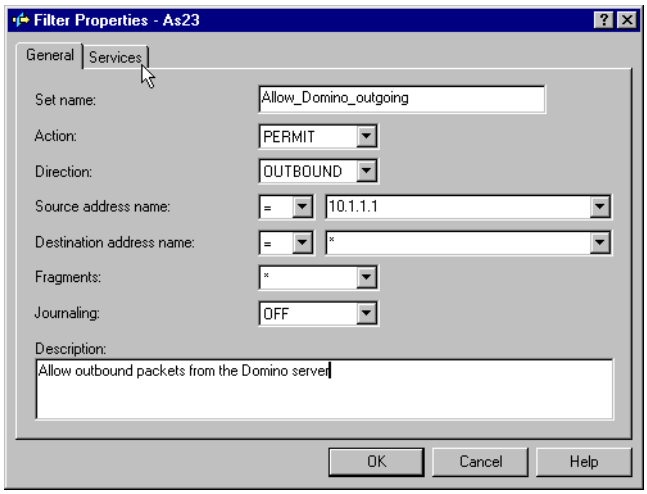


Figure 557. Configuring the Allow_Domino_outgoing rule (Part 1)

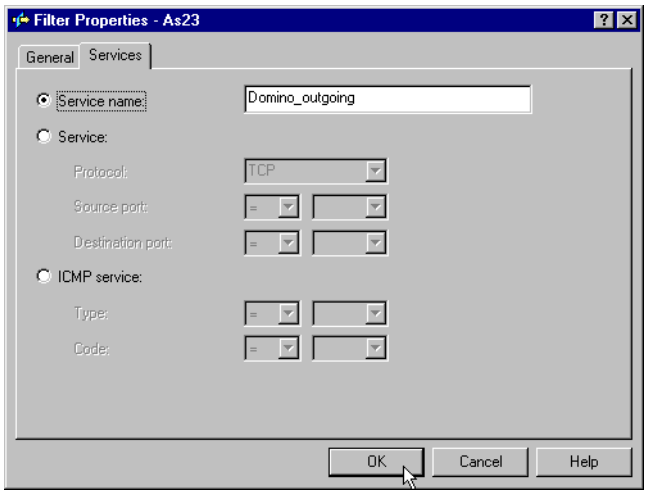


Figure 558. Configuring the Allow_Domino_outgoing rule (Part 2)

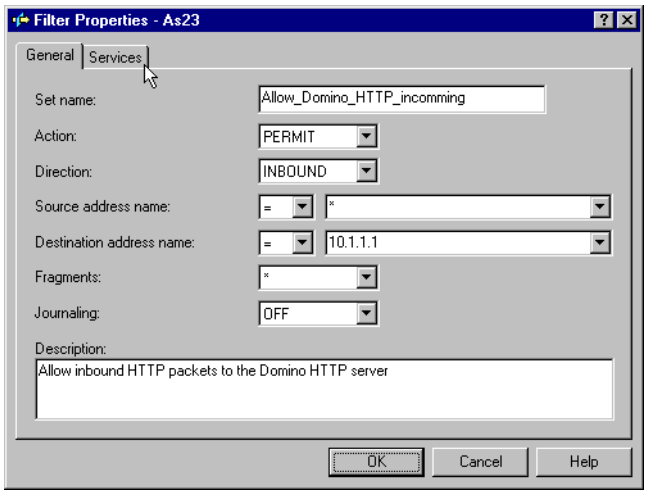


Figure 559. Configuring the Allow_Domino_HTTP_incomming rule (Part 1)

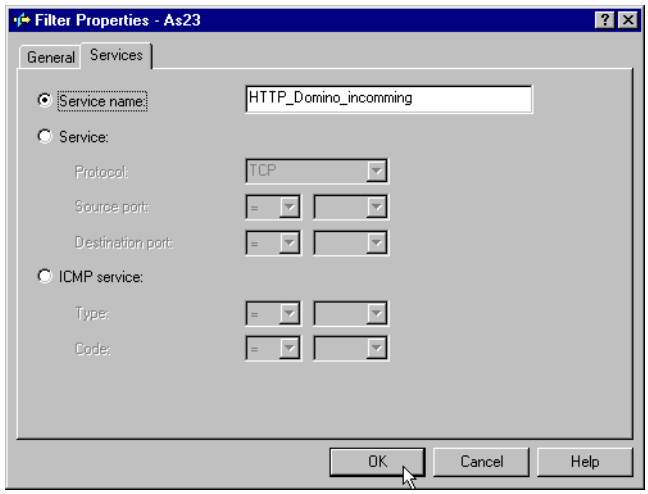


Figure 560. Configuring the Allow_Domino_HTTP_incomming rule (Part 2)

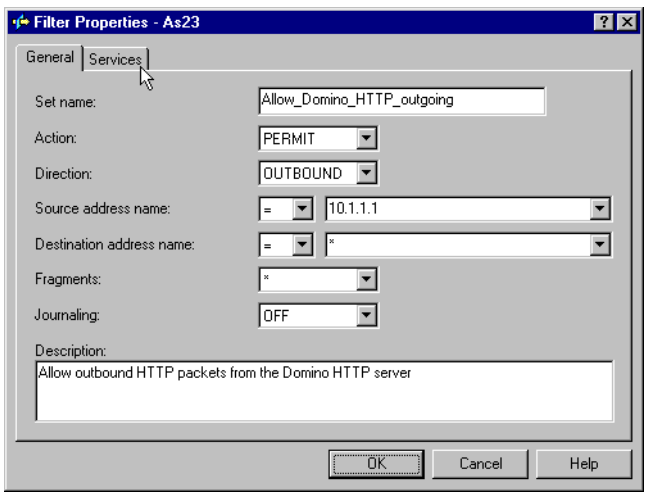


Figure 561. Configuring the Allow_Domino_HTTP_outgoing rule (Part 1)

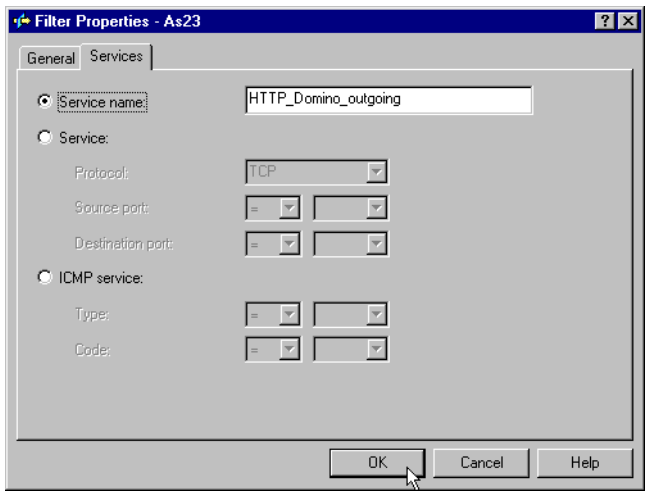


Figure 562. Configuring the Allow_Domino_HTTP_outgoing rule (Part 2)

4. We now need to select the interfaces and the corresponding filter rules to be applied. The AS23 system has two interfaces, the LAN interface (10.1.1.1) and the PPP interface activated when the PPP connection is established. We do not want any restrictions on the LAN interface and, therefore, apply the "Allow_all..." filter rules to this interfaces (Figure 563). The PPP interface should only allow Domino and HTTP requests and should report any attempts to break the system. Therefore, we assign the rules to allow the Domino and Domino HTTP traffic and as the last entry add the rules that catch any intruders (Figure 564 on page 436 through Figure 566 on page 436). We specify the name of the PPP connection profile as the interface name, which is *Scen1* in this case.

   Note: The order of the filter rules is extremely important.

   If a match is found, the processing of the filter rules is skipped and the packet is forwarded or rejected as specified in the rules.

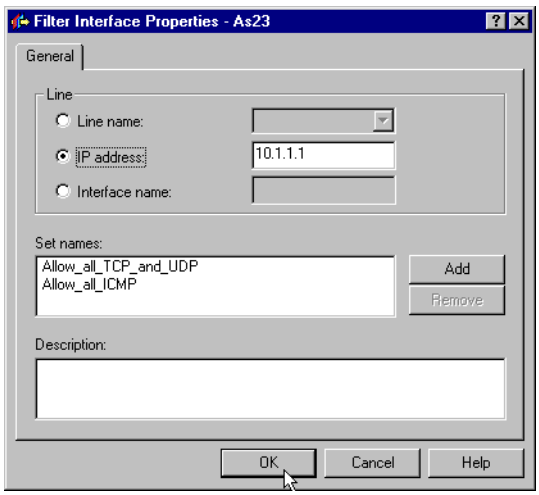


Figure 563. Assigning filters to the LAN interface on AS23

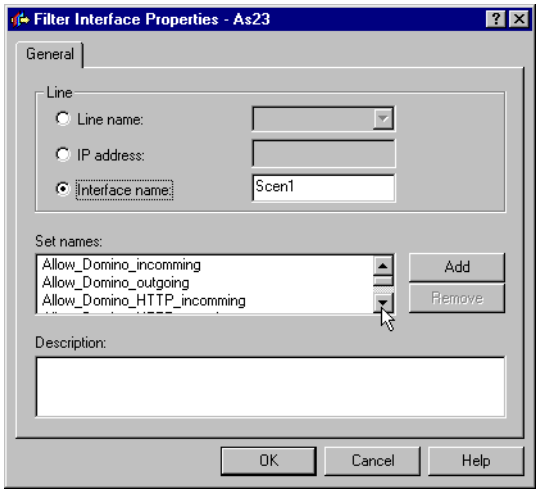


Figure 564. Assigning filters to the PPP interface on AS23 (Part 1)

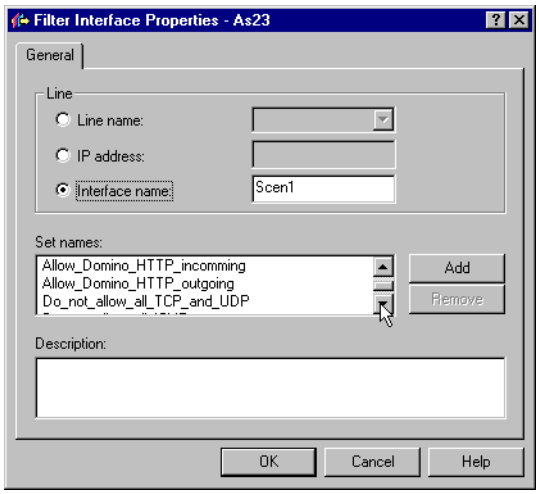


Figure 565. Assigning filters to the PPP interface on AS23 (Part 2)

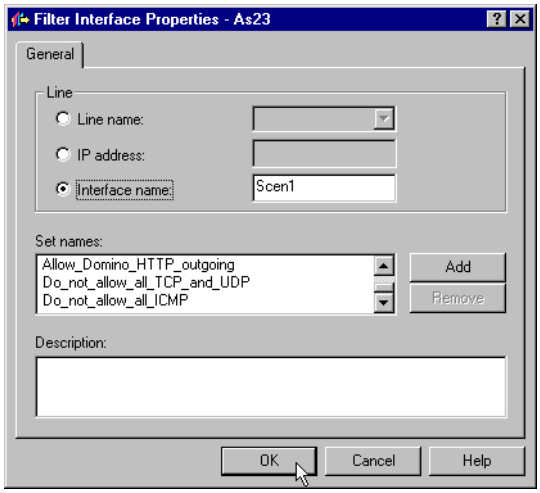


Figure 566. Assigning filters to the PPP interface on AS23 (Part 3)

5. Now you should verify the IP Packet Security rules. Select **File->Verify**. The Save Rules File As windows appears. Enter the name of the rules file. After the file is saved, the rules are verified on the AS/400 system. If successful, you can activate the rule by selecting **File->Activate**.

10.8.4.2 Testing

The test of the scenario is performed by starting the PPP connection from the PC system to the AS/400 system. After the connection is established, the Notes client on the PC is started, and the Domino server running on AS23 is accessed (Figure 567). This confirms the correct configuration of the filter rules permitting traffic to the TCP port 1352.

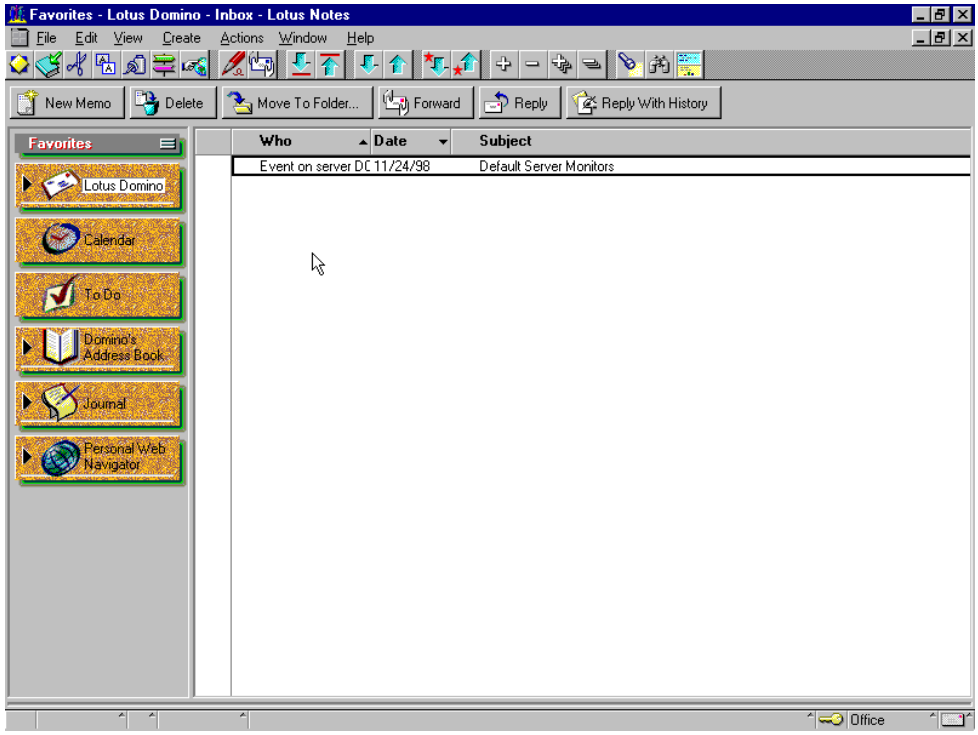


Figure 567. Connected to the Domino server on the AS23 system

The connection to the Domino HTTP server on TCP port 8081 is also confirmed to be correct, since the Notes client is able to browse the server as shown in Figure 568 on page 438.

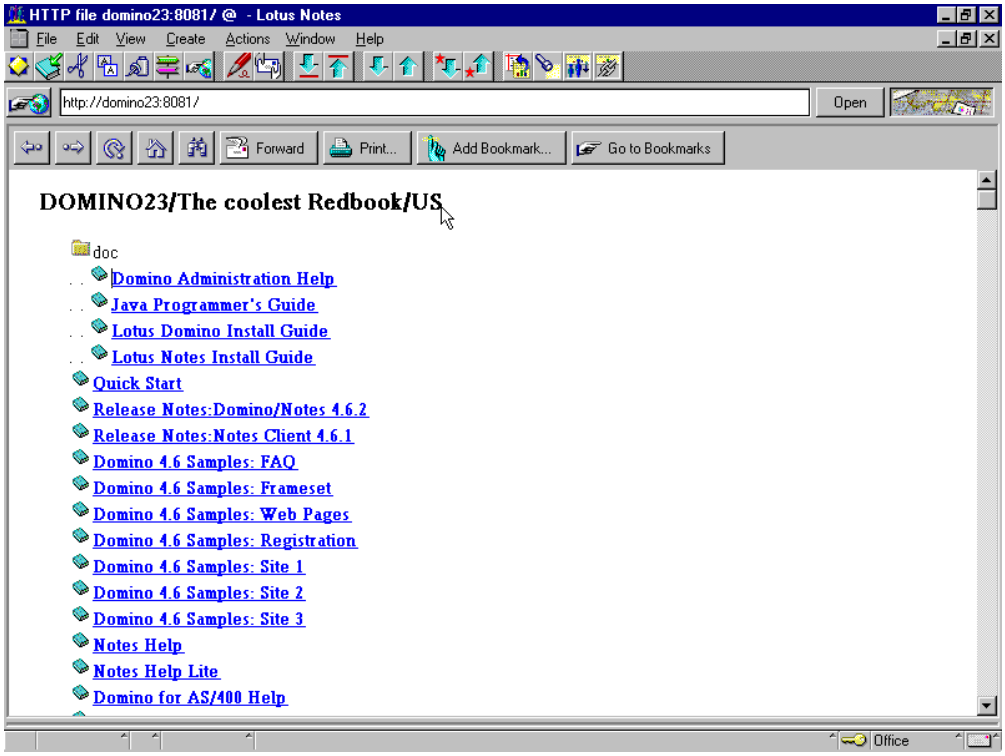


Figure 568. Connected to the Domino HTTP Server on the AS23 system

To check the rules, the Notes browser also tries to access the TCP port 80 where the IBM HTTP Server is running. Furthermore, a test is performed to start a Telnet session from the PC to the AS23 system (TCP port 25). The following example shows that these illegal requests are caught and logged in the journal:

```
SCEN1     A I   17DENY     6   10.1.1.200   4886   10.1.1.2       80
SCEN1     A I   17DENY     6   10.1.1.200   4887   10.1.1.2       23
```

10.9 Troubleshooting

This section provides you with some ideas for troubleshooting problems with the IP Packet Security function.

10.9.1 NAT and IP Packet Filtering: The correct order

Make sure that you understand in what order the NAT and IP Packet Filtering support is performed. Refer to 10.4, “Where and when NAT and IP Packet Filtering is done” on page 378, for a description.

10.9.2 The IP Packet Security journals

As described in 10.6.3, “Monitoring NAT and IP Packet Filtering” on page 391, the use of the journal is extremely useful. Depending on the logging level you specify in the rules, a lot of useful information is logged and is a good starting-point combined with catch-all rules.

10.9.3 Catch-all rules on all interfaces

As shown in 10.8.1, “Using standard filter rules” on page 398, it is a good idea to define rules that allow all traffic (Allow_all_...) and rules that deny all traffic and logs this to the journals (Do_not_allow...). This effectively provides you with information on which packets are being denied. You can extend the debugging by logging all packets, both valid and denied, to get a complete view of the packets flowing in and out of your system. This logging duplicates the communications trace function of OS/400.

10.9.4 Communications trace

The communications trace provides a low-level and complete picture of the IP datagrams on your system. Use the Start Communications Trace (`STRCMNTRC`) and Print Communications Trace (`PRTCMNTRC`) commands to collect and print the information.

10.9.5 Removing all rules

The best way to reset any rules to the system defaults (no rules applied at all) is to issue the Remove TCP/IP Table (`RMVTCPTBL`) command. This removes all the rules from all the interfaces. Make sure you have this command secured.

If you experience problems with an unstable processing of the rules or an unstable logging of the journal entries, try to stop the interface, remove the tables, and restart the interface again. Make sure that all the latest PTFs are applied as well.

10.10 OS/400 IP Packet Security and the firewall licensed program

To some extent, the OS/400 IP Packet Security function is a subset of the functionality provided by the Firewall for AS/400 licensed program. The native OS/400 IP Packet Security *cannot* replace the function of the Firewall for AS/400. Table 19 compares the functions provided.

Table 19. Firewall and native OS/400 NAT and IP filtering function

Function	OS/400	Firewall
IP Packet Filtering	Provide additional protection for a single AS/400 system, such as an Internet Web server or an intranet system with sensitive data. Protect a subnetwork of a corporate intranet when the AS/400 system is acting as a gateway (casual router) to the rest of the network. Control communication with a somewhat trusted partner over a private network where the AS/400 system is acting as a gateway.	Protect an entire corporate network from the Internet. Protect a large subnetwork with heavy traffic from the remainder of a corporate network.
NAT	Enable the connection of two private networks with incompatible addressing structures. Hide addresses in a subnetwork from a less trusted network.	Hide addresses of clients accessing the Internet. Use as an alternative to Proxy and SOCKS. Make services of a system in a private network available to clients on the Internet.

Function	OS/400	Firewall
Proxy Serving	Proxy at remote locations in a corporate network when a central firewall provides access to the Internet.	Proxy an entire corporate network when accessing the Internet.

10.11 Performance consideration

Implementing IP Packet Security will influence the performance of the TCP/IP protocol on the AS/400 system. If NAT or IP Packet Filtering IP Packet Security is enabled, the packets received and sent from the AS/400 system will be investigated and optionally processed. This overhead causes a degradation of the TCP/IP performance of your AS/400 system. If you enable logging of the requests as well, this will also have a negative impact on the performance, since requests are logged to the journals on the system.

Therefore, you should consider the type of IP Packet Security you need and should not include unnecessary rules, since they will undoubtedly have a negative influence of the TCP/IP performance on your system.

The following test demonstrates the influence of the IP Packet Security function. The test is not comprehensive, but gives you an idea of the influence. The test is performed:

- Without any IP Packet Security
- With IP Packet Security, but with no journaling
- With IP Packet Security and journaling.

The test was performed using the Verify TCP/IP Connection (PING) command, from the AS21 system to the AS23 system, using the AS22 system as an intermediate router. The filtering was configured on the AS22 system. The PING command sent 999 packets and 512 bytes to the destination.

The test results are shown in Table 20.

Table 20. Performance comparison

Test type	Minimum (ms)	Average (ms)	Maximum (ms)	% increase on the average column
No filter or NAT	14	15	32	n/a
Filter and NAT without journal	22	22	74	47%
Filter and NAT with journal	26	28	90	87%

Chapter 11. AS/400 VPN implementation

This chapter provides an overview of VPN implementation on the AS/400 system. The objective of this chapter is to provide a high level description of OS/400 VPN in V4R4 and to serve as an introduction to VPN on the AS/400 system. This is an excerpt from the redbook *AS/400 Internet Security: Implementing AS/400 Virtual Private Networks*, SG24-5404, which includes specific scenarios and detailed configuration examples. For updates on AS/400 VPN information, log on to the Web site at: `http://www.as400.ibm.com/vpn`

11.1 AS/400 VPN overview

In V4R3, IBM introduced VPN support on IBM Firewall for AS/400 (5769-FW1). For a description of the VPN implementation on the IBM Firewall for AS/400 product, refer to *IBM Firewall for AS/400: VPN and NAT Support,* SG24-5376.

In V4R4, IBM added VPN natively to OS/400 (5769-SS1). OS/400 VPN support is integrated in the operating system. It is based on the latest Internet Engineering Task Force (IETF) IPSec Drafts.

Note

The VPN support included in Firewall for AS/400 is based on the first set of Request For Comments (RFCs) for VPN. VPN support in the native OS/400 implementation is based on the new IPSec RFCs. These two sets of RFCs are not compatible with each other. Therefore, you cannot use the native VPN functions on the AS/400 system to communicate with VPN on Firewall for AS/400.

The following list summarizes the main features of OS/400 VPN support in V4R4:

- IPSec protocols
  - Authentication Header (AH)
  - Encapsulated Security Payload (ESP)
  - Internet Key Exchange (IKE)
- Manual connections
  - SPI values and cryptographic keys are predefined and manually refreshed
  - Use manual connections when the VPN partner does not support IKE
  - Configuration not supported by the VPN configuration wizard
- Dynamic key connections
  - IKE, AH, and ESP protocols supported, according to the latest Request For Comments (RFC) specifications
  - Support IKE protocol for dynamic key generation and refresh
  - Pre-shared key authentication
  - Configuration supported by the VPN configuration wizard
- Dynamic IP connections
  - Special case of Dynamic Key connection
  - Remote VPN partner is randomly assigned an IP address by an Internet service provider (ISP)

  - Remote VPN partner initiates the connection
  - Configuration supported by the VPN configuration wizard
- Layer 2 Tunneling Protocol (L2TP) connections
  - Provides a virtual Point-to-Point Protocol (PPP) tunnel across a public network, allowing a private corporate address space to be extended out to a remote client
  - Primarily used in remote access scenarios
  - AS/400 supports L2TP Network Server (LNS) and L2TP initiator functions
- Virtual Private Network Address Translation (VPN NAT)

  The AS/400 version of NAT that can be used in conjunction with VPN (conventional NAT and VPN are not compatible)

OS/400 VPN support has been certified by the International Computer Security Association (ICSA). Products that become ICSA certified have met a definable quantitative level of risk reduction against a known set of threats. The ICSA IPSec certification is primarily focused on testing compliance with the specifications, which also implies interoperability with other compliant solutions. For information about ICSA certification, see the Web site at: `http://www.icsa.net`

For AS/400 VPN performance test results, refer to *AS/400 Performance Capabilities Reference V4R4*, SC41-0607.

11.2 VPN software prerequisites

IBM makes the native VPN support available to AS/400 customers in V4R4 at no extra charge. The following software is required to configure OS/400 VPN:

- OS/400 V4R4 (5769-SS1)
- Digital Certificate Manager (DCM) (5769-SS1 option 34)
- Client Access Express for Windows (5769-XE1)
- IBM Cryptographic Access Provider (5769-AC2, or AC3)

  Note: The IBM Cryptographic Access Provider products come in three versions:

  - 5769-AC1 (40-bit encryption, exportable, *not* supported by VPN)
  - 5769-AC2 (56-bit encryption, exportable)
  - 5769-AC3 (128-bit encryption, available in U.S. and Canada)

Recent regulations allow U.S. software vendors to export 128-bit encryption products to banks, financial institutions, medical service providers, insurance companies, and subsidiaries of U.S. companies in 45 countries with the approval of the U.S. Department of Commerce. Check with the IBM U.S. Export Regulation Office for the latest information.

Note

OS/400 VPN support can dynamically determine the cryptographic capabilities of the system and only use currently supported algorithms. This means that, if 5769-AC2 is installed, the highest security available is Data Encryption Standard (DES). If 5769-AC3 is installed, the highest security available is 3DES. The negotiations for the SAs will not negotiate “down”, unless the policy allows it. If Highest Security, Lowest Performance is selected in the VPN configuration wizard for two AS/400 systems, one with 5769-AC2 and the other one with 5769-AC3 system, incompatible policies will result. The key and data policy transforms configured by the wizard must be modified for the negotiation to succeed.

11.3 AS/400 VPN components

AS/400 VPN support consists of the following components:

- VPN configuration GUI (part of Operations Navigator)
- VPN new connection wizard
- VPN server jobs
- VPN policy database
- IP packet filtering with ACTION = IPSEC
- Control Language (CL) commands
- Traces and logs for problem determination

The following sections briefly introduce each of these components.

11.3.1 VPN graphical user interface (GUI)

AS/400 Operations Navigator provides a powerful graphical user interface for Windows 95, 98, and NT PC clients to configure, manage, and administer your AS/400 system.

AS/400 VPN requires the Operations Navigator’s Network component. This component provides the Virtual Private Networking GUI for you to configure and manage VPN connections. IP Packet Security GUI is also part of Operations Navigator. It is required to configure the IP filters needed for VPN connections.

The Point-to-Point Connection Profiles configuration GUI in Operations Navigator has been enhanced to include L2TP.

11.3.2 New connection wizard

The New Connection Wizard guides you through a simple step-by-step configuration process. You input minimum information about your VPN environment, and the wizard takes over the complex configuration tasks. You can configure the following scenarios using dynamic key connections:

- Host to hosts
- Gateway to host
- Host to gateway
- Gateway to gateway

Dynamic IP Users refers to a VPN connection where the initiator is randomly assigned IP addresses (no fixed IP address). The wizard can configure the following Dynamic IP User connections:

- Gateway to Dynamic IP Users
- Host to Dynamic IP Users

To simplify the task of configuring VPN connections, you should always start by configuring with the wizard, customizing the individual objects later if needed.

The wizard does not support the configuration of Manual Connections and L2TP connections. You must use the VPN configuration GUI to configure those connection types.

11.3.3 CL commands

There are no green-screen commands available for VPN configuration. The following OS/400 CL commands are related to VPN management and troubleshooting:

- `STRTCPSVR SERVER(*VPN)` to start the VPN server jobs
- `ENDTCPSVR SERVER(*VPN)` to end the VPN server jobs
- `TRCTCPAPP APP(*VPN)` used by service personnel to collect VPN trace information
- `TRCTCPAPP APP(*L2TP)` used by service personnel to collect L2TP trace information

11.3.4 VPN and L2TP server jobs

You must start the VPN server jobs before activating VPN connections. The VPN server jobs run in the QSYSWRK subsystem. These jobs include:

- **QTOKVPNIKE**: This is the Virtual Private Networking key manager job. The VPN key manager listens to UDP port 500 to perform the Internet Key Exchange (IKE) protocols.
- **QTOVMAN**: This is the VPN connection manager job. The related job log contains messages for every connection attempt that fails.

The L2TP jobs are:

- **QTPPPCTL**: PPP control job. Starts when a virtual line (L2TP, initiator, or terminator) is started.
- **QTPPPL2TP**: Layer Two Tunneling Protocol (L2TP) manager job. If you have problems setting up an L2TP tunnel, look for messages in this job log.
- **QTPPPL2SSN**: L2TP session jobs. These are pre-started jobs that, by default, run in QSYSWRK. You can specify another subsystem for these jobs in the virtual line, subsystem configuration.

11.3.5 VPN policy database

The VPN configuration and policy information is stored in the VPN policy database. The VPN policy database in the QUSRSYS library consists of the objects shown in Table 21.

Table 21. VPN policy database objects

Object	Type	Library	Attribute
QATOVDAH	*FILE	QUSRSYS	PF
QATOVDCDEF	*FILE	QUSRSYS	PF
QATOVDDFLT	*FILE	QUSRSYS	PF
QATOVDDSEL	*FILE	QUSRSYS	PF
QATOVDESP	*FILE	QUSRSYS	PF
QATOVDIID	*FILE	QUSRSYS	PF
QATOVDIPAD	*FILE	QUSRSYS	PF
QATOVDLID	*FILE	QUSRSYS	PF
QATOVDMCOL	*FILE	QUSRSYS	PF
QATOVDNATP	*FILE	QUSRSYS	PF
QATOVDN1	*FILE	QUSRSYS	PF
QATOVDPKEY	*FILE	QUSRSYS	PF
QATOVDRGRP	*FILE	QUSRSYS	PF
QATOVDR1	*FILE	QUSRSYS	PF
QATOVDSRVR	*FILE	QUSRSYS	PF
QATOVDUCP	*FILE	QUSRSYS	PF
QATOVD1PRP	*FILE	QUSRSYS	PF
QATOVD1SP	*FILE	QUSRSYS	PF
QATOVD1TRN	*FILE	QUSRSYS	PF
QATOVD2LST	*FILE	QUSRSYS	PF
QATOVD2PRP	*FILE	QUSRSYS	PF
QATOVD2SP	*FILE	QUSRSYS	PF
QATOVD2TRN	*FILE	QUSRSYS	PF
QTOVDVPKEY	*VLDL	QUSRSYS	
QTOVDVSKEY	*VLDL	QUSRSYS	
QTOVDBJRN	*JRN	QUSRSYS	

You must include the objects listed in Table 21 in your regular backup process.

11.3.6 IP Packet Security

IP Packet Security (IP filtering) is an integrated feature of OS/400 that was first introduced in V4R3. IP Packet Security allows you to implement basic IP Packet Filtering rules to control traffic flowing into and out of your AS/400 system. Initially, filters supported only DENY and PERMIT as Action type. In V4R4, the Action type IPSEC was added to support VPN-specific traffic.

Filter rules are an important part of the AS/400 VPN implementation. Filter rules are required to funnel traffic through the VPN connection, as well as allow IKE negotiations to occur. When you configure a VPN connection with the new connection wizard or the VPN configuration GUI, filters are not created. The configuration of filters is a separate task that you must perform using the Operations Navigator IP Packet Security GUI after configuring the VPN connection.

11.4 Layer Two Tunneling Protocol (L2TP) VPN support

L2TP tunnels can be configured independent of VPN. However, L2TP by itself does not afford the required level of security. We recommend that you always protect the L2TP tunnel with IPSec.

L2TP tunnels are configured through PPP profiles. If an L2TP tunnel is protected by IPSec, a VPN L2TP Connection configuration is also required on the AS/400 initiator.

L2TP support in OS/400 V4R4 includes the following features:

- L2TP Network Server (LNS)
  - In this role, the AS/400 system is the virtual PPP line terminator.
  - Voluntary and compulsory tunnel models are supported.
  - Dial-out or On-Demand functions are *not* supported.
- L2TP initiator
  - In this role, the AS/400 system is the virtual PPP line initiator in an L2TP voluntary tunnel and includes built-in L2TP Access Concentrator (LAC) functions.
  - The AS/400 system does not support the LAC functions required in an ISP environment.

To participate as a client in an L2TP compulsory tunnel, only the regular PPP support is required on the AS/400 system. In a compulsory tunnel, the LAC functions are provided by the ISP.

For more information on L2TP and its configuration on the AS/400 system, refer to *AS/400 Internet Security: Implementing AS/400 Virtual Private Networks*, SG24-5404.

11.5 Virtual Private Network Network Address Translation (VPN NAT)

Conventional Network Address Translation (NAT) allows you to hide internal network IP addresses from an untrusted network. You can use NAT to dynamically translate your internal network IP addresses to public IP addresses

for communicating with the untrusted network. This is sometimes referred to as *masquerading*.

Another objective of NAT is to resolve address conflicts. It allows two networks with overlapping address spaces to be connected without necessarily having to change one set of addresses.

Unfortunately, conventional NAT cannot be used with IPSec protocols, because:

- In tunnel mode, ESP encrypts the inner IP addresses. Therefore, they cannot be translated by NAT.
- AH authenticates inner and outer IP addresses. Therefore, they cannot be translated.
- Even in transport mode, where ESP does not encrypt or authenticate the IP addresses, the Security Associations (SAs) are defined in terms of the destination IP address. Therefore, it cannot be changed.

The AS/400 system provides a unique solution, Virtual Private Network Network Address Translation (VPN NAT). VPN NAT works differently from conventional OS/400 NAT, because it translates addresses before applying IKE and IPSec protocols. It compares to static NAT in OS/400 V4R3 in that the address translation is one-to-one, bi-directional, and fixed (in this case) for the length of the VPN connection.

VPN NAT should be used in the following situations:

- You do not trust the remote VPN partner, and you want to hide you internal network IP address.
- The IP addresses of the two VPN partner networks conflict.

For more information on VPN NAT, refer to *AS/400 Internet Security: Implementing AS/400 Virtual Private Networks*, SG24-5404.

Chapter 12. LDAP on the AS/400 system

This chapter covers the basics of setting up Lightweight Directory Access Protocol (LDAP) on the AS/400 system. LDAP Directory Services is the emerging Internet standard protocol for looking up information, such as e-mail addresses, phone numbers, and certificate information.

In addition to the LDAP server, AS/400 Directory Services includes an AS/400-based LDAP client and a Windows 9x/NT LDAP client. The AS/400-based LDAP client includes a set of APIs that can be used in OS/400 C programs. The Windows 9x/NT LDAP client also includes a set of APIs.

12.1 What a directory is

A *directory* is a listing of information about objects arranged in some order that gives details about each object. Common examples are a city telephone directory and a library card catalog. For a telephone directory, the objects listed are people; the names are arranged alphabetically, and the details given about each person are address and telephone number. Books in a library card catalog are ordered by author or by title, and such information as the ISBN number of the book publication is given.

In computer terms, a directory is a specialized database, also called a data repository, that stores typed and ordered information about objects. A particular directory may list information about printers (the objects) consisting of typed information, such as location (a formatted character string), speed in pages per minute (numeric), print streams supported (for example PostScript or ASCII), and so on.

Directories allow users or applications to find resources that have the characteristics needed for a particular task. For example, a directory of users can be used to look up a person's e-mail address or fax number. A directory can be searched to find a nearby PostScript color printer. Or a directory of application servers can be searched to find a server that can access customer billing information.

The terms white pages and yellow pages are sometimes used to describe how a directory is used. If the name of an object (person, printer) is known, its characteristics (phone number, pages per minute) can be retrieved. This is similar to looking up a name in the white pages of a telephone directory. If the name of a particular individual object is not known, the directory can be searched for a list of objects that meet a certain requirement. This is like looking up a listing of hairdressers in the yellow pages of a telephone directory. However, directories stored on a computer are much more flexible than the yellow pages of a telephone directory because they can usually be searched by specific criteria, not just by a predefined set of categories.

12.2 Differences between directories and databases

A directory is often described as a database, but it is a specialized database that has characteristics that set it apart from general purpose relational databases. One special characteristic of directories is that they are accessed (read or

searched) much more often than they are updated (written). Hundreds of people might look up an individual's phone number, or thousands of print clients might look up the characteristics of a particular printer, but the phone number or printer characteristics rarely change.

Because directories must be able to support high volumes of read requests, they are typically optimized for read access. Write access may be limited to system administrators or to the owner of each piece of information. A general purpose database, on the other hand, needs to support applications, such as airline reservation and banking, with high update volumes.

Because directories are meant to store relatively static information and are optimized for that purpose, they are not appropriate for storing information that changes rapidly. For example, the number of jobs currently in a print queue probably should not be stored in the directory entry for a printer because that information would have to be updated frequently to be accurate. Instead, the directory entry for the printer could contain the network address of a print server. The print server could be queried to learn the current queue length if desired. The information in the directory (the print server address) is static, where the number of jobs in the print queue is dynamic.

Another important difference between directories and general purpose databases is that directories may not support transactions (some vendor implementations, however, do). Transactions are all-or-nothing operations that must be completed in total or not at all. For example, when transferring money from one bank account to another, the money must be debited from one account and credited to the other account in a single transaction. If only half of this transaction completes, or someone accesses the accounts while the money is in transit, the accounts will not balance. General-purpose databases usually support such transactions, which complicates their implementation.

Because directories deal mostly with read requests, the complexities of transactions can be avoided. If two people exchange offices, both of their directory entries need to be updated with new phone numbers, office locations, and so on. If one directory entry is updated, and then another directory entry is updated, there is a brief period during which the directory will show that both people have the same phone number. Because updates are relatively rare, such anomalies are considered acceptable.

The type of information stored in a directory usually does not require strict consistency. It may be acceptable if information, such as a telephone number, is temporarily out of date. Because directories are not transactional, it is not a good idea to use them to store information sensitive to inconsistencies, like bank account balances.

Because general-purpose databases must support arbitrary applications, such as banking and inventory control, they allow arbitrary collections of data to be stored. Directories may be limited in the type of data they allow to be stored (although the architecture does not impose such a limitation). For example, a directory specialized for customer contact information might be limited to storing only personal information such as names, addresses, and phone numbers. If a directory is extensible, it can be configured to store a variety of types of information, making it more useful to a variety of programs.

Another important difference between a directory and a general-purpose database is in the way information can be accessed. Most databases support a standardized, very powerful access method called Structured Query Language (SQL). SQL allows complex update and query functions at the cost of program size and application complexity. LDAP directories, on the other hand, use a simplified and optimized access protocol that can be used in slim and relatively simple applications.

Because directories are not intended to provide as many functions as general-purpose databases, they can be optimized to economically provide more applications with rapid access to directory data in large distributed environments. Because the intended use of directories is restricted to a read-mostly, non-transactional environment, both the directory client and directory server can be simplified and optimized.

12.3 Directory Services (LDAP) overview

AS/400 Directory Services provides a Lightweight Directory Access Protocol (LDAP) server on an AS/400 system. LDAP is an industry standard that is essentially used to exchange “directory information” among systems supporting different directory information architectures. LDAP is gaining popularity as a directory service for both Internet and non-Internet applications.

LDAPs capabilities are also expanding over time. It is being supported by a growing number of software vendors and is being incorporated into a growing number of applications. For example, the two most popular Web browsers, Netscape Navigator or Communicator and Microsoft Internet Explorer, support LDAP functionality as a base feature.

Common uses of LDAP directories include online telephone directories and e-mail directories. Another use of the LDAP directory information could be the configuration of the HTTP Server for AS/400 (5769-DG1) to use the LDAP directory to validate HTTP server users (user ID and password) instead of other validation techniques, such as HTTP server validation lists or SSL digital certificates.

While a complete description of LDAP is beyond the scope of this redbook, we include some overview information and show some of the AS/400 LDAP configuration screens in this section. For more detailed information on OS/400 LDAP configuration and management, please refer to:

- The AS/400 Information Center: `http://www.as400.ibm.com/infocenter`

  You can use the search word LDAP, or you can select **Networking->AS/400 Directory Services (LDAP)**.

- The OS/400 LDAP Web site: `http://www.as400.ibm.com/ldap`

  This Web site has good explanations and references to other LDAP documentation, including the following redbooks:

- *Understanding LDAP*, SG24-4986

  This redbook provides general overview of origination and capabilities of LDAP

- *LDAP Implementation* Cookbook, SG24-5110

  This redbook provides detail planning information and some product specific examples.

The LDAP directory services follow a client/server model. One or more LDAP servers contain the directory data. The LDAP directory service model is based on entries (which are also referred to as objects). The information within an LDAP directory is organized as a hierarchical tree structure, commonly referred to as a *directory information tree* (DIT).

An LDAP client connects to an LDAP Server and makes a request. The server responds with a reply, or with a pointer (a referral) to another LDAP server. One example of an LDAP client would be the HTTP Server for AS/400 accessing the LDAP directory for user validation. An LDAP publishing agent can “publish” (send) LDAP directory entries to another LDAP server.

12.4 LDAP directory example

Each LDAP entry consists of one or more attributes, such as a name or address, and a type. The types typically consist of mnemonic strings, such as *cn* for common name or *mail* for e-mail address.

The example directory, shown in Figure 569, shows an entry for Tim Jones that includes *mail* and *telephoneNumber* attributes. Examples of other attributes include *fax*, *title*, *sn* (for surname), and *jpeg* photo.

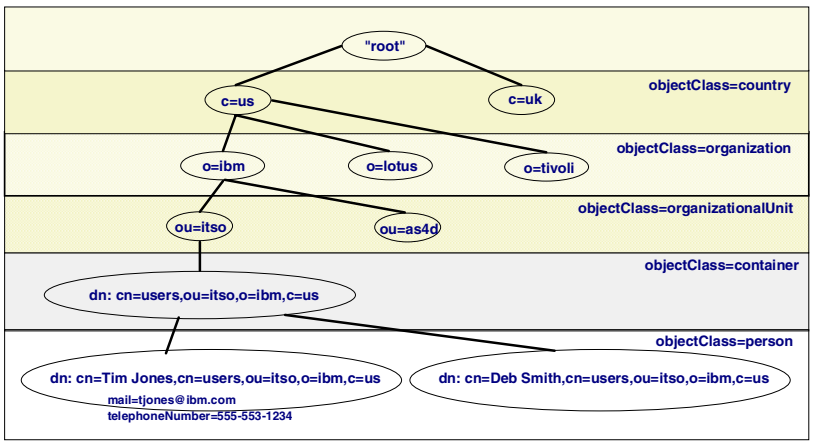


Figure 569. LDAP directory structure example

Each directory has a *schema*, which is a set of rules that determine the structure and contents of the directory. IBM has a defined schema that is shared across IBM LDAP servers.

Each directory entry has a special attribute called *objectClass*. This attribute controls which attributes are required and allowed in an entry. In other words, the

values of the objectClass attribute determine the schema rules the entry must obey.

LDAP directory entries are arranged in a hierarchical structure that reflects political, geographic, or organizational boundaries. Entries that represent countries appear at the top of the hierarchy. Entries representing states or national organizations occupy the second level down in the hierarchy. The entries below that can represent people, organizational units, printers, documents, or other items. Refer to Figure 569 as we use some of these attributes in our examples in this section.

LDAP refers to entries with *Distinguished Names* (DNs). Distinguished names consist of the name of the entry itself as well as the names, in order from bottom to top, of the objects above it in the directory. For example, the complete DN for the Tim Jones entry in Figure 569 is: *cn*=Tim Jones, cn=users, ou=itso, *o*=ibm, *c*=us. Relatively high level DNs are called *parent Distinguished Names*. For example, cn=users,ou=itso,*o*=ibm, *c*=us is the parent DN for the DN cn=Tim Jones, and Dn cn=Deb Smith. Each entry has at least one attribute that is used to name the entry. This naming attribute is called the Relative Distinguished Name (RDN) of the entry. In the example above, cn=Tim Jones names the entry, so it is the RDN.

All of these attributes can be in any order you choose to specify the hierarchy you wish. For example, one enterprise may use ou=itso, o=ibm, c=us and another may use o=ibm, c=us. However, a user of a specific directory must understand the order of attributes defined for that directory to make use of the directory.

To give an LDAP server the capability to manage an LDAP directory, you specify the highest level parent distinguished names in the configuration of the server. These distinguished names are called *suffixes*. The server can access all objects in the directory that are below the specified suffix in the directory hierarchy. For example, if an LDAP server contained the directory shown in Figure 569, it would need to have the suffix ou=itso,*o=ibm, c=us* specified in its configuration in order to be able to answer client queries regarding “Tim Jones”.

One or more *suffixes* are required to define the naming context within the directory tree. One suffix is pre-defined for the LDAP directory server, *cn=localhost.* This local host suffix is enabled to contain objects related to directory replication, if you set up a network where one server contains a replica (copy) of data on this server.

Because LDAP is a directory service, rather than a database, the information in an LDAP directory is usually descriptive, attribute-based information. LDAP users generally read the information in the directory more often than they change it. Updates are typically simple all-or-nothing changes.

To configure a directory server and access the directory, you have to define an administrator and the administrator’s password.

You can share or “move” directory data across AS/400 LDAP servers and, in many cases, another non-AS/400 LDAP server. The ways to move the data include:

- Exporting an LDIF file (LDAP Data Interchange Format) file
- Importing an LDIF

- Setting up a new replica of the directory server
- Publishing AS/400 information to a directory server

For a more in depth discussion about LDAP, directories and concepts, refer to *Understanding LDAP*, SG24-4986. Or, go to the University of Michigan LDAP home page at: `http://www.umich.edu/~dirsvcs/ldap/`

This university has been, and still is, an important contributor in the development of LDAP and can be considered a reliable, neutral source for extensive information and program source code for LDAP servers and clients.

This University of Michigan LDAP page contains, among others, links to online LDAP documentation from the UMICH and others and downloadable software, most of which as source code.

This university's LDAP server code, a C language SDK, and other links to documentation and LDAP mailing lists can be found on its Web site.

Search for IBM's C and Java SDKs at: `http://www.ibm.com/Help`

On this site, type: `LDAP` and `Toolkit`

Netscape offers a C and a Java SDKs at:
`http://developer.netscape.com/software/sdks/index.html`

Look for the Directory SDK.

12.5 Planning your directory

Before you install AS/400 Directory Services and begin to configure your LDAP directory, you should take a few minutes to plan the directory. Important points to consider include:

- Organizing the directory. Plan the structure of your directory. Determine what suffixes and attributes your server will require.
- Deciding how large your directory will be, so that you can estimate how much AS/400 storage you need.
  - The size of the directory depends on the number of attributes in the server's schema.
  - The number of entries on the server.
  - The type of information that you store on the server.

  Services schema (which contains approximately 150 attributes) requires approximately 5 MB of storage space. A directory that uses the default schema, and which contains 1000 entries of typical employee information, requires about 13 MB of storage space. This number would vary depending on the exact attributes that you used. It would also increase greatly if you stored large objects, such as pictures, in the directory.
- Deciding what security measures you will take. AS/400 Directory Services supports the use of Secure Sockets Layer (SSL) and Digital Certificates for communication security. AS/400 Directory Services also allow you to control access to directory objects with access control lists (ACLs). You can use OS/400 security to protect the directory database.

12.6 LDAP on OS/400

You can configure the AS/400 to be a Directory Services (LDAP) server. You can specifically place entries into the directory through an LDIF file. This server can also receive “published LDAP entries” from a publishing agent for another directory server. You can configure an AS/400 LDAP Publishing agent to publish OS/400 system distribution directory entries to an LDAP directory server.

AS/400 support of LDAP requires the installation of OS/400 (5769-SS1) option 32, Directory Services. This support includes:

- OS/400 LDAP server, based on LDAP Version 2
- OS/400 LDAP publishing server, based on LDAP Version 2
- OS/400 LDAP client, based on LDAP Version 2
  Note: The Windows 95/NT LDAP client is based on LDAP Version 3.

To perform LDAP-related functions, you must configure and start the OS/400 LDAP Directory Server and, optionally, the Publishing server. Because LDAP is an industry standard, all LDAP servers share many basic characteristics. However, due to implementation differences, they are not all fully compatible with each other. The LDAP server provided by AS/400 Directory Services is closely compatible with LDAP servers available from IBM on other platforms. OS/400 LDAP server support may not be as compatible with other non-IBM LDAP servers.

Access security to LDAP entries is provided through LDAP access control lists (ACL). There is one ACL for each LDAP object. You could restrict access to only those users within the ACL. You can also use SSL with LDAP Directory Services to exchange encrypted directory data.

You can use OS/400 LDAP Directory Services with LDAP-enabled applications, such as mail applications that look up e-mail and HTTP server user validation.

12.7 AS/400 LDAP Secure Sockets Layer (SSL) support

To make communications with your LDAP directory server more secure, AS/400 Directory Services can use Secure Sockets Layer (SSL) security. If want to use SSL with AS/400 Directory Services, you may need to obtain additional software for Windows 95/NT from IBM. You need this software if you want to do any of these tasks:

- Configure and administer AS/400 Directory Services from your workstation using an SSL connection. This includes tasks that you perform from Operations Navigator.
- Publish AS/400 user information to an LDAP directory from Operations Navigator using an SSL connection.
- Use an SSL connection with applications that you create with the Windows 95/NT client application program interfaces (APIs).

If you see the error message `Secure sockets layer (SSL) initialization failed with reason code 0`, you also need this software. For the latest information on obtaining this software and configuring AS/400 Directory Services to use SSL from your workstation, refer to the Client Access Web page at:
`http://www.as400.ibm.com/clientaccess`

Or, see Informational APAR II11440 *LDAP Client Support for SSL in Client Access V3R2M0*.

You can use SSL to communicate with LDAP clients, as well as with replica LDAP servers. SSL is the standard for Internet security. To use SSL, you must have Digital Certificate Manager (DCM), option 34 of OS/400, installed on your system. DCM provides an interface for you to create and manage digital certificates and key ring files. See Chapter 3, “SSL security on the AS/400 system” on page 41, for information on using DCM and for detailed explanations of SSL concepts.

12.8 LDAP directory referrals

Referrals allow LDAP directory servers on the AS/400 system to work in teams. If the distinguished name (DN) that a client requests is not in one directory, the server can automatically send (refer) the request to any other LDAP server.

AS/400 Directory Services allows you to use two different types of referrals. You can specify a default referral server (see 12.10.1, “Specifying a server for directory referrals” on page 462) where the LDAP server refers clients whenever a DN is not in the directory. You can also use your LDAP client to add entries to the directory server that have the objectClass “referral”. This allows you to specify referrals that are based on what specific DN a client requests. With this method, you can have referrals from one LDAP server to multiple LDAP servers.

Note

With AS/400 Directory Services, referral objects must contain only a distinguished name (dn), an objectClass (objectClass), and a referral (ref) attribute.

Referral servers are closely related to replica servers. Because data on replica servers cannot be changed from clients, the replica refers any requests to change directory data to the master server.

12.9 Replica LDAP directory servers

The information stored on replica LDAP directory servers is identical to the information on your main, or *master*, LDAP directory server. There are two principal benefits to having one or more replicas of your LDAP directory:

- Replicas make directory searches faster. Instead of having all clients direct search requests to a single master server, you can split requests between the master server and the replica servers.
- Replicas provide a backup to the master server. If the master server is unavailable, a replica can still fulfill search requests and provide access to directory data.

Replica servers are read-only. When an authorized user attempts to change an entry on a replica server, it refers the request to the master directory server.

12.10 LDAP configuration

The Operations Navigator lets you configure the LDAP server on the AS/400 system. The configuration task is one of the network server tasks. The LDAP configuration requires these steps:

1. To configure the LDAP server, start with **Network->Servers->TCP/IP-> Directory** icons as shown in Figure 570. The V4R3 Operations Navigator (Client Access V3R2) looks different than V4R2 (Client Access V3R1Mx) in terms of the network configurations. All TCP/IP applications are consolidated under the *TCP/IP* icon.

   There is no green-screen interface for configuring the directory server. With green-screen commands, you can only start it using Start TCP/IP Server (`STRTCPSVR SERVER(*DIRSRV)`) command, or end it using End TCP/IP Server (`ENDTCPSVR SERVER(*DIRSRV)`) command.

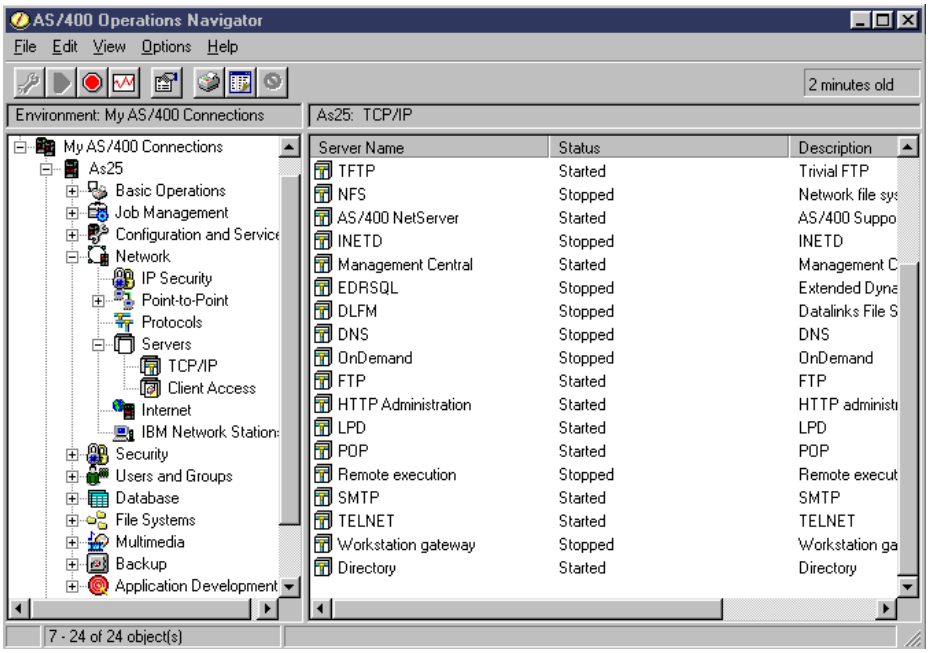


Figure 570. Operations Navigator LDAP Administration

2. Right-click **Directory**, and select **Configure** or **Reconfigure** from the pop-up menu.

   If this is the first time the LDAP server is being configured on your AS/400 system, the configuration wizard (Figure 571 on page 458) will be started to guide you through the configuration task.

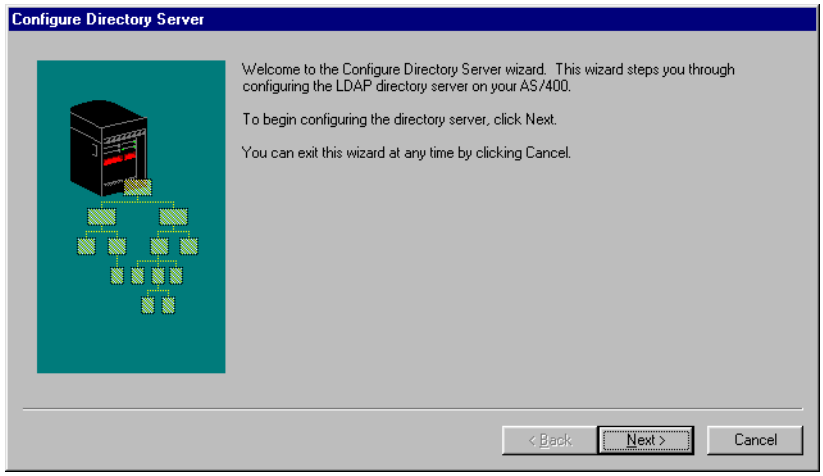


Figure 571. Operations Navigator LDAP configuration wizard (Part 1)

3. You may see the next display (Figure 572), which asks you for the relational database name used for this Directory server. This depends on your version and release of OS/400. In our example, we use ROCHESTER.

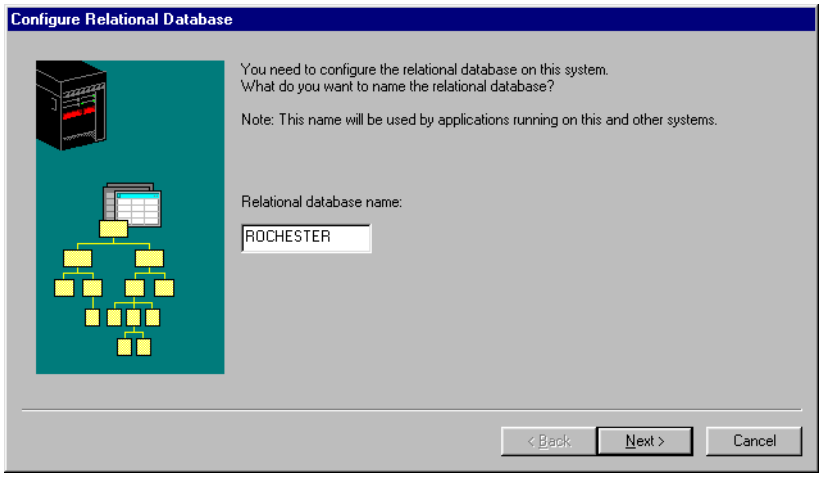


Figure 572. Operations Navigator LDAP configuration wizard (Part 2)

4. As shown in the display in Figure 573, enter the AS/400 library for the relational database. You should specify a library that will only be used by the Directory Server. If the library you specified does not exist, the wizard will ask you for confirmation to create this library.

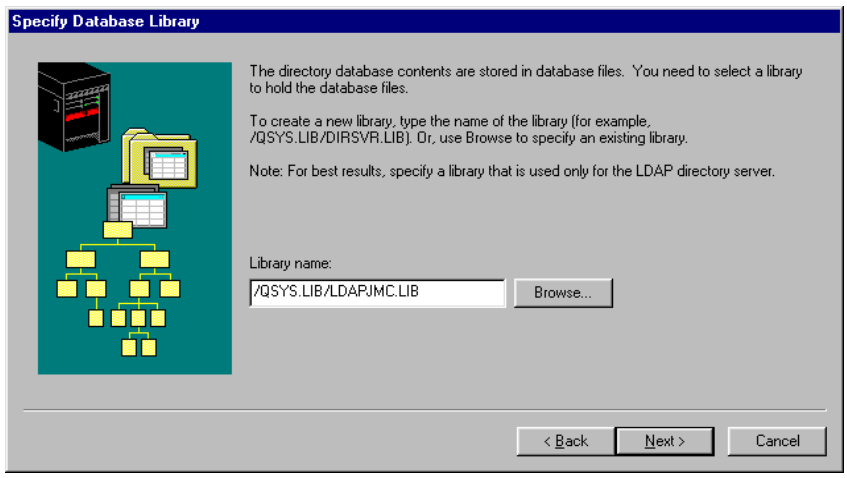


Figure 573. Operations Navigator LDAP configuration wizard (Part 3)

5. On the next display (Figure 574), specify the distinguished name (DN) of the administrator for the directory server and a password. This string of identifying attributes uniquely locates the entry within your directory server directory. Assigning the Administrator name and associated password is very important for any changes made later, for clients to connect to the directory server, and if you later want to configure the Publishing support. *Remember this Administrator name and password!* Click **Next**. The display shown in Figure 575 on page 460 appears.

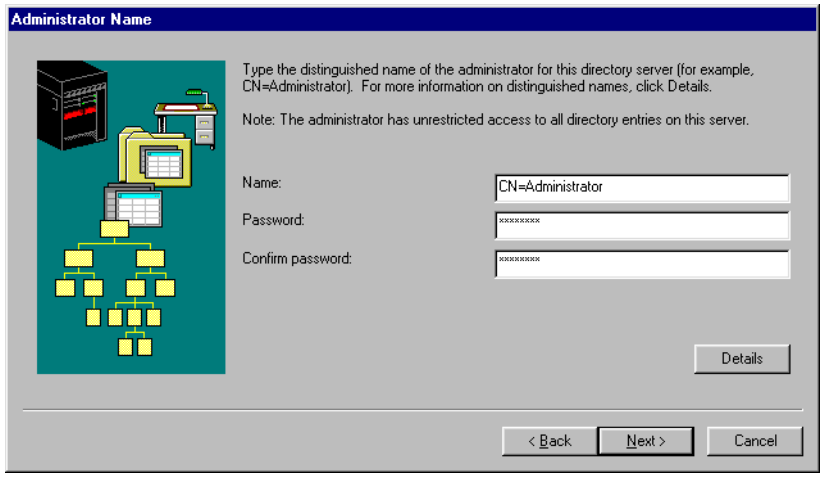


Figure 574. Operations Navigator LDAP configuration wizard (Part 4)

6. Enter the directory suffixes at the next screen (Figure 575 on page 460). You must configure at least one suffix in addition to the cn=localhost (for example, `ou=itso,o=ibm,c=us`). Type the directory suffix, and click **Add**.

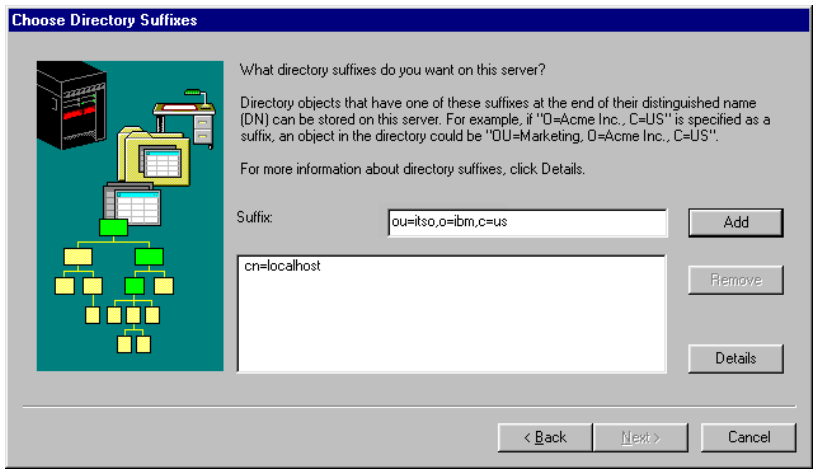


Figure 575. Operations Navigator LDAP configuration wizard (Part 5)

This suffix information is used when adding a new entry into the LDAP directory and later when searching the directory. Remember this suffix information and the hierarchical order in which you enter it. The order of the suffix attributes is important. Once you have organized your directory structure, the DN attributes must always be specified in the same order because a DN represents a path through the directory tree.

After entering the suffix information, click **Add**. Although it is not shown in our example, any new suffix information appears in the window that shows current suffixes under the previous suffix entry, for example cn=localhost. You can enter additional suffix information multiple times. Remember, it is very important that the suffix hierarchy and the characters entered are well planned within your company if directory entries are to be shared or published. You must enter them in the hierarchical search order you want.

Suffix tip

It is important that the suffixes do not overlap. For example, you should not have both “o=ibm,c=us” and “ou=itso,o=ibm,o=us.” This is because “ou=itso,o=ibm,c=us” is *already below* “o=ibm,c=us.”

When you are finished, click **Next**. The display shown in Figure 576 appears.

7. On the next display, you can specify if you want the LDAP server to be started on each TCP/IP start, or manually (Figure 576).

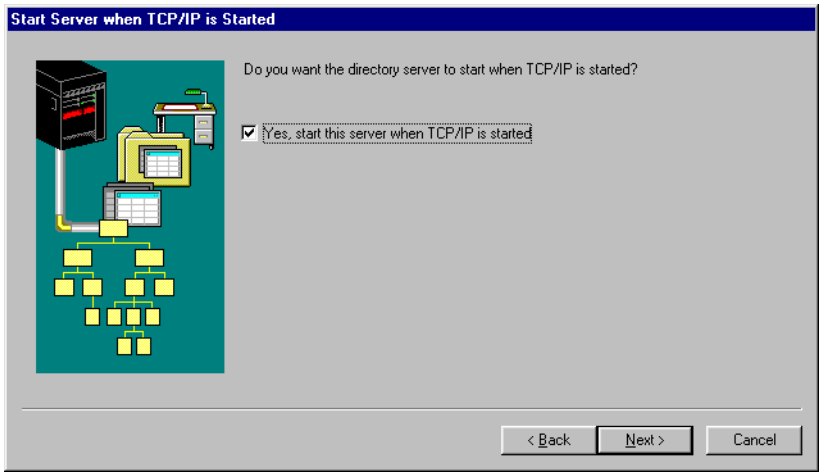


Figure 576. Operations Navigator LDAP configuration wizard (Part 6)

8. The Configuration Summary screen appears as shown in Figure 577. Verify the settings for the Directory Server. If needed, you can go back and correct configuration mistakes by clicking the **Back** button. Click **Finish** to complete the configuration.

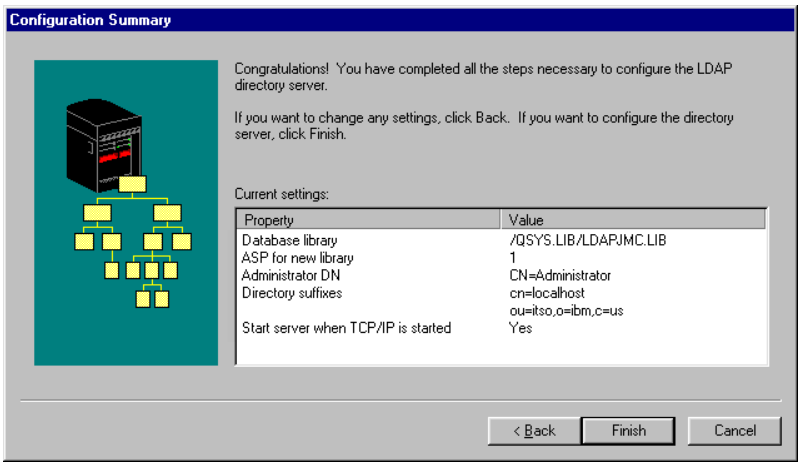


Figure 577. Operations Navigator LDAP configuration wizard (Part 7)

When the wizard finishes, your LDAP directory server has a basic configuration. The wizard interface does not present to you some of the Directory Services server parameters, such as allowing the directory to be updated, IP port number to use, if SSL encryption is to be used, or LDAP “performance parameters”. You can view and change existing directory configuration information and add new information by right-clicking **Properties** on the Directory server. Depending on the value, an add or change may require the server to be in “stopped” status.

One important consideration is that if you already have, or are planning to have, Domino LDAP server running on your system with both the Domino LDAP server and the LDAP server default set to the same port number, 389, you have to change the port number for one of these servers.

After you finish the configuration, you can start the Directory server. During the first start, all needed files are created in the relational database library you specified during configuration.

12.10.1 Specifying a server for directory referrals

Follow these seven simple steps to configure your referral server:

1. In Operations Navigator, expand **Network**.
2. Expand **Servers**.
3. Click **TCP/IP**.
4. Right-click **Directory**, and then select **Properties**.
5. In the Server field, specify the name of the referral server.
6. If the referral server does not use the default port, specify the correct port number in the **Port** field.
7. Click **OK**.

12.10.2 Setting up a new replica of the directory server

You can set up replicas of the LDAP directory server to directory servers on other AS/400 systems. Because the LDAP directory server on the AS/400 server is compatible with LDAP servers on other IBM platforms, you can also:

- Set up a directory server on another IBM platform to be a replica of the directory server on the AS/400 system.
- Set up the directory server on the AS/400 system to be a replica of a directory server on another IBM platform.

In either of these cases, follow the documentation for the other IBM platform for setting up a master or replica server on that platform. AS/400 Directory Services does not support replication with non-IBM LDAP servers.

Follow these steps to set up a new replica of the directory server:

1. If you have not already done so, install and configure both the master server and the replica server.
2. Stop the master server.
3. (Optional) Set up LDAP data for initial replication. You can skip this step if you do not have any initial data that you want to transfer to the replica server from the master server.
4. (Optional) Move LDAP data to the master server. Skip this step if one of the following applies to your replica server:
   - It is a new LDAP directory server.
   - It does not contain data that you want to continue to maintain.
5. Set up the new replica server. See 12.10.3, “Setting up the new replica” on page 463.
6. Set up the master server to have a new replica. See 12.10.4, “Setting up the master server to have a new replica” on page 464.

7. Make sure the master server is allowing updates:
   a. In Operations Navigator, expand the AS/400 system on which the master directory server runs.
   b. Expand **Network**.
   c. Expand **Servers**.
   d. Click **TCP/IP**.
   e. Right-click **Directory**, and select **Properties**.
   f. If it is not already checked, check **Allow directory updates**.

These instructions assume that both the master server and the replica servers are on AS/400 systems that you manage from Operations Navigator on the same PC. If you are managing your AS/400 systems from separate PCs, you can move between two PCs to perform this task. If either the master or replica server is running on an IBM platform other than the AS/400 system, refer to the documentation for that platform to set up that server.

12.10.3 Setting up the new replica

Follow these steps to set up the new replica server:

> **Note**
>
> The replica server must be configured and stopped before you perform this procedure.

1. In Operations Navigator, expand the AS/400 system on which the replica directory server runs.
2. Expand **Network**.
3. Expand **Servers**.
4. Click **TCP/IP**.
5. If the server is not already stopped, stop it now. Refresh the status of the servers until the status is *Stopped*.
6. Right-click **Directory**, and select **Properties**.
7. Select **Server is a replica**.
8. In the *Name used for updates* field, specify a name for the master server to use when it logs on to the replica server when it performs updates.
9. Click the **Password** button next to the Name used for updates field. Enter a password for the master server to use when it logs on to the replica server to perform updates.

> **Note**
>
> You should make note of this password and the name you entered in step 8. You will need them when you set up the master server for replication.

10. In the Name field of the Referral Server frame, enter the name of the master server.

11. If your master server uses a port other than the default, enter this port number in the Port field of the Referral Server frame.
12. Click the **Suffixes** tab. If the suffix that you want to replicate is not on the list, add it.
13. (Optional) If you want to use Secure Sockets Layer (SSL) when replicating, click the **Network** tab.
    a. Select **Secured**, **using SSL**. Or, select **Both** secured and not secured.

    Note

    You need to obtain additional software, as described in 12.7, "AS/400 LDAP Secure Sockets Layer (SSL) support" on page 455, if you select Secured using SSL.

    b. In the Key ring *file* field, enter the name of your key-ring file.
14. Click **OK**

12.10.4 Setting up the master server to have a new replica

Follow these steps to set up the master server to have a new replica:

Note

You must have configured and started the master server before you perform this procedure.

1. In Operations Navigator, expand the AS/400 system on which the master directory server runs.
2. Expand **Network**.
3. Expand **Servers**.
4. Click **TCP/IP**.
5. Right-click **Directory**, and select **Properties**.
6. If it is not already checked, check **Allow directory updates**.
7. Click **OK**.
8. Stop, and then restart the LDAP directory server. Refresh the status of the servers until the status is *Started*.
9. Click the **Replicas** tab. Operations Navigator may prompt you to enter connection information. Enter this information, and then click **OK**.
10. Click **Add**.
11. In the Server field, enter the name of the replica server.
12. In the Connect *as* field, enter the name you specified in step 8 when you set up the replica server.
13. Click **Password**, and enter the password you specified in step 9 when you set up the replica server.
14. If you want to use Secure Sockets Layer (SSL) for replication, select **Secured, using SSL**.

Note

You need to obtain additional software, as described in 12.7, “AS/400 LDAP Secure Sockets Layer (SSL) support” on page 455, if you select this option.

15. If your server does not use the default port, enter the port number in the Port field.
16. If you do not want to update the replica server every time an entry on the master server changes, select **Time**. Then specify how often you want the master server to update the replica.
17. Click **OK**.
18. Click the **Suffixes** tab. If the suffix that you want to replicate is not on the list, add it.
19. Enable directory updates on each replica server:
    a. In Operations Navigator, expand the AS/400 system on which the replica directory server runs.
    b. Expand **Network**.
    c. Expand **Servers**.
    d. Click **TCP/IP**.
    e. Right-click **Directory**, and select **Properties**.
    f. If **Allow directory updates** is unchecked, check it.
    g. Click **OK**.
20. If each replica server is not already started, start it now.

Note

A server cannot be both a master server and a replica server.

When you import an LDIF file into the master server, this data is not replicated. If you need LDIF data replicated, you can do one of the following options:

- Import it to both systems.
- Use an LDAP client to do LDAP adds of the LDIF file data to the master server. These adds will then be replicated to the replica server.

12.10.5 Exporting and importing an LDAP LDIF file

An LDIF file can be used to interchange LDAP directory information among LDAP servers separate from the publishing function. By using the correct syntax within an LDIF file, new entries can be added to a new directory through the OS/400 QShell command program *ldapadd.* Ideally, you can export all or part of an existing LDAP directory into an LDIF file. That LDIF file can be imported to another LDAP directory server.

In this section, we show examples of the LDIF export process.

Right-click on the AS/400 directory server, and select **Tools** from the pull-down menu. This brings up the Export and Import options.

Select the **Export** function. A display similar to the one shown in Figure 578 appears.

Figure 578. AS/400 LDAP export LDIF file

Enter the location and file name to contain the directory information and whether you want the entire directory or a portion of the directory exported. In our case, we export the entire directory. Note the caution that this option “may take a long time“. During the export function, a window describing the process is shown.

The directory we exported had 53 entries, and it took less than 10 seconds to build the contents of file \ldap\las25ldif.ldf. Figure 579 shows the first set of records within our LDIF file, which includes the operational unit entry, the common name for users entry, and the first user entry for Adan.

Figure 579. LDAP LDIF file contents

This LDIF file can be copied to a diskette or tape or sent across a network with TCP/IP FTP. It can then be imported onto the target LDAP directory server.

If you were to use the OS/400 LDAP QShell `ldapadd` command to enter LDAP entries, you could use the syntax shown in the LDIF file to help you create correct entries. For example, enter the Start QShell (`STRQSH`) command, and then enter an `ldapadd` command such as:

```
ldapadd -h as25 -p 390 -D cn=Administrator -w l1dap -f /ldap/addentry.txt
```

Here, file *addentry.txt* contains the appropriate LDAP entry syntax. Refer to the AS/400 Information Center Web site for a description of LDAP and these commands. The site is located at: `http://www.as400.ibm.com/infocenter`

12.10.6 Adjusting performance of the LDAP directory server

You can adjust the performance of your LDAP directory server by changing any of the following points:

- The number of simultaneous connections that are allowed
- The size of searches
- The maximum time allowed for searches

You can also create indexes of attributes to improve search times. The amount of resources that the AS/400 system allocates for searching the LDAP directory

increases when you increase the number of maximum connections allowed. You can experiment with different maximum connection values to determine the value that works best for your situation.

To adjust the performance values of the directory server, follow these steps:

1. In Operations Navigator, expand **Network**.
2. Expand **Servers**.
3. Click **TCP/IP**.
4. Right-click **Directory**, and select **Properties**.
5. Click the **Performance** tab (Figure 580).

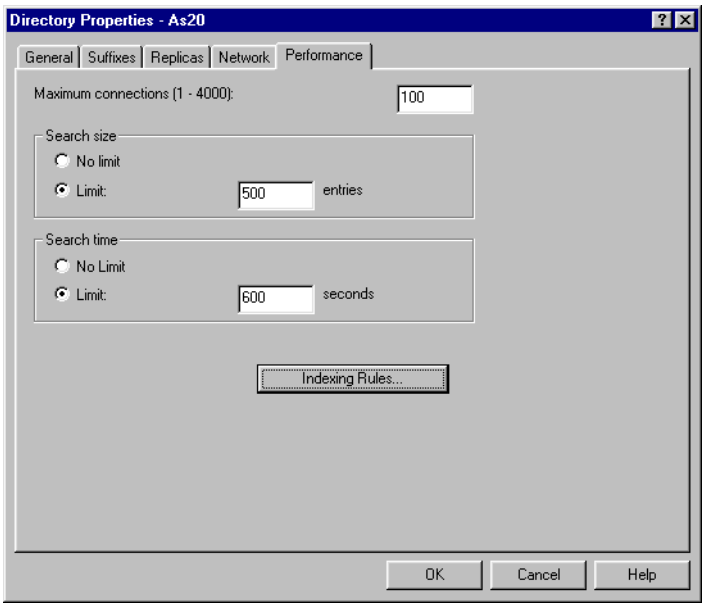


Figure 580. LDAP performance

12.11 Propagating AS/400 users to the directory

If you want your AS/400 users propagated to the AS/400 Directory Server, you can use a system API from the AS/400 command line or program, or use Operations Navigator.

We just described how to configure the Directory server on the AS/400 system. Now, you need to add entries into the LDAP directory. As stated earlier in this chapter, you can get new entries into the directory server in several ways:

- Configuring an LDAP Publishing agent server (to automatically update the designated Directory server when specifying to publish “users” or “computers” type entries).

- Importing an LDIF file that was exported; you can get to the OS/400 Directory Services Import or Export function in the pull-down menu option by right-clicking on a Directory server and selecting the **Tools** option.
- Entering an OS/400 provided LDAP command program from a QShell session (the `STRQSH` command).

12.11.1 Publishing directory information

By configuring the AS/400 publishing server to publish users or computers information, you initialize the LDAP directory with the current OS/400 system distribution directory entries. Then, each time an OS/400 system distribution directory entry is added, deleted, or changed (for example, with the Work with Directory Entry (WRKDIRE) command), the results of this change are published to the Directory server identified during configuration. You can add, change, or remove directory servers that are to be “published to”.

To configure the AS/400 server to automatically publish AS/400 information into an LDAP Directory server, complete these steps:

1. In Operations Navigator, right-click on your AS/400 system, for example As25, and select **Properties** for that system. Click the **Directory Services** tab. Click on the types of information that you want to publish, Users and/or Computers. Selecting **Users** and completing the publishing configuration successfully enables publishing of the system distribution directory for the first time and for any updates.
2. Click the **Configure** button.
3. Click the **Publish AS/400 information for** (Users or Computers) check box for Users.
4. In the Directory server field, enter the name of the LDAP directory server where you want to publish AS/400 information. This can be a remote Directory server or a local Directory server. In our example, we specify As25, the local server (Figure 581 on page 470).

OS/400 LDAP publishing tips

1. You can look in the job log of the publishing agent job (QGLDPUBA) to confirm that the system distribution directory has been published successfully. Look for message ID GLD0305.
2. When setting up the LDAP directory server and the LDAP publishing server, you may perform several "reconfigurations" and find that you cannot see all the OS/400 system distribution directory entries you intended to publish. A way to check to see if the distinguished name entries are actually in the LDAP directory is to use an AS/400 SQL Select statement against one of the OS/400 LDAP database files (tables) in the library you specified when configuring or reconfiguring the LDAP directory server. In our example, we used library LDAPJMC. For that library, use the following SQL statement to get an indication of the entries in the LDAP directory:

   ```
   SELECT * from ldapjmc/ldap_entry
   ```

LDAP is a very powerful function for storing and conveying directory information in a format that can be communicated across systems with differing directory

information structures or “schema”. However, you have to know some details of the LDAP architecture to properly implement LDAP within your organization and to exchange directory information with other systems supporting LDAP. This LDAP topic contains several good references for additional LDAP information.

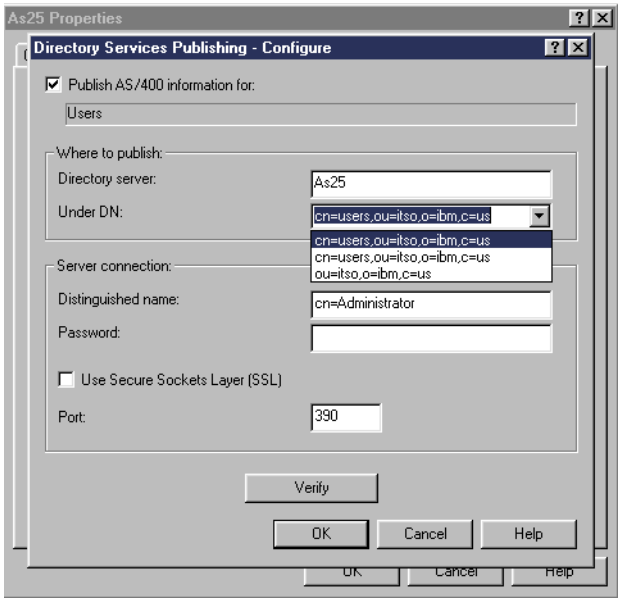


Figure 581. Directory services publishing: Configuration

1. In the Under DN field, enter the distinguished name (DN) under which you want AS/400 information added to the directory server. If this is the first time you are configuring the parent distinguished name for Users, you can click the **Verify** button to have the Directory Services Publishing function add the DN path to the LDAP directory. We use the Verify button later in this sequence of steps.
2. In the Distinguished name field, enter a DN to use when connecting to the directory server to publish AS/400 information. The cn=Administrator is the default. You can authorize other DNs to also publish.
3. In the Password field, enter the password for the DN you specified in the previous step.
4. If your directory server uses Secure Sockets Layer (SSL), select Secured using the SSL button. You must separately configure SSL on OS/400 through the Digital Certificate Manager interface described in Chapter 3, “SSL security on the AS/400 system” on page 41.
5. If your directory server does not use the default port, enter the correct port number in the Port field. In our example, we use port 390 because Domino for AS/400 is also active on As25.
6. Click the **Verify** button to verify that the specified DN exists on the server and the Server connection DN and password are valid. If the directory path does not exist, a dialog prompts you to create it.

   Note: If the parent DN does not exist, and you do not create it, publishing will not be successful.

In this example, we used the Verify button twice for “Users”. The first time we used Under DN ou=itso,o=ibm,c=us. The second time we used Under DN cn=users,ou=itso,o=ibm,c=us. In both cases, we let the verify function create the entry for us. Assuming the directory path exists or is created successfully, the message shown in Figure 582. appears.

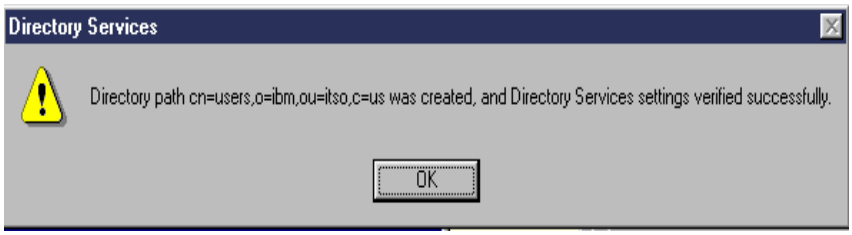


Figure 582. Directory services publishing: Successful directory path configuration

7. Click **OK**.

   You are finished specifying the LDAP Publishing functions.

12.11.2 LDAP and system distribution directory cross referencing

Table 22 shows the correspondence between OS/400 system distribution directory entry field content and the LDAP directory entry attribute.

Table 22. System distribution and LDAP directories information correspondence

System distribution directory field	LDAP attribute
User profile	uid
Description	description
Last name	sn (surname), cn (common name)
First name	givenName,cn (common name)
Preferred name	cn (common name)
Full name	cn (common name)
User id	cn (common name)
Department number	departmentNumber
Job title	title
Telephone number 1 and 2	telephoneNumber
FAX telephone number	facsimileTelephoneNumber
Office	roomNumber
Address lines 1-4	registered address
SMTP name	mail

The common name (cn) uses the following formats:

- 'First name' 'Middle Name' 'Last name'
- 'Preferred name' 'Last name'
- 'Full name'
- 'UserID'

Let us use Deborah Goodman as an example. The first name of “Deborah” has a, preferred name of “Deb”, middle name (initial) of “T”, last name of “Goodman”, and a user ID of “DEBGOOD”. DEBGOOD would have the common names:

- cn=Deborah T. Goodman
- cn=Deb Goodman
- cn=Goodman, Deborah T. (Deb)
- cn=DEBGOOD

The distinguished name (DN) is the first common name (cn) combined with the directory path. For example, if the directory path is 'ou=itso, o=ibm, c=us', the distinguished name (dn) for this user would be:

'cn=Deborah T. Goodman,ou=itso, o=ibm, c=us'.

Not all entries in the OS/400 system distribution directory are published to the LDAP directory. Some entries are prevented automatically from being published to LDAP. They are the *ANY system distribution directory entries used for generic routing of distributions and some other IBM-supplied entries starting with the letter Q (QSECOFR, QDOC, QSYS, QDFTOWN, QUSER for example).

A specific user can be explicitly prevented from being published to the LDAP directory through use of the user-defined field QREPL QLDAP for a system distribution directory entry.

For more details, refer to the Web site at: `http://www.as400.ibm.com/ldap`

12.12 Searching the directory server

There are several different LDAP clients for different platforms available. In this section, we show you the Netscape and the Microsoft Outlook LDAP clients.

12.12.1 Viewing LDAP entries

You can view all, or portions of, the OS/400 LDAP directory through the use of any of the following options or their equivalent:

- OS/400 QShell LDAP command `ldapsearch`: This LDAP command searches the LDAP directory for entries, based upon search criteria.
- The LDAP search facility from a browser configured to support LDAP.

Figure 583 shows an example of the Qshell `ldapsearch` command (1) and the first lines of the search results (starting at 2).

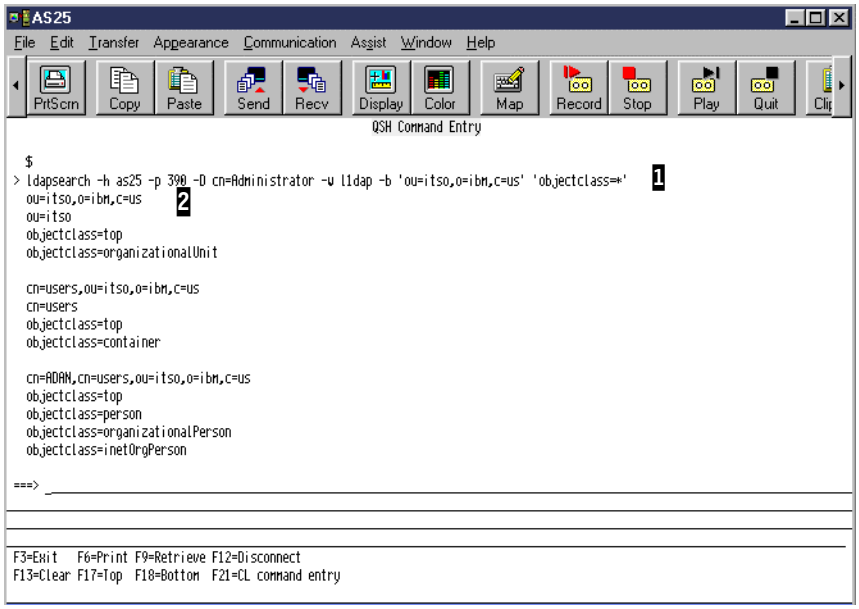


Figure 583. LDAP search from Qshell

You can enter the LDAP search function from most modern browsers. Figure 584 on page 474 and Figure 585 on page 475 are two example displays from an LDAP search request from a browser. In Figure 584, you see the same first three directory entries as shown in Figure 579 on page 467. Examine the Location URL **1** that represents the LDAP search request from the browser.

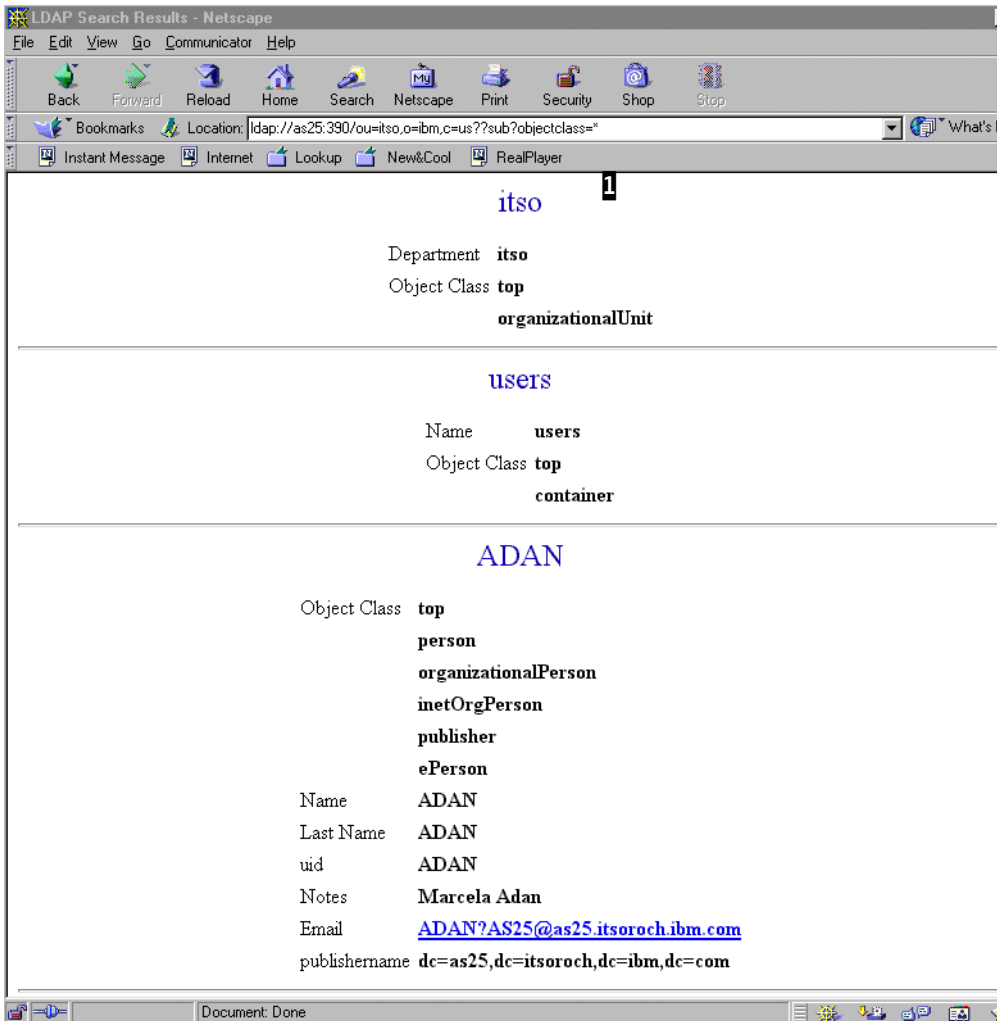


Figure 584. LDAP search from browser results: Initial screen

In Figure 585, we scrolled down the search results to the entry for James T CookAS25.

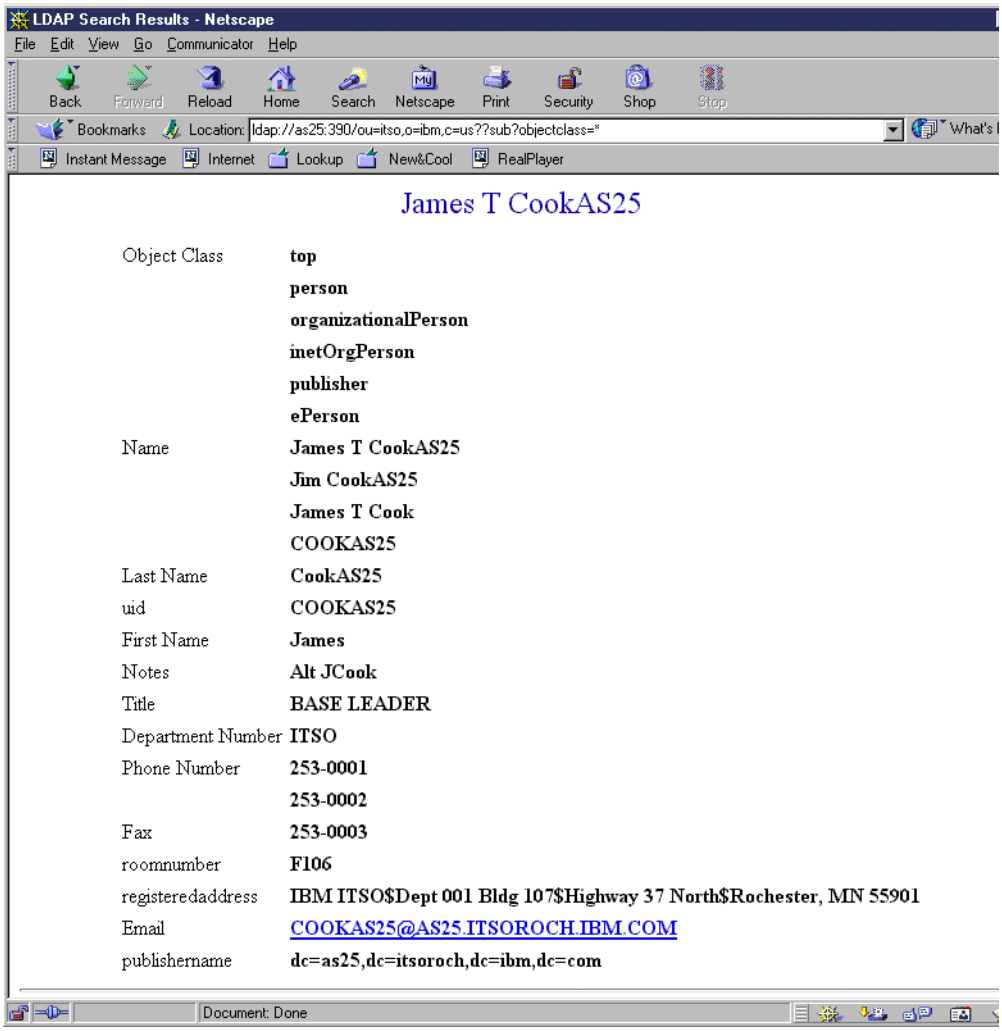


Figure 585. LDAP search from browser: OS/400 system distribution directory entry

For user CookAS25, you can see the LDAP entry attributes that correspond to OS/400 system distribution directory entry fields, such as user profile (COOKAS25), department number, job title, telephone number 1 (253-0001), telephone number 2 (253-0002, FAX telephone number (253-0003), office, address lines 1 through 4, and so on.

12.12.2 Netscape LDAP client

Netscape Communicator offers LDAP client support. In this example, we use Netscape Communicator Version 4.5. From the Communicator menu, select Address Book entry.

To add your Directory Server to Netscape's Address Book, select from the **File** menu **New Directory**. See Figure 586 on page 476 for an example. Enter a description for the server, a server name, the search root, and, if needed, correct the other settings.

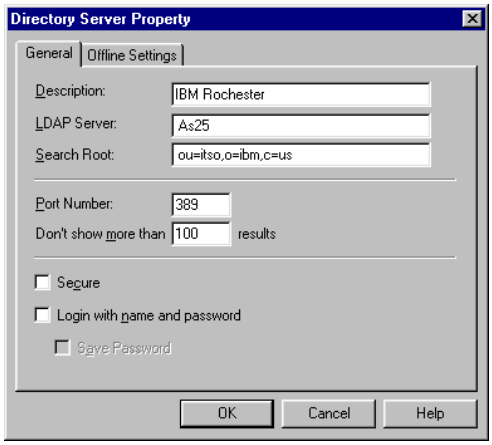


Figure 586. Netscape directory client configuration

Once successfully configured, you can search your directory server by entering a name or part of the name, or click on the **Search For..** button to specify other search criteria. An example for a name search is shown in Figure 587.

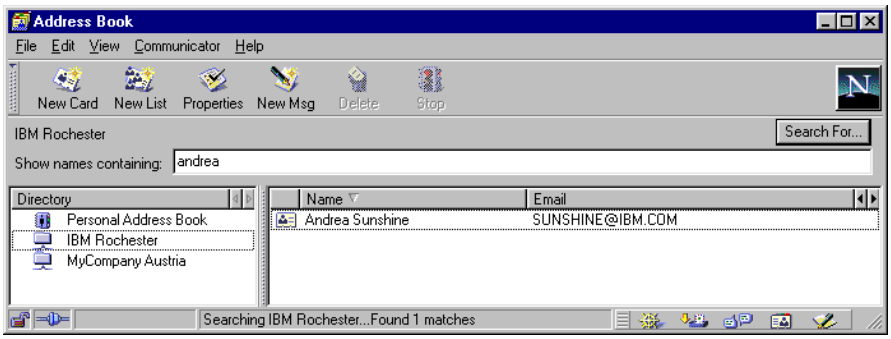


Figure 587. Netscape Address Book

12.12.3 Microsoft Outlook for the LDAP client

You can also configure Microsoft Outlook to use a LDAP Directory to look up directory information. Follow those steps to configure Outlook:

1. Start Microsoft Outlook.
2. From the **Tools** menu, select **Accounts**.
3. Select the **Directory Service** tab as shown in Figure 588.

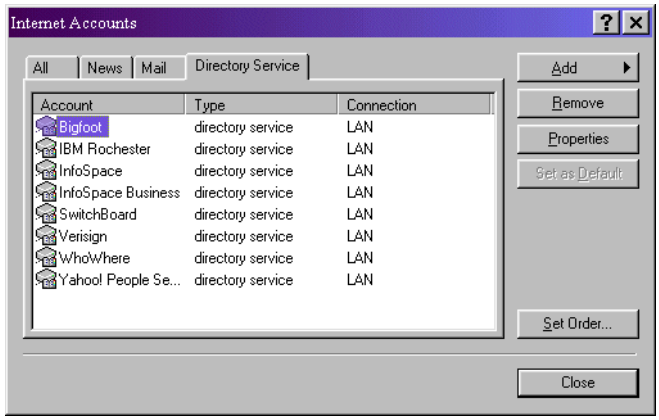


Figure 588. Outlook Directory Service tab

4. Select **Add**.
5. Select **Directory Service**.

   A wizard guides you through the next steps:

   a. Enter the name or IP address of your directory server.

   b. Select if you want Outlook to check e-mail addresses against this directory server.

   c. Enter a “Friendly Name” for this Directory Server.

   The Directory Server is now configured to be used by Microsoft Outlook.
6. To configure the advanced options, select the directory server, and then click on the **Properties** button.
7. Select the **Advanced** tab (Figure 589).

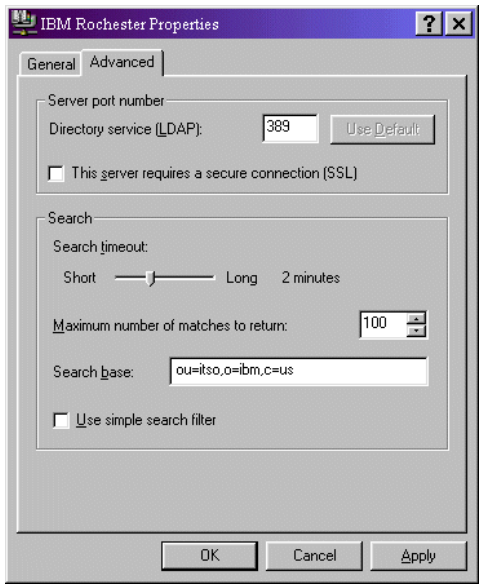


Figure 589. Outlook Directory Client Advanced Configuration tab

8. To search the directory server, select **Edit**.

9. Select **Find People**. Enter a part or the name of the person you are searching for, and press the **Find Now** button. A display similar to the example in Figure 590 appears.

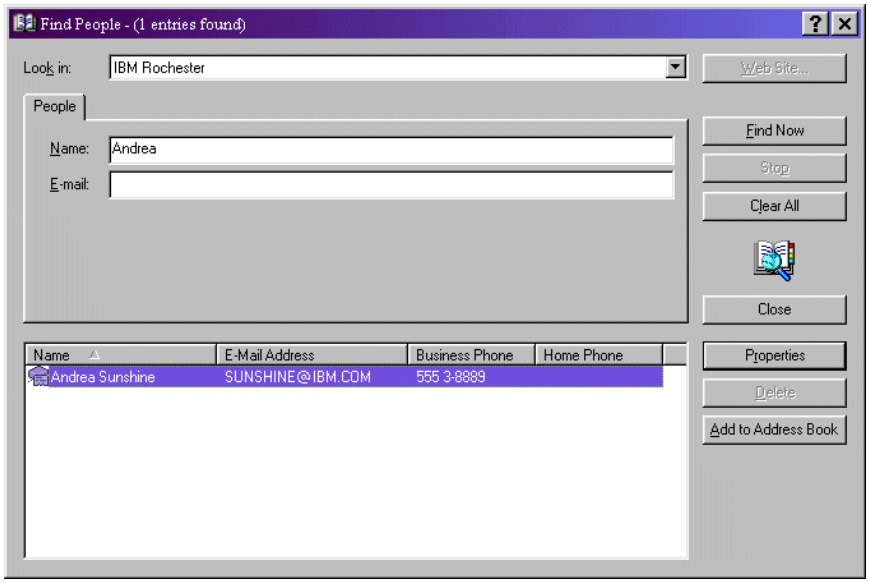


Figure 590. Outlook search example

12.13 Troubleshooting AS/400 Directory Services

Unfortunately, even reliable servers, such as the AS/400 Directory Services LDAP server, sometimes have problems. When your LDAP directory server has problems, the following information can help you figure out what is wrong and how to fix the problem.

12.13.1 Basic troubleshooting procedure for AS/400 Directory Services

On OS/400, the directory server jobs have the name QDIRSRV. The publishing server jobs have the name QGLDPUBA (publishing agent) and QGLDPUBE (publishing engine). The publishing and directory server jobs run in subsystem QSYSWRK.

When you get an error on your LDAP directory server and want more detail, view the QDIRSRV job log. AS/400 Directory Services uses several Structured Query Language (SQL) servers. When an SQL error occurs, the QDIRSRV job log usually contains the message: `SQL error -1 occurred`.

In these cases, the QDIRSRV job log refers you to the SQL server job logs. However, in some cases, QDIRSRV may not contain this message and this referral, even if an SQL server is the cause of the problem. In these cases, it will help you to know what SQL servers should be started, and for what AS/400 Directory Services uses them.

When the LDAP directory server starts normally, it generates the messages similar to those shown in Figure 591.

```
                    System:   AS20
Job . . :   QDIRSRV       User . . :   QDIRSRV       Number . . . :   174440

 >> CALL PGM(QDIRSRV/QGLDSVR)
    Job 128297/QUSER/QSQSRVR used for SQL server mode processing.
    Job 128298/QUSER/QSQSRVR used for SQL server mode processing.
    Job 128297/QUSER/QSQSRVR used for SQL server mode processing.
    Job 128295/QUSER/QSQSRVR used for SQL server mode processing.
    Job 128302/QUSER/QSQSRVR used for SQL server mode processing.
    Directory Services server started successfully.
```

Figure 591. QDIRSRV job log

AS/400 Directory Services uses the first SQL server, *128297/QUSER/QSQSRVR*, during LDAP server start-up. After start-up, this SQL server is dropped.

AS/400 Directory Services uses the second SQL server, *128298/QUSER/QSQSRVR*, used for LDAP adds, deletes, and modifies.

AS/400 Directory Services uses the third SQL server, *128297/QUSER/QSQSRVR*, only for replication. In the example above, the first SQL server has been reclaimed for replication.

AS/400 Directory Services uses the remaining SQL servers for LDAP searches. The number of SQL servers that it uses varies.

This value is based on the maximum connections value that you specify when you set up the performance values for AS/400 Directory Services in Operations Navigator, under the performance tag of the Directory Services server. In the example above, there are two SQL servers used for LDAP searches, 128295/QUSER/QSQSRVR and 128302/QUSER/QSQSRVR.

To display the directory server jobs using Operations Navigator, you select **TCP/IP Servers**. Scroll through the list of servers, and right-click on **Directory**. Select **Server Jobs**. To display publishing server jobs, right-click on the system name (As25, for example). Select **Properties** from the pull-down menu. On the Properties window, select the **Directory Services** tab. On the Directory Services dialog box, select the **Server Jobs** button.

You can use the job logs for those jobs to determine the possible problem cause.

12.13.2 Common LDAP client errors

Knowing the causes of common LDAP client errors can help you to solve problems with your server. For a complete list of LDAP client error conditions, see *System API Reference*, SC41-5801.

Common messages include:

- **ldap_search: Timelimit exceeded**:

  This error occurs when ldapsearches are performing slowly. To correct this error, you can do one or both of the following options:

- Increase the search time limit for the LDAP directory server. See 12.10.6, “Adjusting performance of the LDAP directory server” on page 467, for information on doing this.
- Reduce the activity on the AS/400 system. You can also reduce the number of active LDAP client jobs running.

- **[Failing LDAP operation]: Operations error**:

  Several things can generate this error. To get information about the cause of this error for a particular instance, view the QDIRSRV and Structured Query Language (SQL) server job logs as described in Basic troubleshooting procedure for AS/400 Directory Services.

- **ldap_bind: No such object**:

  This error is usually generated when the LDAP client attempts to bind with a DN that does not exist. For details about the error, view the QDIRSRV job log as described in Basic troubleshooting procedure for AS/400 Directory Services.

- **ldap_bind: Inappropriate authentication**:

  This error is usually generated when the client attempts to bind with a password that is not valid. To obtain detail about the error, view the QDIRSRV job log as described in Basic troubleshooting procedure for AS/400 Directory Services.

- **[Failing LDAP operation]: Insufficient access**:

  This error is usually generated when the binding DN does not have authority to do the operation (such as an add or delete) that the client requests. To get information about the error, view the QDIRSRV job log as described in Basic troubleshooting procedure for AS/400 Directory Services.

- **[AS/400 System]: A remote host refused an attempted connect operation**:

  The most common causes of this error include:

  - An LDAP client makes a request before the LDAP server on the specified AS/400 system is up and in select wait status.
  - The user specifies a port number that is not valid.

    To get information about the error, view the QDIRSRV job log as described in Basic troubleshooting procedures for AS/400 Directory Services. If the Directory Services server started successfully, the message `Directory Services server started successfully` will be in the QDIRSRV job log.

Chapter 13. Printing using TCP/IP

This chapter explains printing over a TCP/IP network. There are three ways to print using TCP/IP:

- Using LPR/LPD
- Using Telnet printer pass-through
- Using the OS/400 TCP/IP printer driver

13.1 Printing using LPR/LPD

Line Printer Requester (LPR) and Line Printer Daemon (LPD) allow the AS/400 system to send and receive spooled files across a TCP/IP network. Line Printer Requester is the client portion that allows spooled files to be sent to another system. Line Printer Daemon is the process on the receiving system that allows the file to be printed.

As well as sending the file itself, LPR and LPD allow the user to send certain attributes or options. These attributes are different than the attributes that the AS/400 maintains for spooled files. They allow information about the sending system and user and some simple information about how the file should be printed, such as the number of copies or the presence of a separator page.

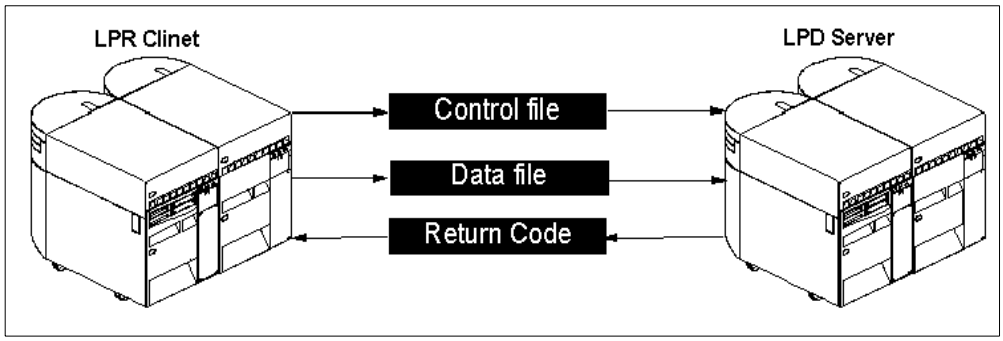


Figure 592. Data flow between an LPR client and an LPD server

13.1.1 LPR/LPD prerequisites

TCP/IP must be started on the AS/400 host before LPR can be used to send files or LPD can be used to receive files. The command to start TCP/IP on the AS/400 host is:

```
STRTCP
```

13.1.2 Configuring LPR on the AS/400 host

LPR is used to send a print file from the AS/400 host to an LPD Server. The LPD Server can be another AS/400 host or any other machine that provides an LPD Server. There are LPD Servers available for most popular operating systems. Some printers, such as the IBM Network Printer, contain an operating system independent of LPD Server.

Before you send the file using LPR, you need to know the spooled file number. This number is shown in the Work with Spooled Files (WRKSPLF) display. Enter

WRKSPLF, and then press F11 twice to go to View 3. You may also require the file, job, and user name for the spooled file.

```
                          Work with All Spooled Files

Type options, press Enter.
  1=Send   2=Change   3=Hold   4=Delete   5=Display   6=Release   7=Messages
  8=Attributes        9=Work with printing status


                File
Opt  File       Nbr   Job         User        Number  Queue       Library
     QPJOBLOG     1  QPADEV0003  USERID01    001314  QEZJOBLOG   QUSRSYS

                                                                       Bottom
Parameters for options 1, 2, 3 or command
===>
F3=Exit   F10=View 2   F11=View 4   F12=Cancel   F22=Printers   F24=More keys
```

Figure 593. WRKSPLF shows the spooled file number (View 3)

Note

The 1=Send option on the Work with Spooled Files display can only be used to send the file to an SNA device. The LPR or SNDTCPSPLF command is used to send the file to a TCP/IP device.

13.1.2.1 Using the LPR command to send the file to the LPD server

The LPR command (or the equivalent SNDTCPSPLF command) is used to send a file from the AS/400 system to the LPD Server. The LPD host can be another AS/400 or a non-AS/400 device.

Figure 594 and Figure 595 show the AS/400 LPR command. The screen shown in Figure 595 has been modified to save space. Table 23 describes the fields used in the command. Table 24 on page 484 documents some for the printers supported by the LPD command.

```
                    Send TCP/IP Spooled File (LPR)

Type choices, press Enter.

Remote system  . . . . . .



Printer queue  . . . . . .



Spooled file . . . . . .                Name
Job name . . . . . .          *         Name, *
  User . . . . . .                      Name
  Number . . . . . .                    000000-999999
Spooled file number  . . . . .  *ONLY   1-9999, *ONLY, *LAST
Destination type . . . . . .    *OTHER  *AS400, *PSF2, *OTHER
Transform SCS to ASCII . . . .  *YES    *YES, *NO

                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display        F24=More keys
```

Figure 594. Sending a spooled file to a remote host using LPR

```
                    Send TCP/IP Spooled File (LPR)

Type choices, press Enter.

Internet address . . . . . .



                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display        F24=More keys
Messages pending on other displays.
```

*Figure 595. LPR command when remote system is set to *INTNETADR (Screen 2)*

Table 23. LPR command parameters

Parameter	Description
Remote system	The name of the system to which the spooled file is sent. The host name can be: - A host name added by Selecting Work with TCP/IP Host Table Entries from the CFGTCP menu. - A host name that is resolved to a TCP/IP address by a Domain Name Server. - *INTNETADR to use the value of the INTNETADR parameter.

Parameter	Description
Printer queue	Specifies the name of the destination printer queue. For destination systems that are AS/400 systems, this is the name of an output queue to which the spooled file is sent. The printer queue can contain both the library name and the name of the printer queue (for example QGPL/OUTQ1). If no library is specified, the library list of the sending user is searched on the destination system. When sending a file to a non-AS/400 system, this name is system independent. For some systems, the name may be case sensitive. Enclose the name in apostrophes to ensure the case is preserved. For example use “OutQ1” to ensure that the case in maintained.
Spooled file	The name of the spooled file, as displayed by the Work with Spooled files command.
Job name	The name of the job that created the spooled file. The special value * is used to specify that the spooled file was created by the same job issuing the LPR command.
User	If the spooled file name is anything other than *, this field can be used to specify the user that created the spooled file. If the user name is not specified, all jobs currently running are searched for a match against the job name specified above.
Number	If the spooled file name is anything other than *, and the user name has been specified, this field can be used to specify the job number that created the spooled file.
Spooled file number	The spooled file number, as shown by the Work with Spooled Files command.
Destination type	The type of host that will receive the spooled file. This parameter should be set to *AS400 when sending the file to another AS/400 host.
Transform SCS to ASCII	Set this parameter to *YES to transform the spooled file to ASCII. When sending to another AS/400 system, no transformation is necessary.

Table 24. Example printer queue names for some common printers

Interface used	Queue name
HP JetDirect Card (internal)	“text” for unformatted output “raw” for formatted output
HP JetDirect Server (external) (3 port - 1 IP address)	“text1” or “raw1” for port 1 “text2” or “raw2” for port 2 “text3” or “raw3” for port 3
Integrated Network Option (IBM 4039, 3112, 3116, Lexmark OPTRA)	“prt0”
Lexmark MarkNet XLe	“/prt1” for parallel port 1 “/prt2” for parallel port 2 “/prt9” or “/ser” for serial port

Interface used	Queue name
IBM Network Print Server	“pr1” through to “pr8”
IBM Network Printer (4312, 4317, 4324)	PASS (or TEXT if PASS does not work)
IBM 3130	“afccu2”
Intel Netport XL	TEXT1 for parallel port 1 TEXT2 for parallel port 2
Intel Netport Pro	LPTx_PASSTHRU where x = port no. LPTx_TEXT where x = port no.
UNIX/RISC	printer queue name (case sensitive)
PC	printer queue name (often case sensitive)

Note

- The Send TCP/IP Spooled File (SNDTCPSPLF) command can be used in place of the LPR command. The parameters and capabilities of these two commands are identical.
- Using LPR to send spooled files to a V2R3 system or a V3R0M5 system requires these PTFs to be applied:
  - V2R3: PTF SF16482 or higher
  - V3R0M5: PTF SF16483 or higher

13.1.2.2 Sending a file from one AS/400 system to another

When sending a spooled file to another AS/400 system using TCP/IP, implementation-specific extensions to LPR are available to retain the spooled file attributes when the file is not transformed to ASCII. The control file is still sent, because it is part of the protocol, and the receiving AS/400 system checks for the control file extended print attribute. From the extended print attribute, the receiving AS/400 system can determine that the spooled file was sent from another AS/400 system.

In this case, all the spooled file attributes are sent in the data file. The option and attribute information in the control file is not used, and all the attributes of the spooled file are the same on both the sending and receiving systems, except for Job ID and output queue. The job user ID that ran the SNDTCPSPLF command is sent in the control file. The spooled file is created on the receiving system under this user profile, with a job name of QPRTJOB. If the user profile does not exist, the default user profile QTMPLPD is used. User IDs that do not exist on the receiving system can be prevented from sending with LPR by setting the *PUBLIC access of QTMPLPD to *EXCLUDE.

When the destination is an AS/400 system, the print queue value (PRTQ parameter) specified can be the name of any defined output queue on the destination AS/400 system. The full library name and output queue name should be specified when sending to another AS/400 system. If no library name is specified, the user IDs library list is searched. If the output queue is still not found, or the user ID is not authorized to it, the default output queue of QPRINT in library QGPL is used.

If the file is transformed to ASCII, the extended print attribute is not sent in the control file. The spooled file attributes are used by the transform program to produce an ASCII print data stream specific to the printer model parameter specified. The spooled file is created on the receiving AS/400 system as type *USERASCII with default attributes. Also, the spooled file name is changed to LPDxxxx, where x represents any valid hex character. This forms a unique signature identifying the LPR client that sent the file.

13.1.2.3 Sending a file from an AS/400 system to a non-AS/400 system

When using SNDTCPSPLF or LPR to send a spooled file to a non-AS/400 system, it may be necessary to refer to the LPD documentation for that implementation to determine the print queue value (PRTQ parameter). For example, the print queue value for the OS/2 licensed program is the physical name of the destination printer object on the desktop.

On some systems, the name of the destination printer queue can be case sensitive. If the name of the destination printer queue is lowercase or mixed case, enclose the name in apostrophes (‘ ’) to prevent the AS/400 system from making the name uppercase. This also applies to the destination-dependent options.

When sending to non-AS/400 systems, the LPD server reads the control file, which can contain a number of printing options and attributes, such as width of output and number of copies. Only attributes that are supported by the LPR/LPD protocol are sent. In particular, the page-range-to-print attribute is not supported. Implementation of the control file attributes is determined by the LPD server. This control file is part of the LPR/LPD protocol and is automatically built by LPR.

13.1.2.4 Authority for sending spooled files

To send a spooled file, you must have one of the following authorities to the file or to the output queue on which the file is located:

- Be the owner of the spooled file
- Have spool control authority (*SPLCTL)
- Have job control (*JOBCTL) special authority on an operator-controlled (OPRCTL(*YES)) output queue
- Be the owner of the output queue
- Have add, delete, and read authority to an output queue created with AUTCHK(*DTAAUT)
- Have read authority to output queue created with DSPDTA(*YES)

If you are using LPR through the remote writer, the remote writer changes its job attributes to run under the user profile that created the spooled file. This user ID must meet the authority requirements above, and must be enabled. The writer will not change its job attributes to those of a system profile. Instead, it uses the default user profile of QSPLJOB. This fails the authority test unless all users have read access to the file.

13.1.3 Using printer pass-through to send files to the LPD server

Printer pass-through enables you to route print files automatically to another system which supports TCP/IP LPD.

13.1.3.1 Creating an output queue for printer pass-through

To enable printer pass-through, you need to create an output queue with the Create Output Queue (CRTOUTQ) command and specify an RMTSYS parameter other than *NONE. You may specify either the remote system (host) name or *INTNETADR, in which case you will be prompted for the IP address of the remote system. Host names can be added by entering option 10 (Work with TCP/IP host table entries) from the CFGTCP menu. Figure 596 through Figure 598 on page 488 shows the parameters that we used on the CRTOUTQ command.

```
                      Create Output Queue (CRTOUTQ)

Type choices, press Enter.

Output queue . . . . . > OUTQ1         Name
  Library  . . . . . .     *CURLIB     Name, *CURLIB
Maximum spooled file size:
  Number of pages  . . . . . .   *NONE         Number, *NONE
  Starting time  . . . . . .                   Time
  Ending time  . . . . . .                     Time
              + for more values
Order of files on queue  . . . .   *FIFO         *FIFO, *JOBNBR
Remote system  . . . . . > *INTNETADR



Remote printer queue . . . . . > LPT1



                                                                      More...
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display        F24=More keys
```

Figure 596. Creating an output queue with the CRTOUTQ command (Screen 1)

```
                        Create Output Queue (CRTOUTQ)

Type choices, press Enter.

Writers to autostart . . . . . > 1            1-10, *NONE
Queue for writer messages  . . .  QSYSOPR     Name
  Library  . . . . . .  . . . .    *LIBL      Name, *LIBL, *CURLIB
Connection type  . . . . . > *IP              *SNA, *IP, *IPX, *USRDFN
Destination type . . . . . > *OTHER           *OS400, *OS400V2, *PSF2...
Host print transform . . . . .   *YES          *YES, *NO
Manufacturer type and model  . > *HP5SI
Workstation customizing object     *NONE      Name, *NONE
  Library  . . . . . .  . . . .                Name, *LIBL, *CURLIB
Image configuration  . . . . .   *NONE        *NONE, *IMGA01, *IMGA02...
Internet address . . . . . > '10.5.33.133'
Destination options  . . . . .   *NONE


Print separator page . . . . .   *YES         *YES, *NO

                                                                      More...
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display       F24=More keys
```

Figure 597. Creating an output queue with the CRTOUTQ command (Screen 2)

```
                        Create Output Queue (CRTOUTQ)

Type choices, press Enter.

User defined option  . . . . .   *NONE        Option, *NONE
            + for more values
User defined object:
  Object . . . . . .  . . . .   *NONE         Name, *NONE
    Library  . . . . . .  . .                 Name, *LIBL, *CURLIB
  Object type  . . . . . .  .                 *DTAARA, *DTAQ, *FILE...
User driver program  . . . . .   *NONE        Name, *NONE
  Library  . . . . . .  . . . .               Name, *LIBL, *CURLIB
Spooled file ASP . . . . . .   *SYSTEM       *SYSTEM, *OUTQASP
Text 'description' . . . . . > 'TCP/IP Output Queue'









                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display       F24=More keys
```

Figure 598. Creating an output queue with the CRTOUTQ command (Screen 3)

13.1.3.2 Starting printer pass-through

To start printer pass-through, perform the following tasks:

1. Enter the Start Remote Writer (STRRMTWTR) command.
2. Specify the queue name on the local (LPR) system that you created in the previous section. This starts a writer in the QSPL subsystem which, when using TCP/IP, uses LPR to send print files to the remote (LPD) system.

3. Ensure that LPD is operational on the remote system and, in the case of the AS/400 system, that the remote printer writer is active.

In the event of problems with this processing, the writer's job log is often a useful source of problem analysis material. To access this while the job is running, enter the `WRKACTJOB SBS(QSPL)` command and enter option `5` next to the writer job. This shows the Work with Writers display, from which it you must press F17 to see the job log. This sequence is different from the usual one for viewing batch and interactive job logs while the jobs are running.

13.1.4 Configuring and using LPD on the AS/400 host

This section shows how to set up the LPD function on the AS/400 system. It also shows you how to administer the LPD server using Operations Navigator and the AS/400 command line interface.

13.1.4.1 Changing the LPD server attributes

The LPD server can be started and configured using Operations Navigator. Complete these steps:

1. Start Operations Navigator by clicking **Start->Programs->IBM AS400 Client Access->AS/400 Operations Navigator**. The AS/400 Operations Navigator window appears (Figure 599).

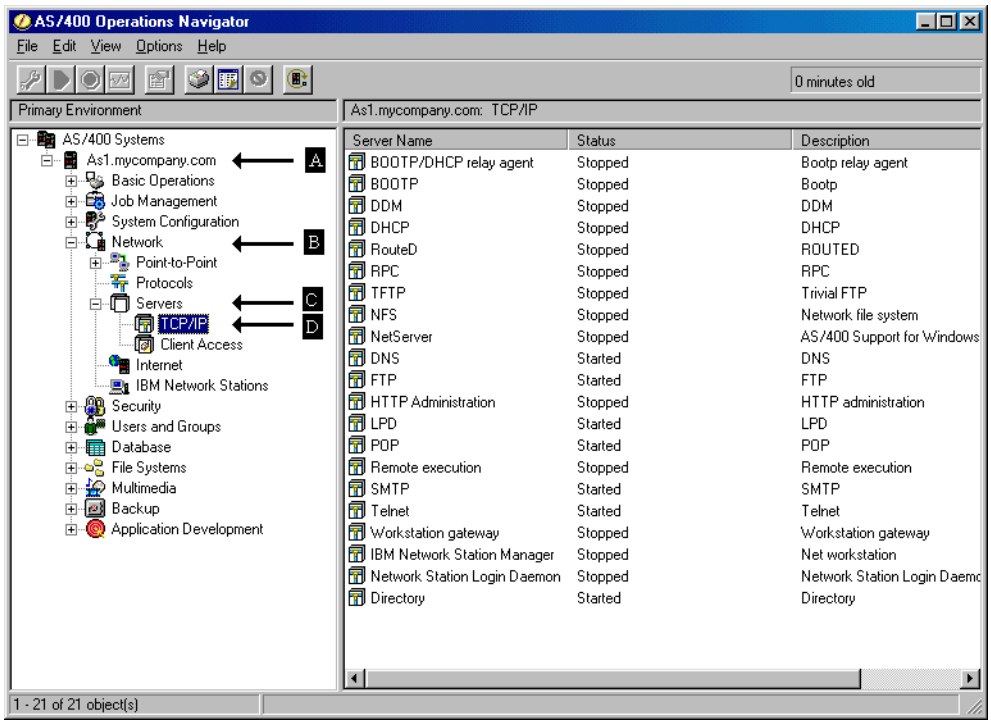


Figure 599. Operations Navigator: TCP/IP servers

2. Double-click the system icon (A) for the AS/400 system that you are configuring. The system components appear.
3. Double-click the **Network** icon (B). The network components appear.
4. Double-click the **Servers** icon (C). The available server types appear.

5. Double-click the **TCP/IP** icon (D). All of the TCP/IP servers are listed in the right window.
6. Right-click the LPD server in the right-hand window. The context menu for the LPD server appears (Figure 600).

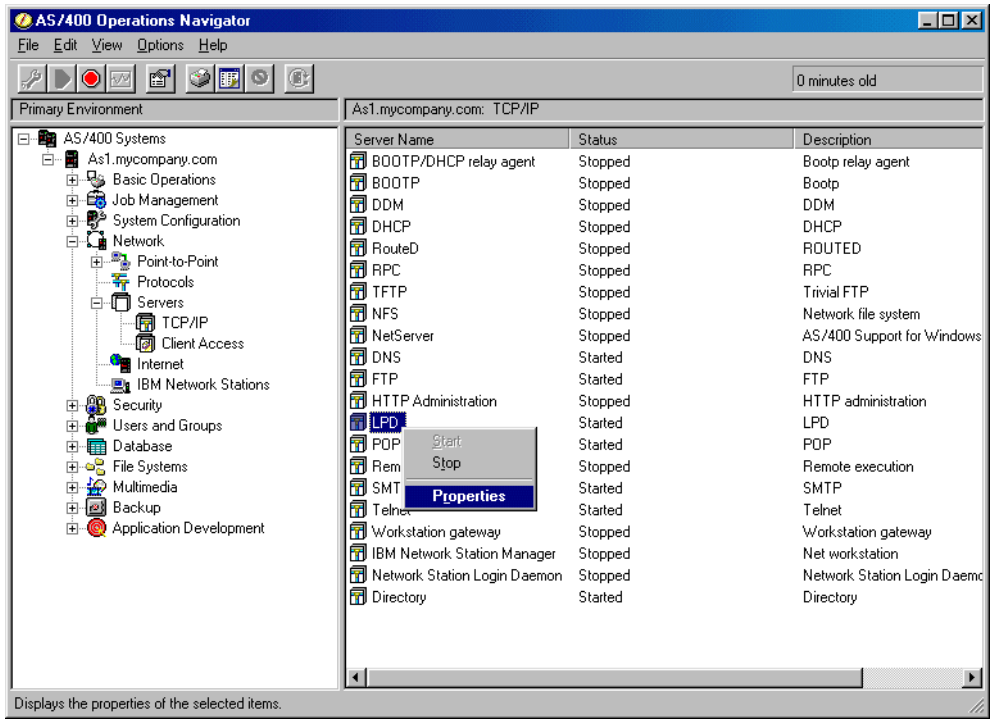


Figure 600. Properties option on the LPD Server Context menu

7. Select **Properties** from the context menu. The LPD Server properties window appears (Figure 601).

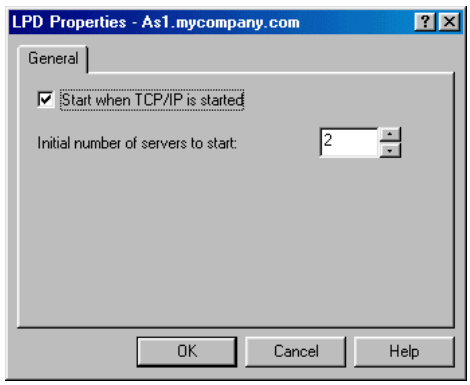


Figure 601. LPD Server Properties window

You can now select the number of listener jobs to be started and whether the LPD server should be started whenever TCP/IP is started on the AS/400 system.

You can also specify that one or more LPD Servers can be started automatically using the Change LPD Attributes (CHGLPDA) command. See Figure 602 for an example of the attributes that can be set using this command.

To start the LPD daemon automatically whenever TCP/IP is started, set the Autostart servers parameter to *YES. To ensure that the Line Printer Daemon must be started manually, set the Autostart servers parameter to *NO.

The Change LPD Attributes (CHGLPDA) command can also be used to specify how many LPD Servers are started when LPD is started. See Figure 602 for an example of the attributes that can be set using this command.

When LPD is started with the STRTCP command (see 13.1.4.2, “Starting the LPD server manually” on page 491), the full number of servers specified on this screen are started. When LPD is started with the STRTCPSVR command (see 13.1.4.1, “Changing the LPD server attributes” on page 489), only one server is started.

An LPD server is required for each file that is received at one time. One LPD server can be started, but it means that only one file can be received from a remote LPR client at a time. It is more often useful to start two or more LPD servers so that LPR clients are not forced to wait to send files.

```
                    Change LPD Attributes (CHGLPDA)

Type choices, press Enter.

Autostart servers  . . . . . .   *YES          *YES, *NO, *SAME
Number of initial servers  . . .   2            1-20, *SAME, *DFT

                                                                  Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 602. Changing the LPD server to start automatically using CHGLPDA

13.1.4.2 Starting the LPD server manually

If you set the Autostart server parameter to *YES when setting the LPD server attributes (see 13.1.4.1, “Changing the LPD server attributes” on page 489), the server is started when TCP/IP starts.

You can start the LPD server manually through Operations Navigator by following these steps:

1. Start Operations Navigator and open the TCP/IP servers display (see steps 1 through 5 in 13.1.4.1, “Changing the LPD server attributes” on page 489).
2. Right-click the LPD server in the right-hand window. The context menu for the LPD server appears (Figure 603 on page 492).

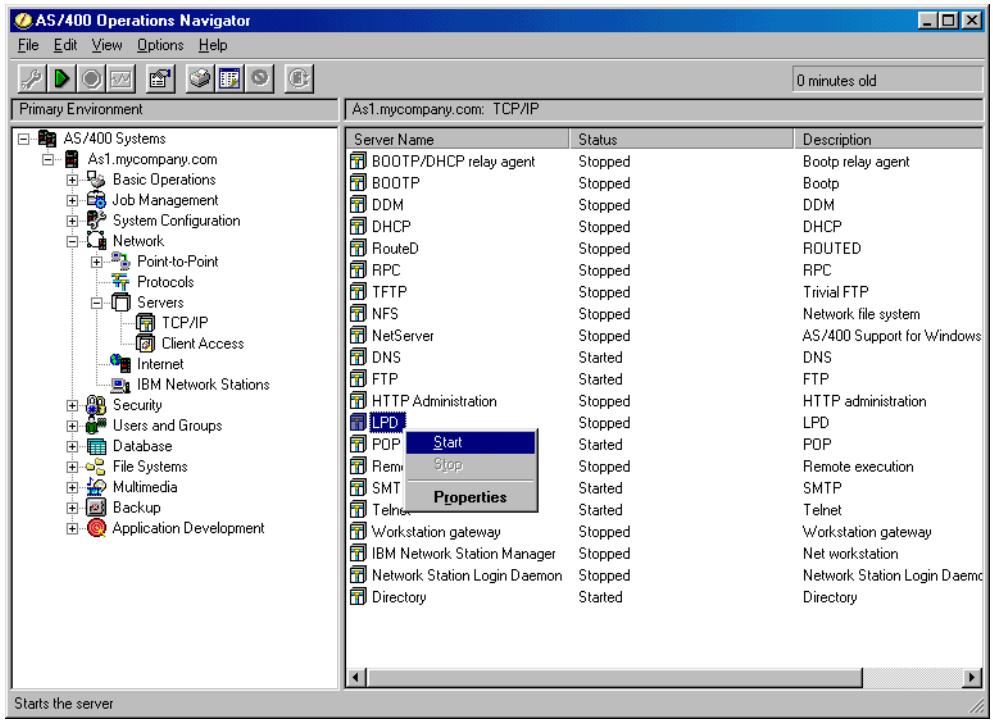


Figure 603. Start option on the LPD server context menu

3. Select **Start** from the context menu.

   You can also start the LPD Server manually by issuing the command:

   ```
   STRTCPSVR SERVER(*LPD)
   ```

13.1.4.3 Viewing the active LPD servers

You can use the Work with Active Jobs (WRKACTJOB) command to view the currently active LPD server jobs. The servers run in the QSYSWRK subsystem and have names in the format QTLPDnnnn.

```
                          Work with Active Jobs                    AS21
                                                                11/18/98  15:16:14
CPU %:    4.4     Elapsed time:   00:00:10     Active jobs:   128

Type options, press Enter.
  2=Change   3=Hold   4=End   5=Work with   6=Release   7=Display message
  8=Work with spooled files   13=Disconnect ...

Opt  Subsystem/Job  User        Type  CPU %  Function        Status
       QTFTP31808   QTCP        BCH      .0                  DEQW
       QTGTELNETS   QTCP        BCH      .2                  DEQA
       QTGTELNETS   QTCP        BCH      .0                  DEQW
       QTLPD00839   QTCP        BCH      .0                  DEQW
       QTLPD02503   QTCP        BCH      .0                  TIMW
       QTLPD02560   QTCP        BCH      .0                  DEQW
       QTMSNMP      QTCP        BCH      .0  PGM-QTOSMAIN     DEQW
       QTMSNMPRCV   QTCP        BCH      .0  PGM-QTOSRCVR     TIMW
       QTPOP00577   QTCP        BCH      .0                  DEQW
                                                                More...
Parameters or command
===>
F3=Exit   F5=Refresh       F7=Find      F10=Restart statistics
F11=Display elapsed data   F12=Cancel   F23=More options   F24=More keys
```

Figure 604. Using WRKACTJOB to view the active LPD servers

13.1.4.4 Stopping the LPD server manually

You can stop the LPD server manually through Operations Navigator by completing these tasks:

1. Start Operations Navigator and open the TCP/IP servers display (see steps 1 through 5 in 13.1.4.1, “Changing the LPD server attributes” on page 489).
2. Right-click the LPD server in the right-hand window. The context menu for the LPD server appears (Figure 605 on page 494).

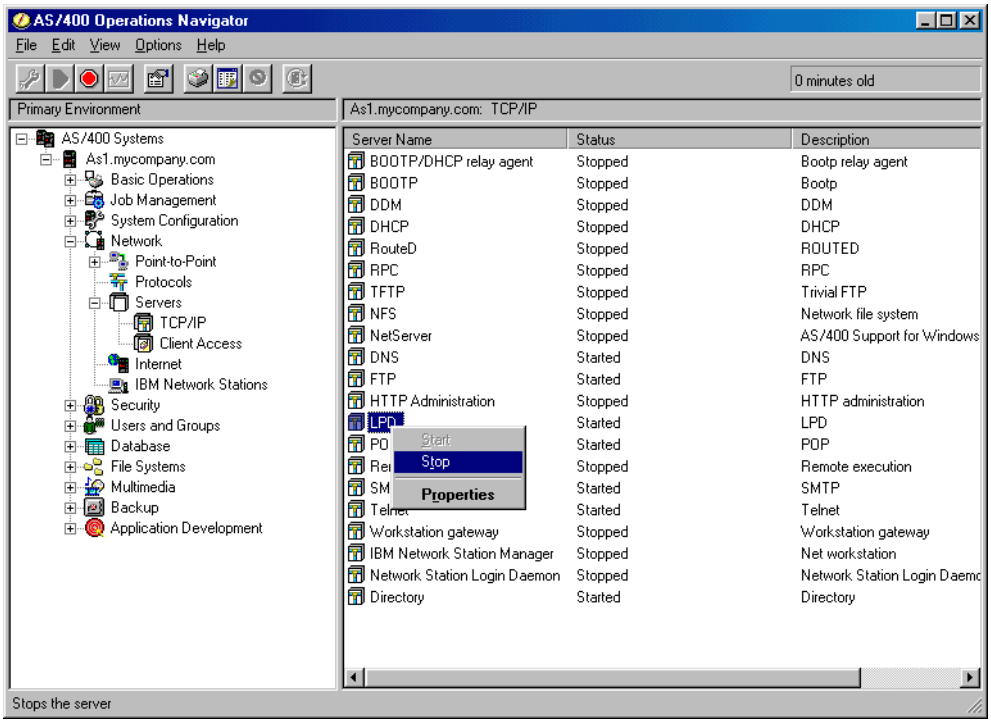


Figure 605. Stop option on the LPD server context menu

3. Select **Stop** from the context menu.

   You can also stop the LPD server jobs manually with the command:

   ```
   ENDTCPSVR SERVER(*LPD)
   ```

   This command ends all LPD Server jobs, including any currently receiving any LPR requests.

13.1.4.5 Changing the QTMPLPD printer file default values

If you want to change the default values in the QPTMPLPD printer file, use the Change Printer File (CHGPRTF) command. For example, to change the number of lines per page for all files received from non-AS/400 systems, type the following command statement:

```
CHGPRTF FILE(QUSRSYS/QPTMPLPD) PAGESIZE(60)
```

The change takes effect immediately and remains in effect until other changes are made, or until the printer file is created again. The following CHGPRTF parameters are overridden by LPD and cannot be changed through the use of the CHGPRTF command:

- FILE
- TOFILE
- PAGESIZE(*N n) (width option only is overridden)
- SPLFNAME
- OUTQ
- COPIES
- USRDTA

Note

The installation programs copy the QPTMPLPD printer file from the QTCP library into the QUSRSYS library. Two versions of this printer file exist: one in the QTCP library and one in the QUSRSYS library. To preserve the installation values, you should change the QUSRSYS version.

13.1.4.6 Authority for putting spooled files on the output queue

When LPD creates spooled files, it checks to ensure that the requester has the proper authority to place spooled files on the output queue. It checks to ensure that:

- The user has *READ authority to the output queue.
- The user has *SPLCTL special authority.
- The user has *JOBCTL special authority and the output queue is OPRCTL(*YES).

If any of these conditions are true, the user has authority to place the file on the requested output queue. If none of these conditions are true, the file goes to the default output queue QPRINT in library QGPL.

Note

There is no process to notify the requesting system that an authority error was detected, because the TCP connection is usually closed by the time this is determined. Any success messages posted by the LPR client application simply mean the file was temporarily received, but not necessarily kept. You should not delete any files until you have verified that you have proper authority and that your files were received successfully on the destination system.

13.1.4.7 Naming received spooled files on the AS/400 system

Prior to V3R1M0, if AS/400 LPR was used with the DESTTYP(*OTHER) and TRANSFORM(*YES) parameters, the name of the spooled file that was created became that of the PRTF file, QPTMPLPD. The original spooled file name was placed into the user data field of the created file. Likewise, for non-AS/400 LPR clients, the newly created file name was also named QPTMPLPD, and the original file name was placed into the user data field.

Beginning with V3R1M0, spooled files that are received by LPD have file names in the form LPDxxxx. The x represents any valid hex character. These hex characters are the result of cyclic redundancy checking that is performed on client information to identify the LPR client for LPRM support. Spooled files that are named in this manner are unique to each client. The name is used as a signature. All files from a single client have the same signature. A client must generate a matching signature to use LPRM to delete any LPR spooled file.

Clients can use LPQ (line printer queue) and LPRM (line printer removal) commands to query and remove LPR spooled files, since AS/400 LPD supports both functions. However, these commands cannot be issued by the AS/400 system as a client.

If the LPR client is another AS/400 system using the DESTTYP(*AS400) and TRANSFORM(*NO) parameters, the spooled file has exactly the same attributes on the receiving queue that it had on the sending queue. The spooled file name is not converted to LPDxxxx form and is left unchanged.

The LPQ command sent to an AS/400 system requires a joblist parameter, or, more specifically, a user profile under which the query is to be performed. For example, on OS/2, this would be:

```
lpq -pmylib/myoutq -sas400.endicott.ibm.com ProfileName
```

13.1.4.8 Ownership of received files on the destination AS/400 system

If the user ID on the sending system exists on a destination AS/400 system, the spooled file is created under that user profile. However, if the user profile does not exist on the destination system, the spooled file is created under the QTMPLPD user profile.

When sending from a non-AS/400 system to an AS/400 system, the file is always created using the QPTMPLPD printer file. This is also true when sending from an AS/400 system to an AS/400 system with the TRANSFORM(*YES) parameter. In these cases, spooled files received by the LPD server are placed under special jobs.

For example, if user ID JOHN exists, the file is placed under job 999999/JOHN/QPRTJOB. If user ID JOHN does not exist, the file is placed under 999999/QTMPLPD/QPRTJOB.

13.2 Printing using Telnet Printer Pass-Through

Printing from an AS/400 system over TCP/IP can also be achieved by using Telnet Printer Pass-Through. Telnet Printer Pass-Through is available with the enhanced Telnet server in OS/400 V3R2 and above. The Telnet Printer Pass-Through mode allows the AS/400 user with a Telnet client that supports printer emulation to attach printer devices on the AS/400 system over the network and to work over native TCP/IP.

Telnet Printer Pass-Through is explained in 5.1.14, “Printer emulation support” on page 219.

13.3 Printing using the TCP/IP print driver

Printing from an AS/400 system over TCP/IP can also be achieved using the AS/400 PCL/PJL TCP/IP printer driver. This driver enables the AS/400 system to send spooled files directly to a TCP/IP attached printer that accepts the Hewlett Packard (HP) PCL/PJL printer languages.

No TCP/IP printer protocol has emerged as an industry standard, except for what Hewlett Packard provides on their printers. By their dominance, and the fact that other vendors clone their support (in HP emulation mode), this has become a “standard” protocol. This support uses printer control language (PCL) for the data stream and uses printer job language (PJL) for device status.

A print driver has been developed to drive HP-compatible network printers. This driver sends PCL/PJL commands to a dedicated TCP/IP port at a specified IP address.

This print driver works similar to and configures similar to the support for Lexmark Lexlink printers that drives network printers attached directly to the TCP/IP network. This driver opens up a socket connection to the printer on a TCP/IP port (9100 in many cases) and sends HP PCL/PJL commands to the printer. The PCL is the actually print data stream, and the driver uses PJL commands to get the printer status.

In other words, any printer that accepts PCL/PJL on a raw TCP/IP port would most likely work with this driver. This driver uses Host Print Transform to convert SCS or AFP to PCL. The driver also supports page ranges, forms control, and everything else the LexLink driver supports.

The CRTDEVPRT CL command is used to configure the printer driver. Figure 606 shows a sample of the CRTDEVPRT command.

```
                  Create Device Desc (Printer) (CRTDEVPRT)

Type choices, press Enter.

Device description . . . . . .               Name
Device class . . . . . .                     *LCL, *RMT, *VRT, *SNPT, *LAN
Device type  . . . . . .                     3287, 3812, 4019, 4201...
Device model . . . . . .                     0, 1, 2, 3, 4, 10, 13, 301...

                                                                        Bottom
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display       F24=More keys
```

Figure 606. Create Device Desc (Printer) (CRTDEVPRT) display

Complete the parameters on this page as shown in Table 25.

Table 25. Create Device Desc (Printer) parameters

Parameter	Value
Device Description	The name of your printer
Device class	*LAN
Device Type	3182
Device model	1

After you enter these values, more parameters appear as shown in Figure 607.

```
                  Create Device Desc (Printer) (CRTDEVPRT)

Type choices, press Enter.

Device description . . . . . > TCPPRINTER    Name
Device class . . . . . > *LAN          *LCL, *RMT, *VRT, *SNPT, *LAN
Device type  . . . . . > 3812          3287, 3812, 4019, 4201...
Device model . . . . . > 1             0, 1, 2, 3, 4, 10, 13, 301...
LAN attachment . . . . . > *IP         *LEXLINK, *IP, *USRDFN
Port number  . . . . . > 2501          0-65535
Online at IPL  . . . . . *YES          *YES, *NO
Font:
  Identifier . . . . . > 3             3, 5, 11, 12, 13, 18, 19...
  Point size . . . . . *NONE           000.1-999.9, *NONE
Form feed  . . . . . > *AUTOCUT        *TYPE, *CONT, *CUT, *AUTOCUT
Separator drawer . . . . . *FILE       1-255, *FILE
Separator program  . . . . . *NONE     Name, *NONE
  Library  . . . . . .                 Name, *LIBL, *CURLIB
Printer error message  . . . . . *INQ  *INQ, *INFO

                                                                    More...
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display        F24=More keys
```

Figure 607. Create Device Desc (Printer) (CRTDEVPRT) Extended display (Screen 1)

Complete the parameters on this page according to those shown in Table 26.

Table 26. Create Device Desc (Printer) parameters

Parameter	Value
LAN attachment	*IP
Port number	2501 IBM Network Printer 9100 HP, Lexmark and other IBM printers
Form feed	*AUTOCUT

Page down to continue to the display shown in Figure 608.

```
                Create Device Desc (Printer) (CRTDEVPRT)

Type choices, press Enter.

Message queue  . . . . . . . . . > QSYSOPR       Name, QSYSOPR
  Library  . . . . . . . . . . . >   *LIBL       Name, *LIBL, *CURLIB
Activation timer . . . . . . . .   170           1-2550, *NOMAX
Inactivity timer . . . . . . . .   *ATTACH       1-30, *ATTACH, *NOMAX...
Host print transform . . . . . .   *YES          *NO, *YES
Manufacturer type and model  . . > *IBM4312
Paper source 1 . . . . . . . . .   *MFRTYPMDL    *MFRTYPMDL, *LETTER...
Paper source 2 . . . . . . . . .   *MFRTYPMDL    *MFRTYPMDL, *LETTER...
Envelope source  . . . . . . . .   *MFRTYPMDL    *MFRTYPMDL, *MONARCH...
ASCII code page 899 support  . .   *NO           *NO, *YES
Image configuration  . . . . . .   *NONE         *NONE, *IMGA01, *IMGA02...
Character identifier:
  Graphic character set  . . . .   *SYSVAL       1-32767, *SYSVAL
  Code page  . . . . . . . . . .                 1-32767


                                                                      More...
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display        F24=More keys
```

Figure 608. Create Device Desc (Printer) (CRTDEVPRT) Extended display (Screen 2)

Complete the parameters on this page according to those shown in Table 27.

Table 27. Create Device Desc (Printer) parameters

Parameter	Value
Activation timer	170 If the printer is shared between multiple system then set the activation timer to *NOMAX. This will cause the AS/400 to wait to establish a connection if the printer is busy with another system. Set the value to 170 if the printer will only service one host.
Host print transform	*YES
Manufacturer type and model	*IBM4312

Page down to continue to the display shown in Figure 609 on page 500.

```
                Create Device Desc (Printer) (CRTDEVPRT)

Type choices, press Enter.

Remote location:
  Name or address  . . . . . > '10.5.62.45'



User-defined options . . . . . *NONE         Character value, *NONE
              + for more values
User-defined object:
  Object . . . . . . . . . . . *NONE         Name, *NONE
    Library  . . . . . .                     Name, *LIBL, *CURLIB
  Object type  . . . . . .                   *DTAARA, *DTAQ, *FILE...
Data transform program . . . . . *NONE       Name, *NONE
  Library  . . . . . .                       Name, *LIBL, *CURLIB
System driver program  . . . . . > *IBMPJLDRV
Text 'description' . . . . . > 'TCP/IP attached IBM Network Printer 12'

                                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display        F24=More keys
```

Figure 609. Create Device Desc (Printer) (CRTDEVPRT) Extended display (Screen 3)

Complete the parameters on this page according to those shown in Table 28.

Table 28. Create Device Desc (Printer) parameters

Parameter	Value
Remote location	10.5.62.45 Enter the IP address of the printer
System driver program	*IBMPJLDRV Use *HPPJLDRV for a non-IBM HP-compatible PJL/PCL printer. Use *NETSTNDRV for a Network Station driver.
Text ‘description’	Enter a description for the printer device

Note

The TCP/IP print driver under V3R7M0 requires PTF SF43497, SF44339, and SF45336. Under V4, no PTF is required. The PCL/PJL driver is not supported prior to V3R7.

13.4 Considerations when printing over TCP/IP

Printing over TCP/IP can introduce some issues that are not present when printing over an SNA network.

13.4.1 Printing page ranges using LPR/LPD

The LPR command does not allow you to send page ranges of a spooled file to a printer. It will always send the entire file. If you want to print page ranges using

LPR/LPD, you need to use a third-party utility to extract the pages ranges and then send the extracted portion.

13.4.2 LPRM and LPQ clients

The AS/400 LPD Server supports line printer queue (LPQ) and line printer remove (LPRM) clients. Clients can use LPQ and LPRM commands to query and remove LPR spooled files, as AS/400 LPD supports both functions. However, these commands cannot be issued by the AS/400 system as a client.

If the LPR client is another AS/400 system using the DESTTYP(*AS400) and TRANSFORM(*NO) parameters, the spooled file has exactly the same attributes on the receiving queue that it had on the sending queue. The spooled file name is not converted to LPDxxxx form and is left unchanged.

The LPQ command sent to an AS/400 system requires a joblist parameter or, more specifically, a user profile under which the query is to be performed. For example, on OS/2 this would be:

```
lpq -pmylib/myoutq -sas400.endicott.ibm.com ProfileName
```

13.4.3 Address mapping

When using TCP/IP, each connected display device may have a dynamically allocated IP address and device name. This can cause difficulties when associating printer devices to display devices. OS/400 V3R2 and above contain enhancements to assist with the management of display and printer mapping.

Section 13.5.5, “Using an initial program to map printer IP addresses” on page 514, contains a sample program that uses these enhancements to manage IP address mapping.

13.4.4 Security

Spooled files that are printed using any of the supported TCP/IP methods (LPR/LPD, Telnet Printer Pass-Through, and the TCP/IP printer driver) will be sent to the destination printer in an unencrypted form. This means that you will need to be careful if you are going to use an untrusted TCP/IP network (such as the Internet) to print sensitive information.

It is possible to secure the printer traffic by using a Virtual Private Network (VPN) that will ensure that the data sent is properly protected. VPNs can be implemented using the Firewall for AS/400 or other products, such as Point-to-Point Tunneling Protocol (PPTP).

For more information on securing your AS/400 when connected to the Internet, see *AS/400 Internet Security: IBM Firewall for AS/400,* SG24-2162.

13.5 TCP/IP printing scenarios

This section describe, several common TCP/IP printing scenarios. It contains step-by-step instructions on how to configure these scenarios.

If you experience errors when trying to print over, consult 19.2, “TCP/IP printing problem determination” on page 659. You can display extra information about errors by positioning the cursor on the error message and pressing F1.

13.5.1 Using LPR/LPD to send files from one AS/400 system to another

This section describes how to configure two AS/400 system so that LPR/LPD can be used to send spooled files from an output queue on one AS/400 system to an output queue on the other AS/400 system. The files are sent in such a way that the original attributes of the file are preserved on the receiving AS/400 system.

The LPD Server must be started on the remote AS/400 system before you can able to send the files with the LPR or SNDTCPSPLF commands. See 13.1.4, “Configuring and using LPD on the AS/400 host” on page 489, for instructions on starting and configuring the LPD Server.

Before sending the spooled file, you need to collect spooled file details. Use the Work with spooled files (`WRKSPLF`) command to collect the following information about the spooled file:

- Name
- Job name
- User
- Number

The next step is to set up a host table entry on the sending AS/400 system so that it knows about the receiving AS/400 system. This step is only necessary if you want to avoid using the actual TCP/IP address of the receiving AS/400 system (10.5.69.233) and would prefer to use a more meaningful text name (AS22).

Issue the command `GO CFGTCP` and enter option `10` (Work with TCP/IP host table entries) to add the entry. If the system you want to add is not already there, add a new entry using option `1`.

```
                    Work with TCP/IP Host Table Entries
                                                              System:   AS21
Type options, press Enter.
  1=Add   2=Change   4=Remove   5=Display   7=Rename

     Internet         Host
Opt  Address          Name

     10.5.62.163      PCPL
     10.5.69.230      AS20
     10.5.69.232      AS21
     10.5.69.233      AS22
     127.0.0.1        LOOPBACK
                      LOCALHOST









                                                                       Bottom
F3=Exit   F5=Refresh   F6=Print list   F12=Cancel   F17=Position to
```

Figure 610. Work with TCP/IP Host Table Entries screen

Once the host table entry exists, you can send the spooled file using the LPR command. Type `LPR`, and then press F4. Fill out the parameters as shown in Table 29.

Table 29. LPR parameters

Parameter	Value
Remote system	The name that you entered in the TCP/IP host table entry
Printer queue	qgpl/qprint This specifies the print queue that will receive the spooled file on the destination AS/400 system
Spooled file	The spooled file name that you retrieved using the WRKSPLF command
Job name	The job name that you retrieved using the WRKSPLF command
User	The user name that you retrieved using the WRKSPLF command
Number	The job number that you retrieved using the WRKSPLF command
Spooled file number	*ONLY
Destination type	*AS400
Transform SCS to ASCII	*NO

Figure 611 shows an example of using the LPR command to send a printout to System AS22.

```
                      Send TCP/IP Spooled File (LPR)

Type choices, press Enter.

Remote system  . . . . . . . . . > AS22



Printer queue  . . . . . . . . . > 'qgpl/qprint'



Spooled file . . . . . . . . . . > QSYSPRT       Name
Job name . . . . . . . . . . . . > QPADEV0003    Name, *
  User . . . . . . . . . . . . . >   ITSCID57    Name
  Number . . . . . . . . . . . . >   002485      000000-999999
Spooled file number  . . . . . .   *ONLY         1-9999, *ONLY, *LAST
Destination type . . . . . . . . > *AS400        *AS400, *PSF2, *OTHER
Transform SCS to ASCII . . . . . > *NO           *YES, *NO

                                                                      More...
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display       F24=More keys
```

Figure 611. Using LPR or SNDTCPSPLF to send a spooled file to another AS/400

13.5.2 Using LPR/LPD to send files from an AS/400 system to a PC

This section describes how to configure two AS/400 system so that LPR/LPD can be used to send spooled files from an output queue on one AS/400 system to an output queue on a PC running an LPD server.

The LPD server must be started on the PC system before you can send the files with the LPR or SNDTCPSPLF commands.

Before sending the spooled file, you need to collect spooled file details. Use the Work with Spooled Files (`WRKSPLF`) command to collect the following information about the spooled file:

- Name
- Job name
- User
- Number

The next step is to set up a host table entry on the sending AS/400 system so that it knows about the receiving PC. This step is only necessary if you want to avoid using the actual TCP/IP address of the receiving AS/400 system (10.5.69.233) and would prefer to use a more meaningful text name (PCHOST).

Issue the command `GO CFGTCP` and enter option `10` (Work with TCP/IP host table entries) to add the entry. If the system you want to add is not already there, add a new entry using option `1`.

```
                    Work with TCP/IP Host Table Entries
                                                              System:   AS22
Type options, press Enter.
  1=Add   2=Change   4=Remove   5=Display   7=Rename

     Internet         Host
Opt  Address          Name

     10.5.62.57       PCHOST
     10.5.69.230      AS20
     10.5.69.232      AS21
     10.5.69.233      AS22
     10.1.2.2         NTPL
     127.0.0.1        LOOPBACK
                      LOCALHOST









                                                                      Bottom
F3=Exit   F5=Refresh   F6=Print list   F12=Cancel   F17=Position to
Internet address entry 10.5.62.57 added to host table.
```

Figure 612. Using Work with TCP/IP Host Table Entries to add a PC LPD server entry

Once the host table entry has been added, you can issue the `LPR` command to send the spooled file to the PC LPD server.

You then enter the details of the remote system and printer queue and the details of the spooled file you wish to print. Be sure to specify the printer queue name

exactly as the PC has it defined. With some PC LPD servers, the printer queue name is case sensitive. The LPR entry screen is shown in Figure 613.

After filling out the remote system and job information, press F10 to display additional parameters and then press the Page Down key to display the second screen (Figure 614).

```
                      Send TCP/IP Spooled File (LPR)

Type choices, press Enter.

Remote system  . . . . . > PCHOST



Printer queue  . . . . . > 'pctest'



Spooled file . . . . . > QSYSPRT       Name
Job name . . . . . . . . . . . > QPADEV0002    Name, *
  User . . . . . . . . . . . . >   ITSCID57    Name
  Number . . . . . . . . . . . >   001544      000000-999999
Spooled file number  . . . . . *ONLY         1-9999, *ONLY, *LAST
Destination type . . . . . > *OTHER        *AS400, *PSF2, *OTHER
Transform SCS to ASCII . . . . . > *YES          *YES, *NO
Manufacturer type and model  . . > *IBM4312
                                                                    More...
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display       F24=More keys
```

Figure 613. Using LPR or SNDTCPSPLF to send a spooled file to a PC (Screen 1)

```
                      Send TCP/IP Spooled File (LPR)

Type choices, press Enter.


                           Additional Parameters

Workstation customizing object     *NONE         Name, *NONE
  Library  . . . . . .                           Name, *LIBL, *CURLIB
Delete file after sending  . . .   *NO          *NO, *YES
Destination-dependent options  .


Print separator page . . . . . .   *YES          *NO, *YES









                                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 614. Using LPR or SNDTCPSPLF to send a spooled file to a PC (Screen 2)

13.5.3 Printing directly to a TCP/IP-attached IBM Network Printer 12

This example prints directly to a network attached IBM Network Printer 12. The printer used in the sample scenario has a Token-Ring card and an IP address of 10.5.62.45.

13.5.3.1 Creating the printer device

The first step when configuring this printer is to create the device description with the Create Device Desc (Printer) (CRTDEVPRT) command. The appropriate parameter values are shown in Figure 615 through Figure 618 on page 508.

```
                Create Device Desc (Printer) (CRTDEVPRT)

Type choices, press Enter.

Device description . . . . . > TCPPRINTER    Name
Device class . . . . . > *LAN          *LCL, *RMT, *VRT, *SNPT, *LAN
Device type  . . . . . > 3812          3287, 3812, 4019, 4201...
Device model . . . . . > 1             0, 1, 2, 3, 4, 10, 13, 301...
LAN attachment . . . . . > *IP         *LEXLINK, *IP, *USRDFN
Port number  . . . . . > 2501          0-65535
Online at IPL  . . . . . *YES          *YES, *NO
Font:
  Identifier . . . . . > 3             3, 5, 11, 12, 13, 18, 19...
  Point size . . . . . *NONE           000.1-999.9, *NONE
Form feed  . . . . . *TYPE           *TYPE, *CONT, *CUT, *AUTOCUT
Separator drawer . . . . . *FILE     1-255, *FILE
Separator program  . . . . . *NONE   Name, *NONE
  Library  . . . . . .               Name, *LIBL, *CURLIB
Printer error message  . . . . . *INQ  *INQ, *INFO

                                                                    More...
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 615. Create Device Desc (Printer) (CRTDEVPRT) command (Screen 1)

```
                  Create Device Desc (Printer) (CRTDEVPRT)

Type choices, press Enter.

Message queue  . . . . . .   QSYSOPR       Name, QSYSOPR
  Library  . . . . . .         *LIBL       Name, *LIBL, *CURLIB
Activation timer . . . . . .   170          1-2550, *NOMAX
Inactivity timer . . . . . .   *ATTACH      1-30, *ATTACH, *NOMAX...
Host print transform . . . . .   *YES        *NO, *YES
Manufacturer type and model  . > IBM4312
Paper source 1 . . . . . .   *MFRTYPMDL    *MFRTYPMDL, *LETTER...
Paper source 2 . . . . . .   *MFRTYPMDL    *MFRTYPMDL, *LETTER...
Envelope source  . . . . . .   *MFRTYPMDL  *MFRTYPMDL, *MONARCH...
ASCII code page 899 support  . .   *NO      *NO, *YES
Image configuration  . . . . .   *NONE       *NONE, *IMGA01, *IMGA02...
Character identifier:
  Graphic character set  . . . .   *SYSVAL   1-32767, *SYSVAL
  Code page  . . . . . .                     1-32767


                                                                    More...
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 616. Create Device Desc (Printer) (CRTDEVPRT) command (Screen 2)

```
                  Create Device Desc (Printer) (CRTDEVPRT)

Type choices, press Enter.

Remote location:
  Name or address  . . . . . > '10.5.62.45'



User-defined options . . . . . .   *NONE       Character value, *NONE
              + for more values
User-defined object:
  Object . . . . . .   *NONE       Name, *NONE
    Library  . . . . . .           Name, *LIBL, *CURLIB
  Object type  . . . . . .         *DTAARA, *DTAQ, *FILE...
Data transform program . . . . .   *NONE       Name, *NONE
  Library  . . . . . .           Name, *LIBL, *CURLIB
System driver program  . . . . .  *IBMPJLDRV
Text 'description' . . . . . .   *BLANK

                                                                    More...
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 617. Create Device Desc (Printer) (CRTDEVPRT) command (Screen 3)

```
                  Create Device Desc (Printer) (CRTDEVPRT)

Type choices, press Enter.


                            Additional Parameters

Remote network identifier  . . .   *NETATR       Name, *NETATR, *NONE
Workstation customizing object     *NONE         Name, *NONE
  Library  . . . . . . . . . . .                 Name, *LIBL, *CURLIB
Authority  . . . . . . . . . . .   *LIBCRTAUT    Name, *LIBCRTAUT, *CHANGE...

                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 618. Create Device Desc (Printer) (CRTDEVPRT) command (Screen 4)

13.5.3.2 Starting the printer

Once the printer is defined, it can be started using Operations Navigator. Complete these steps:

1. Start Operations Navigator by clicking **Start->Programs->IBM AS400 Client Access->AS/400 Operations Navigator**. The AS/400 Operations Navigator window appears (Figure 619).

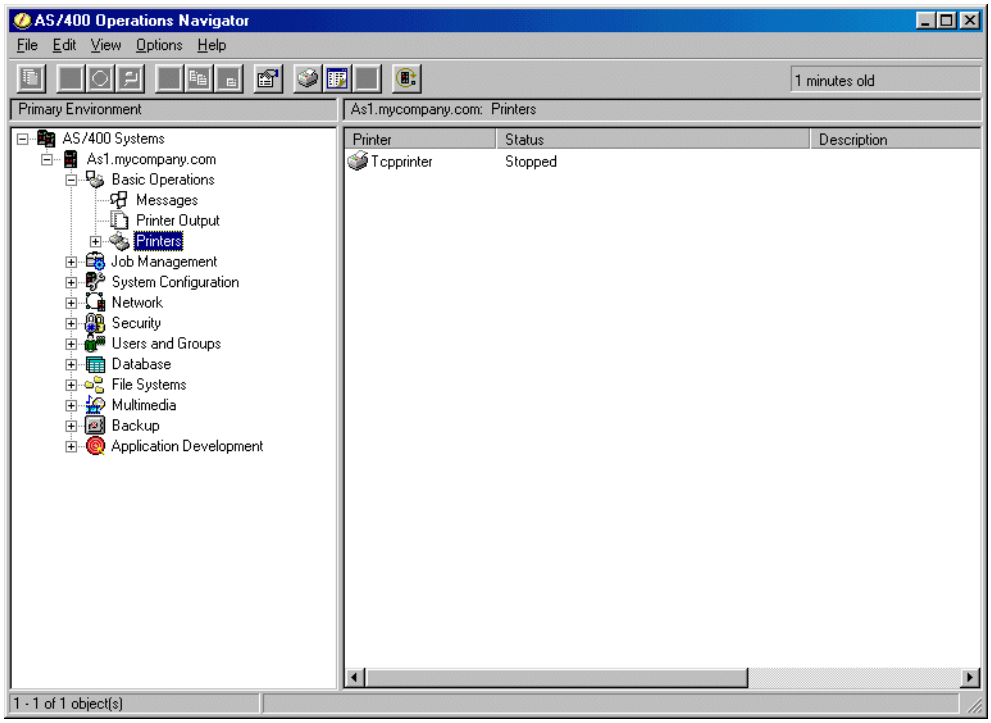


Figure 619. Operations Navigator: Printers

2. Double-click the system icon for the AS/400 system that you are configuring. The system components appear.
3. Double-click the **Basic Operations** icon. The operations components appear.
4. Click the **Printers** icon. The available printers appear.
5. Right-click the printer icon in the right-hand window. The context menu for the printer appears (Figure 620 on page 510).

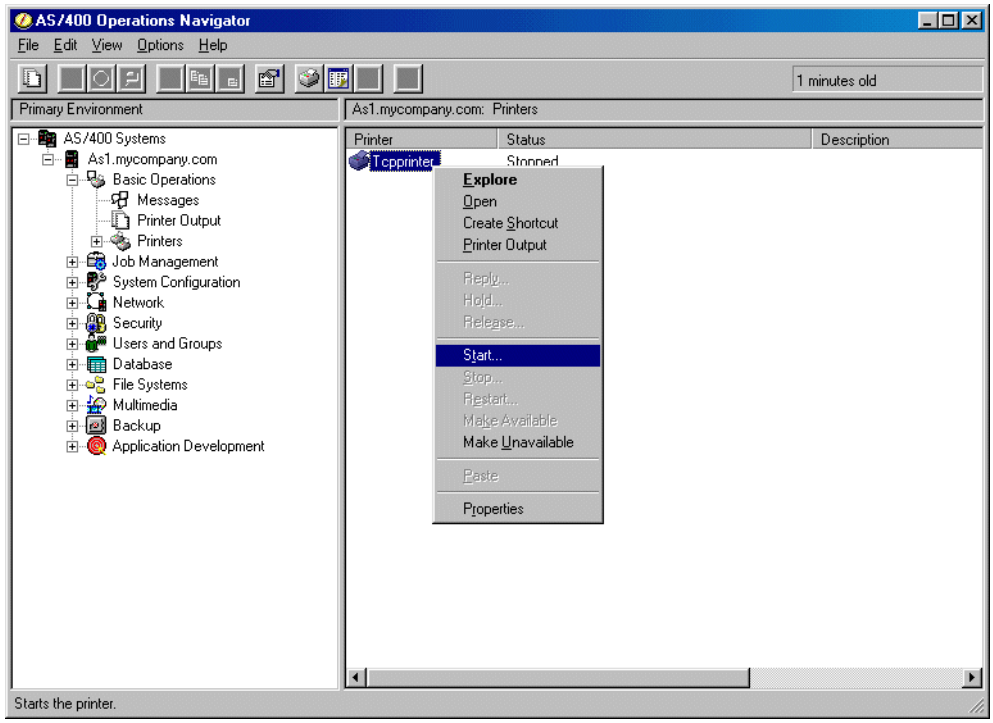


Figure 620. Start option on the Printers context menu

6. Select **Start** from the context menu. The Start panel appears (Figure 621).

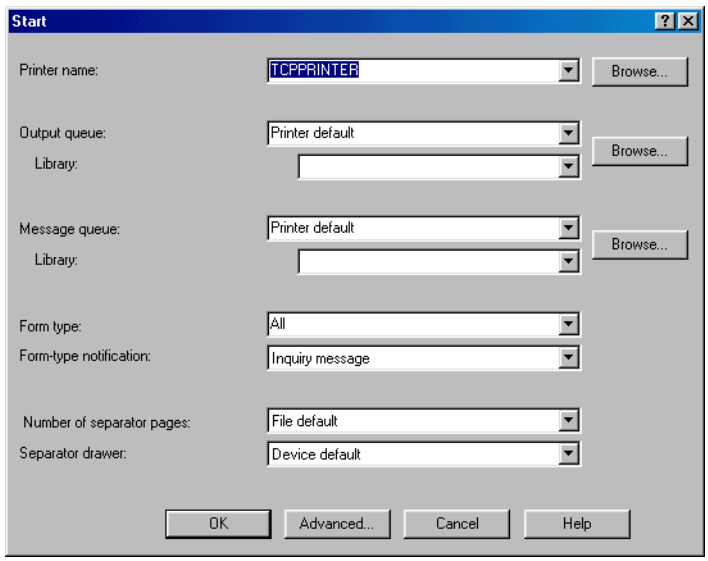


Figure 621. Start panel for Printers

7. Click **OK** to start the printer.

13.5.3.3 Moving the spooled file

Once the printer is started, you can use Operations Navigator to move the spooled file across to the printer. Follow these steps:

1. Click the **Printer Output** icon (Figure 622). The printer files are listed in the right-hand window.

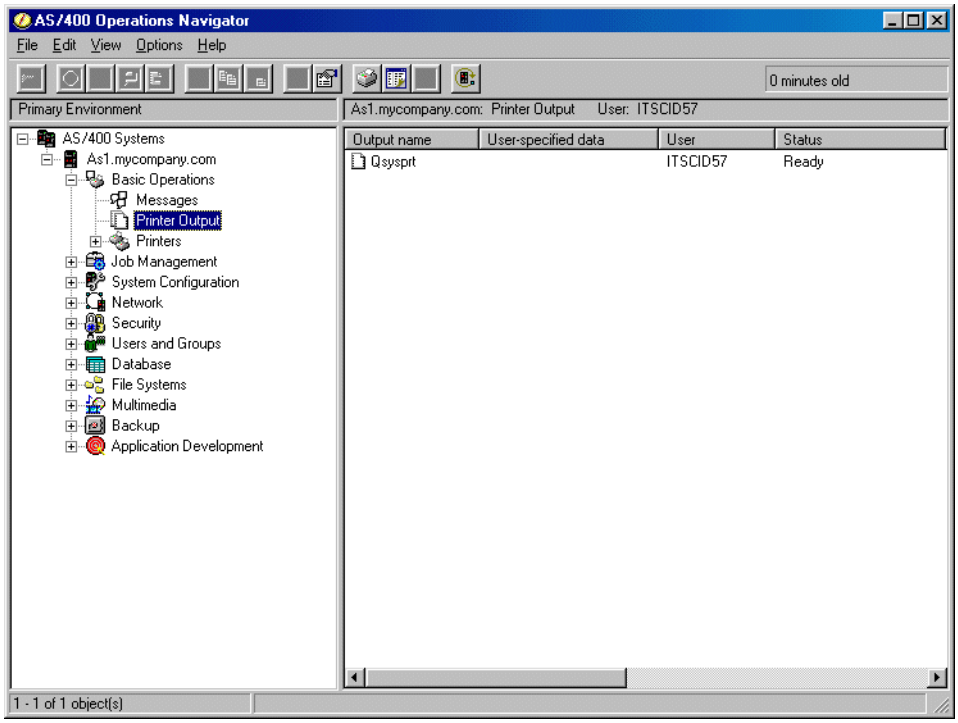


Figure 622. Operations Navigator: Printer output

2. Right-click on the file you want to print. The context menu appears (Figure 623).

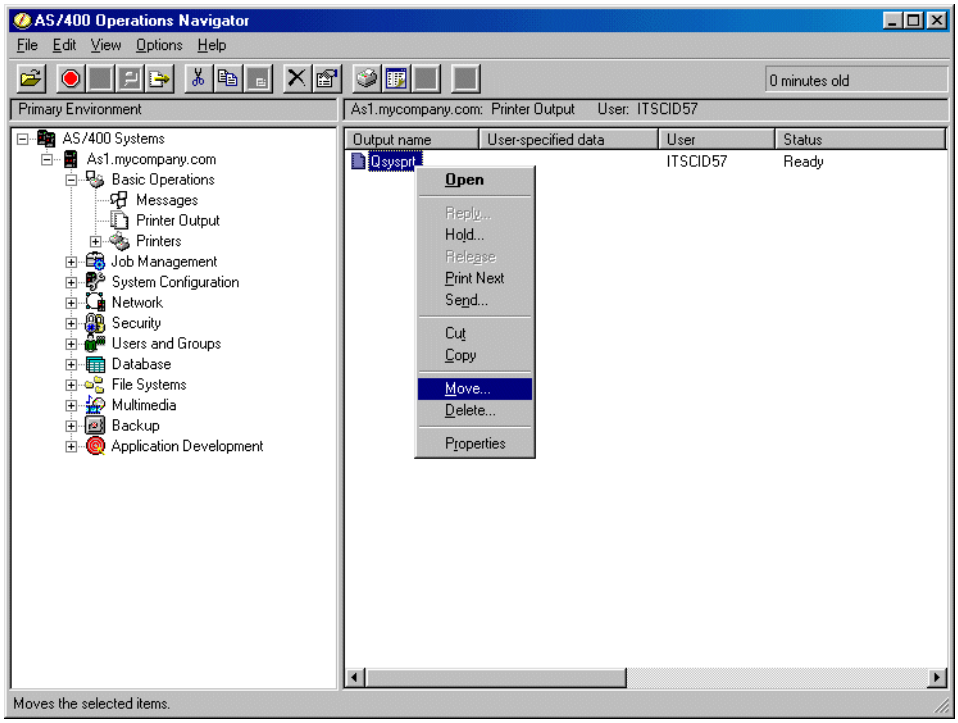


Figure 623. Move option on the Printer Output context menu

3. Select **Move** from the context menu. The Move panel appears (Figure 624).

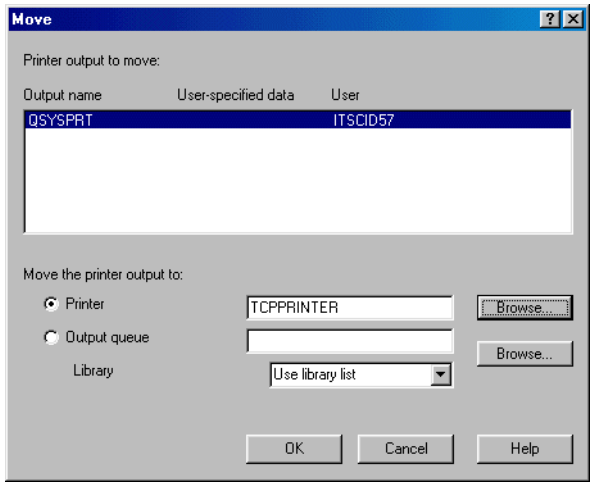


Figure 624. Move panel for printer output

4. Select the **Printer** radio button.
5. Type the name of the printer in the radio button.
6. Click **OK**.

The printer file is now on the printer.

13.5.4 Printing to an IBM Network Printer 12 using LPR/LPD

Some of the more sophisticated printers contain an embedded LPD Server. LPR can be used on the AS/400 system to send the files directly to the printer without any intervening host.

To send the file, you simply issue the `LPR` command and specify `*INTNETADR` for the Remote system parameter. After you enter this, the display expands to allow you to enter the IP address of the printer. As an alternative to entering the IP address, you can enter option `10` (Work with TCP/IP host table entries) from the CFGTCP menu to add a host table entry for the printer and specify the name in the remote system parameter.

As well as specifying the remote system name or address, you need to specify the details of the spooled file that you want to print (which can be collected using the Work with Spooled Files (`WRKSPLF`) command). You also need to specify a Destination type of `*OTHER` and the Transform SCS to ASCII parameter should be set to `*YES`.

When the file is to be transformed to ASCII, you are asked to specify a “Manufacturer type and model” for the destination server. For an IBM Network Printer 12, this is set to *IBM4312.

```
                      Send TCP/IP Spooled File (LPR)

Type choices, press Enter.

Remote system  . . . . . > *INTNETADR



Printer queue  . . . . . > PR1



Spooled file . . . . . > QSYSPRT       Name
Job name . . . . . . . . > QPADEV0003    Name, *
  User . . . . . >   ITSCID57    Name
  Number . . . . . >   002485      000000-999999
Spooled file number  . . . . . *ONLY         1-9999, *ONLY, *LAST
Destination type . . . . . *OTHER        *AS400, *PSF2, *OTHER
Transform SCS to ASCII . . . . . *YES          *YES, *NO
Manufacturer type and model  . . *IBM4312
                                                                    More...
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display       F24=More keys
```

Figure 625. Sending a file to IBM Network Printer 12 using LPR/LPD (Part 1)

```
                      Send TCP/IP Spooled File (LPR)

Type choices, press Enter.

Internet address . . . . . .   10.5.62.45









                                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display       F24=More keys
```

Figure 626. Sending a file to IBM Network Printer 12 using LPR/LPD (Part 2)

When all these parameters are set, this file prints when the command is issued.

If the printer queue specified is not valid, or does not exist on the destination LPD Server, you will receive error message TCP3719. When you specify the correct printer queue name, the file will print successfully.

13.5.5 Using an initial program to map printer IP addresses

The QDCRDEVD API can be used by an initial program to enable the mapping of the IP address of the client to a particular printer. This allows your application to control where and how to send the print file. This mapping sends print files back to the client's workstation, to a network printer, or to a printer on the application system.

This sample demonstrates how to use the QDCRDEVD API within an initial program to modify the IP address of the output queue, depending on the IP address (and optionally the user name) of the display device. The sample program is used with output queues that are configured to use LPR/LPD to send the spooled file to the remote printer or print server.

Note

The IP address mapping sample requires the following minimum PTF levels:

- **V3R2**
  - 5763TC1: SF38885
  - 5763SS1: SF38876, SF38688, SF38886
    (PTF cover letter SF38876 describes API field offsets)
- **V3R7**

- 5763TC1: SF385535
- 5763SS1: SF37172, SF38357, SF38536
  (PTF cover letter SF38565 describes API field offsets)

13.5.5.1 Creating and compiling the initial program

The first step is to create and compile the initial program. The file can be created with the command:

```
CRTPF FILE(QCSRC) RCDLEN(240) MBR(QRMTWTR) MAXMBRS(*NOMAX) SIZE(*NOMAX)
AUT(*CHANGE) TEXT('Sample IP mapping C source code')
```

Once the file is created, the C source can be entered or copied to the file using FTP. The source code for the sample program and the compiled sample program are available with the save file for this redbook, which is available at:
`http://www.redbooks.ibm.com`

You can locate the file under the **Additional materials** tab.

When the source code is available, the program can be compiled with the command:

```
CRTBNDC PGM(QRMTWTR) SRCFILE(QCSRC) TEXT('TCP/IP Print Control')
LANGLVL(*EXTENDED) SYSIFCOPT(*NOIFSIO) REPLACE(*YES) USRPRF(*USER) AUT(*USE)
TGTRLS(*CURRENT) SYSINC(*YES)
```

Note

The sample program provided only changes the output queue parameters when the name of the output queue and the user profile name are the same. This is a restriction imposed by the sample program and not a restriction of the API.

13.5.5.2 Creating the map file to be used by the program

The exit sample provided reads in a file to map device IP addresses (and optionally user names) to the relevant printer IP addresses. Figure 627 shows a sample mapping file.

The physical file for the map file can be created with the command:

```
CRTPF FILE(QUSRSYS/QRMTWTR) RCDLEN(240) MBR(MAP) MAXMBRS(*NOMAX) SIZE(*NOMAX)
AUT(*USE) TEXT('Printer Map')
```

Once the file has been created, the contents can be entered or transferred across. The sample map file is available with the save file for this redbook on the IBM Redbooks home page at: `http://www.redbooks.ibm.com`

```
# Start of file
#
# Rules:
#
# 1.) Any line with a '#' in column 1 is a comment line
# 2.) Case sensitivity is maintained when file is read
# 3.) A '|' vertical bar is used as a field delimiter
# 4.) Any field with a '#' means to use blanks as the value
# 5.) A '*' character can be used as a wildcard in IP address octets
# 6.) A '*CLIENT' for printer IP means to substitute client IP
# 7.) DestType must be: *SAME or any DESTTYPE parm value
# 8.) Transform must be: *SAME, *YES, or *NO (a TRANSFORM parm value)
# 9.) Type/Model must be: *SAME or any MFRTYPMDL parm value
#
# Printer    Print IP     Client IP     User       Xform  MfgType     DestType
# -------    --------     ---------     --------   -----  ---------   --------
| pctest   | *CLIENT    | 10.5.62.161 | #        | *YES | *IBM4312  | *OTHER  |
| PASS     | 10.5.62.45 | 10.5.62.*   | #        | *YES | *IBM4312  | *OTHER  |
| lpt1     | *CLIENT    | *.*.*.*     | #        | *YES | *HP4      | *OTHER  |
#
# End of file
```

Figure 627. Sample map file for the QRMTWTR sample

The mapping file is formatted with the following rules:

- A "#" in column 1 is a comment line.
- A "|" is used to separate fields.
- Each mapping entry is included on a single line.

The fields used by the sample exit program are shown in Table 30.

Table 30. QRMTWTR sample mapping file fields

Field	Description
Printer	The name of the remote LPD printer queue.
Print IP	The IP address of the remote LPD server. The special value *CLIENT means that the printer is to be set to the same address as the display device.
Client IP	The IP address of the display device that use this entry. "*" can be used as a wildcard to match multiple addresses with a common prefix.
User	The user name that will use this entry. The value "#" in this field means that the entry applies to all user names.

Field	Description
Xform	*SAME or a valid TRANSFORM parameter value for the LPR command.
MfgType	*SAME or a valid MFRTYPMDL parameter for the LPR command.
DestType	*SAME or a valid DESTTYPE parameter for the LPR command.

13.5.5.3 Changing the user profile to call the initial program

The user profile can be used to call the initial program with the command:

```
CHGUSRPRF USRPRF(USRPRFNM) INLPGM(*LIBL/QRMTWTR) OUTQ(QUSRSYS/USRPRFNM)
```

13.5.5.4 Creating the custom output queue

Note

The sample program requires the name of the output queue and the user profile name to be the same. If they do not match, the output queue information will not be modified.

The output queue can be created with the command:

```
CRTOUTQ OUTQ(QUSRSYS/USRPRFNM) RMTSYS(*INTNETADR) RMTPRTQ(PASS) AUTOSTRWTR(1)
CNNTYPE(*IP) DESTTYPE(*OTHER) TRANSFORM(*YES) MFRTYPMDL(*IBM4312)
INTNETADR('10.5.62.45')
```

13.5.5.5 IP address mapping results

The sample program and mapping file listed above produces the following results:

- When a user connects to the AS/400 system from IP address 10.5.62.161, they will have their output queue set to use the LPD server at the same address. The LPD server at this address has a printer queue called “pctest” which is directed to an IBM Network Printer 12.

  The mapping program modifies the default *OUTQ for client 10.5.62.161 with the command:

  ```
  CHGOUTQ OUTQ(USRPRFNM) RMTSYS('10.5.62.161') RMTPRTQ('pctest')
  AUTOSTRWTR(1) CNNTYPE(*IP) TRANSFORM(*YES) MFRTYPMDL(*IBM4312)
  DESTTYPE(*OTHER)
  ```

- When a user connects to the AS/400 system from any IP address on the 10.5.62.* subnet other than IP address 10.5.62.161, they will have their output queue set to use the LPD server at 10.5.62.45. The LPD server at this address is an IBM Network Printer 12 itself, and it accepts print files on a queue called “PASS”.

  This mapping program modifies the default *OUTQ for this client with the command:

  ```
  CHGOUTQ OUTQ(USRPRFNM) RMTSYS('10.5.62.45') RMTPRTQ('PASS') AUTOSTRWTR(1)
  CNNTYPE(*IP) TRANSFORM(*YES) MFRTYPMDL(*IBM4312) DESTTYPE(*OTHER)
  ```

- When a user connects to the AS/400 system from any other IP address (such as using an ISP), they will have their output queue set to use the LPD server on the client workstation. The LPD server on the client workstation has a printer queue called “lpt1” that directs print files to a HP Laserjet 4 compatible printer.

This mapping program modifies the default *OUTQ for this client with the command:

```
CHGOUTQ OUTQ(USRPRFNM) RMTSYS('1.2.3.4') RMTPRTQ('lpt1') AUTOSTRWTR(1)
CNNTYPE(*IP) TRANSFORM(*YES) MFRTYPMDL(*HP4) DESTTYPE(*OTHER)
```

Chapter 14. Using routing with the AS/400 system

This chapter covers IP routing in relation to the AS/400 system. General information about routing and different types of routing, static and dynamic, are discussed. Several scenarios, both using static and dynamic Routing Internet Protocol (RIP), are shown.

Note: The word *router* is preferred to *gateway* throughout this chapter.

14.1 Routing in a network: An overview

This section provides a brief overview of routing in IP networks. It is not intended as a complete description of routing in general, so please refer to *TCP/IP Tutorial and Technical Overview,* GG24-3376, and *IBM 2210 Nways Multiprotocol Router Description and Configuration Scenarios - Volume 1,* SG24-4446, for further information on routing.

From a host point of view, a network is nothing but a big cloud. Two hosts communicating with each other know nothing about the path the data packets travel from host A to host B. Figure 628 shows this. The hosts are freed from the burden of controlling and managing how to get from A to B.

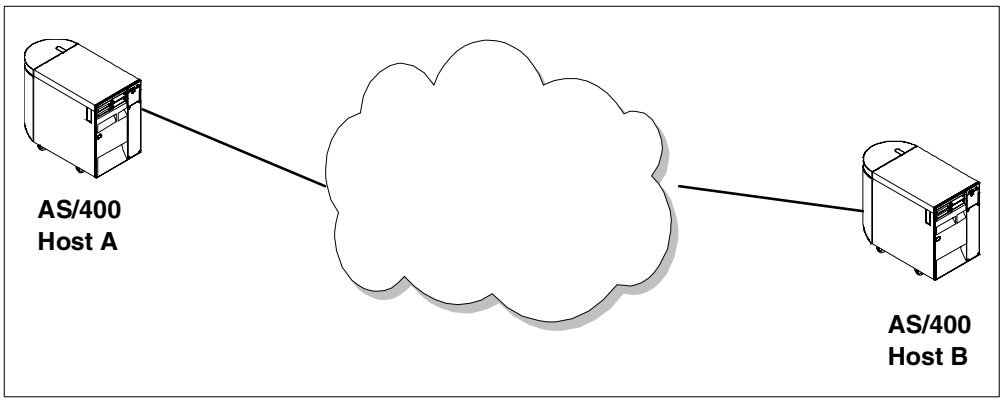


Figure 628. The network as seen from the two hosts

In reality, a complete network is usually much more complex. An example is the Internet, which is a large number of networks connected by routers.

A company network, the Intranet, can be quite complex too, connecting networks across buildings, locations, cities, states and countries. Figure 629 on page 520 shows an example of several networks connected by routers.

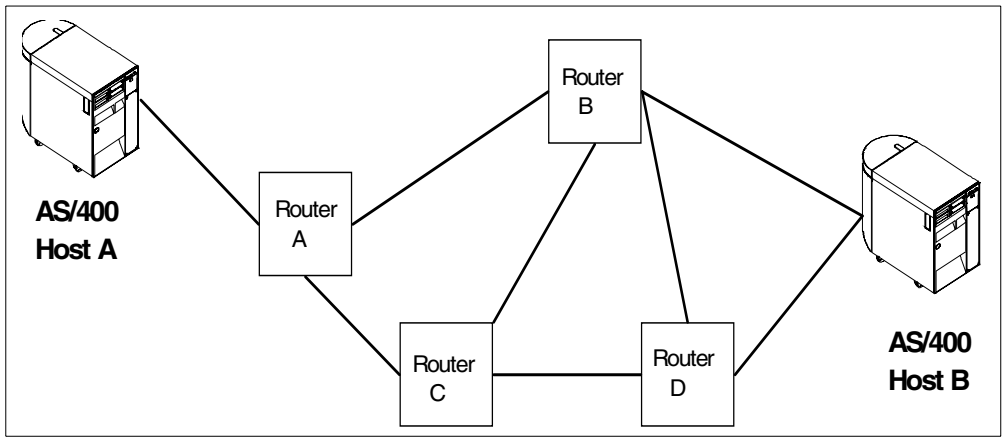


Figure 629. The real network is connected by routers

On its way from host A to host B, the data packet travels across the network, passing through several routers. Each router being passed in turn examines the data packet received and decides how the data packets should be forwarded. The data packets are processed by the TCP/IP protocol stack on each host and router. Since the routing of the data packets takes place at the network-layer (IP), the routers only used the lower levels of the TCP/IP protocol stack, where the hosts use the complete TCP/IP protocol stack. Refer to Figure 630 for an overview.

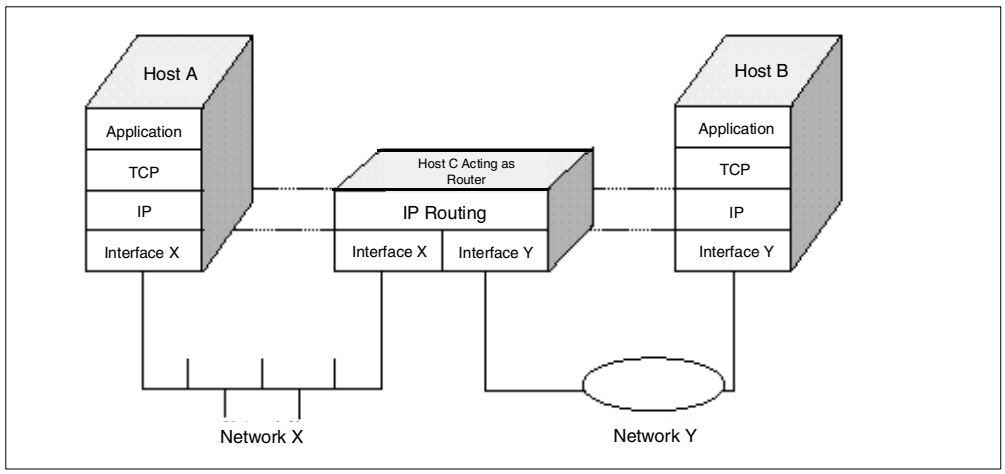


Figure 630. Host and router IP operation

14.2 Types of routing

Basically, two kinds of routing exists: static and dynamic. Both have their strengths and weaknesses.

14.2.1 Static routing

Static routing provides a simple approach to routing. The system administrator manually specifies what router should be used when connecting to a specific network. One entry is required for each network accessed. The system administrator can specify a *catch-all* routing entry, if the destination network is not

found in the routing table. This is known as the default route, the *DFTROUTE entry. A sample static routing table is shown in Figure 631.

```
                              Work with TCP/IP Routes
                                                                 System:   AS21
Type options, press Enter.
  1=Add   2=Change   4=Remove   5=Display

     Route            Subnet           Next             Preferred
Opt  Destination      Mask             Hop              Interface

     *DFTROUTE        *NONE            10.153.200.1     *NONE
     10.151.0.0       255.255.0.0      10.153.200.2
     10.152.0.0       255.255.0.0      10.153.200.3

                                                                      Bottom
F3=Exit      F5=Refresh   F6=Print list   F11=Display type of service
F12=Cancel   F17=Top      F18=Bottom
```

Figure 631. Static routing table on the AS/400 system

14.2.1.1 Advantages

Static routing is fast and simple. The system administrator has complete control of where the data packets are forwarded.

14.2.1.2 Disadvantages

The manual maintenance of the routing table becomes more and more complex and overwhelming as the number of networks increase. This is likely to introduce configuration errors and cause some parts of the network to become unavailable.

In case of a router failure, the static routing is not able to reconfigure to the new state of the network. Static routing is not able to adapt new routes using alternative routers.

14.2.2 Dynamic routing

Dynamic routing is provided by Interior Gateway Protocols (IGP), such as Routing Internet Protocol (RIP) and Open Shortest Path First (OSPF). These protocols allow the system administrator to configure the hosts as part of an RIP or OSPF network. The complete topology of the network is automatically propagated across the network when new hosts enter the network or when changes occur in the network, for example, hosts crashing.

14.2.2.1 Advantages

Dynamic routing requires minimal maintenance. Automatic reconfiguration of the routing table occurs when the network changes.

14.2.2.2 Disadvantages

There aren't any major disadvantages. However, to a certain extent, the administrator's overview of the network and the routing is lost when the protocols take care of the routing.

14.2.3 When to use what type of routing

When designing a network, the administrator must decide what type of routing to use in the network. The following sections offer a rough recommendation on which protocol to use. Please refer to *TCP/IP Tutorial and Technical Overview,* GG24-3376, for more detailed information. The terms small, medium, and large are relative to the organization and the country where the network is located.

14.2.3.1 Small network (one to five networks)

Static routing can be used as an easy way of connecting the systems. If dynamic routing is required, RIP is fine.

14.2.3.2 Medium network (five to 15 networks)

RIP is OK to start with, but due to limitations in the RIP protocol, not more than 15 networks can be accessed using RIP. If the network is expected to grow, OSPF is recommended.

14.2.3.3 Large network (more than 15 networks)

OSPF is the choice when larger networks are built. There are no limitations to the number of networks and recovery speed is fast in the case of router failures.

14.3 The AS/400 system as a router

The AS/400 system can be used as a router connecting TCP/IP networks. The AS/400 system, at OS/400 V4R3M0, supports static routing and the RIP protocol.

We recommend that you do not use the AS/400 system as the *only* router in a network. Dedicated routers are recommended. These routers are 100% dedicated to routing, where the AS/400 system typically is busy doing application serving, file serving, printer serving, Web serving, client serving etc. Furthermore, the routing is not possible when the AS/400 system is in *restricted state*, for example, when doing a complete system save.

In a RIP network, the AS/400 system typically acts as a pure host, only receiving RIP messages and not supplying the network with messages.

In order for the AS/400 system to act as a *primitive* router, the IP datagram forwarding (IPDTGFWD) parameter on the Change TCP/IP Attributes (CHGTCPA) command should be set to `*YES`. See Figure 632.

```
                    Change TCP/IP Attributes (CHGTCPA)

Type choices, press Enter.

TCP keep alive . . . . . . . . . TCPKEEPALV     120
TCP urgent pointer . . . . . . . TCPURGPTR      *BSD
TCP receive buffer size  . . . . TCPRCVBUF      8192
TCP send buffer size . . . . . . TCPSNDBUF      8192
UDP checksum . . . . . . . . . . UDPCKS         *YES
IP datagram forwarding . . . . . IPDTGFWD       *YES
IP source routing  . . . . . . . IPSRCRTG       *YES
IP reassembly time-out . . . . . IPRSBTIMO      10
IP time to live  . . . . . . . . IPTTL          64
ARP cache timeout  . . . . . . . ARPTIMO        15
Log protocol errors  . . . . . . LOGPCLERR      *NO




                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 632. Changing the IP Datagram Forwarding (IPDTGFWD) parameter

Changing this parameter to *YES enables the flow of IP datagrams across an AS/400 system if the system has more the one IP interface (multihomed). This should be a security consideration, since the system can now route IP datagrams ahead in the network. If you do not need the AS/400 system to act as a primitive router, you should make sure that the IP datagram forwarding (IPDTGFWD) parameter is set to *NO.

Please note that the AS/400 system cannot act as a router if the TCP/IP communication protocol not is active at the system. In other words, the Start TCP/IP (STRTCP) command must be run before this can happen. This command is usually found in the program being executed at IPL.

14.4 Routing Information Protocol (RIP) on the AS/400 system

This section provides an overview of the AS/400 Routing Information Protocol (RIP) support in V4.

14.4.1 RIP concepts

One of the basic functions of IP is its ability to form connections between different physical networks. This is due to the flexibility of IP to use almost any physical network below it, and to the IP routing algorithm. A system that does this is called a *router*.

The routing function is a part of the IP layer, but the primary function of a routing protocol is to exchange routing information with other routers.

When two host systems are attached in different subnetworks, IP packets should be routed through the subnetworks to reach the each end system. A router between two subnetworks routes IP packets having the information of systems

attached to the subnetworks. The AS/400 system, before V4R1, can be a router, but the routing information has to be maintained manually, using the TCP/IP route configuration option of the Configure TCP/IP (CFGTCP) command. This is because the AS/400 system does not have the dynamic routing protocol before V4R1. Manually maintaining the routing information is cumbersome. Moreover, each time a network is updated, the routing information has to be maintained manually. The RIP server function, called *Route Daemon (RouteD)* gives automatic routing information maintenance based on the RIP information from routers that are capable of advertising the RIP packets.

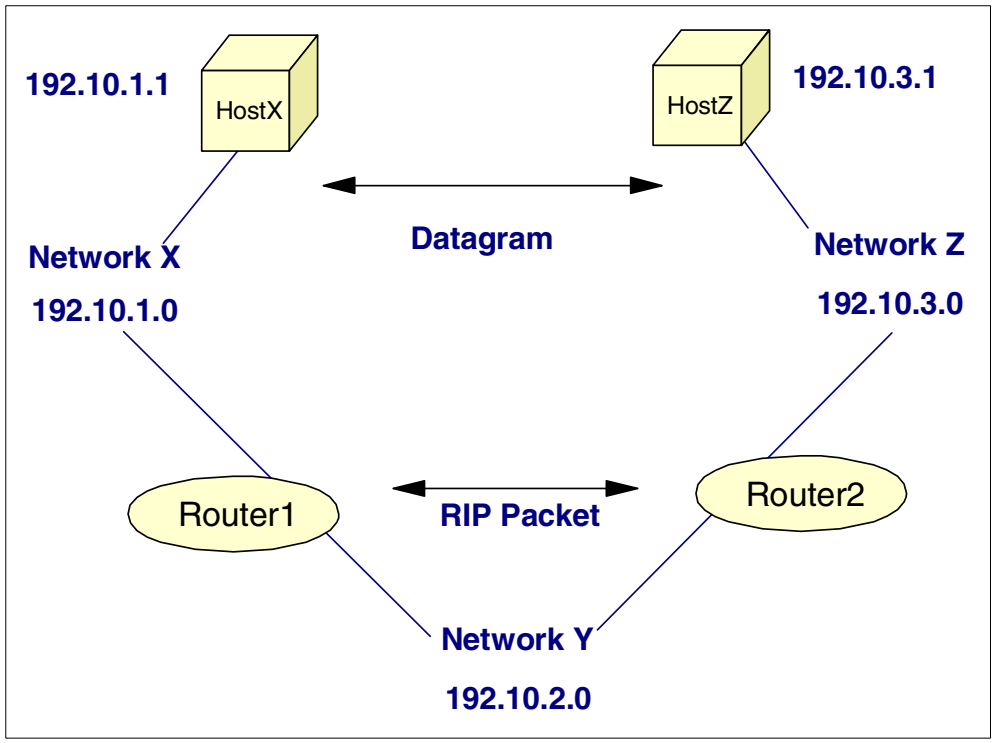


Figure 633. Routing information in the network

RIP is a widely used routing protocol today. It is an Interior Gateway Protocol (IGP) used to assist TCP/IP in the routing of IP data packets within an autonomous system (a group of networks and gateways controlled by a single administrative authority). Within an autonomous system, the network administrator has control over subnetting and routing architecture. Dynamic routing protocols allow you to handle larger networks where automatic switching to redundant routes or use of multiple routers on a network is desirable.

RIP uses UDP as a transport protocol (at destination port 520) and communicates with other adjacent routers informing each other of other networks to which it is connected. RIP is considered a distance vector routing protocol. It partitions the participants in active and passive (silent) ones.

Typically, gateways are running RIP in active mode and hosts are running RIP in passive mode. Active gateways advertise their routes to others, while passive machines listen and update their routes based on advertisements, but they do *not* advertise themselves. Both active and passive RIP participants listen to all broadcast messages and update their tables according to the vector-distance algorithm. Each router running RIP in active mode broadcasts a message every

30 seconds. This message contains the IP network address of the network and the integer distance to that network. Once a routing device receives an update, it compares with the existing routing table and updates it if necessary. If the routing update includes a new destination network, it is added to the routing table. If the router receives a route with a smaller metric to an existing destination, it replaces the existing route. If an entry in the update message has the same destination network and gateway but a different metric, it uses the new metric to update the routing table.

RIP uses hop count metric to measure the distance to a destination: the number of hops equals the number of gateways.

Once a gateway learns a route from another gateway, it must keep that route until it learns of a better one (with strictly lower cost) to prevent looping. When a gateway adds a route in its table, it starts a timer for that route. The timer must be restarted when the gateway receives another RIP message advertising that route. The route becomes invalid if 180 seconds pass without the route being advertised again (60 seconds later the route is deleted from the local routing table). The RIP protocol assumes that, if a route is not received again within 180 seconds, it is no longer available.

The maximum possible distance with RIP is 16. This looks unsuitable for the largest corporate networks. RIP's interpretation of infinity as the number 16 relates to the Time To Live (TTL) field in the IP layer header. Each time a packet travels through a router, its TTL field (with initial value of 15) is decreased by one. When the TTL value reaches 0, it is discarded and no longer exists in the network. This feature is implemented to stop a packet caught in a routing loop from being switched back and forth forever between routers.

Each router transmits packets with destination and cost pairs to its neighbors. The packets determine whether the AS/400 routing table should be updated.

Figure 634 on page 526 shows you roughly how RIP works. Gateway G1 broadcasts a message on network 2 that contains the pair (1,1), meaning that it can reach network 1 at cost 1. Gateway G2 receives the broadcast and adds a route to network 1 through G1 (at cost 2). Later, gateway G2 includes the pair (1,2) when it broadcasts the RIP message on network 3.

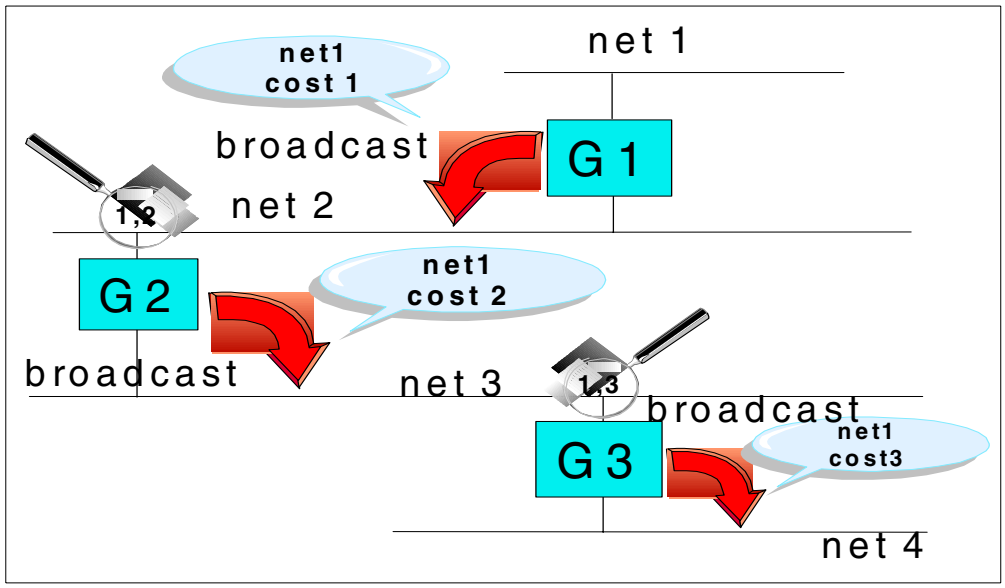


Figure 634. Broadcasting RIP messages

14.4.2 AS/400 RIP support

The following section provides a detailed description of the implementation of the RIP protocol on the AS/400 system. Both RIP v1 and RIP v2 are covered.

14.4.2.1 AS/400 RIP V1 support: OS/400 V4R1

The following list describes the features of the RIP v1 implementation in OS/400 V4R1, followed by a more detailed description:

- RouteD provides support for RIP on the AS/400 system
- RIP uses UDP as transport protocol
- Distance vector routing protocol
- RIP messages are broadcasted
- RouteD configuration on the AS/400 system
  - Member QATOCRTDC.CONFIG in QUSRSYS
  - RIP configuration entries
  - Add, change, copy, remove individual entries
  - Gateway and interface options can be specified
    - Static
    - Passive
    - Active

OS/400 V4R1 started supporting RIP version 1.

The Route Daemon (RouteD) provides support for the Routing Information Protocol (RIP) on the AS/400 system. RouteD software was originally designed at the University of California at Berkeley. The name comes from the UNIX convention of attaching to the names of daemon processes.

To manage the RouteD with the AS/400 system, use the Configure TCP/IP RouteD (CFGTCPRTD) command to start configuration tasks, such as changing its attributes and editing the routing information database. The AS/400 system can operate in active or passive mode.

14.4.2.2 AS/400 RIP V2 support: AS/400 V4R2

The following list describes the features of the RIP v2 implementation in OS/400 V4R2, followed by a more detailed description:

- Multicasting
  - Full subnet addressing
  - Authentication
  - Route tagging
- RouteD server functions access:
  - Using command line interface
  - Through Operations Navigator
- RIP_INTERFACE statement replaces OPTIONS INTERFACE
  - Options configured on a per interface basis
  - New supply value: SUPPLY RIP2
  - New DIST_ROUTES_IN option
- RIPv2 is defined in RFC1723

An existing RouteD configuration file QUSRSYS/QATOCRTDC will automatically be migrated at installation time of OS/400 V4R2.

RIP v2 supports multicasting. This can reduce the load on hosts which are not listening to RIP v2 messages.

RIP v2 includes full subnet addressing that enables Classless Inter-Domain Routing (CIDR) support. CIDR does not route according to the class of the network number, but according to the high order bits of the IP address (also called IP prefix). Each CIDR routing entry contains a 32-bit IP address and up to a 32-bit network mask which, together, give the length and value of the IP prefix. CIDR handles the routing for a group of networks with a common prefix with a single routing entry. This enables that multiple class C networks work as if they are a single network.

Other improvements over RIP v1 include authentication and route tagging. Authentication improves security by using the COMMUNITY parameter. Route tagging improves compatibility with exterior gateway protocols (EGP). Those protocols are designed to route between autonomous systems.

You can access RouteD server functions via the command line interface or the Operations Navigator interface. However, not all RouteD functions are available on both interfaces.

In V4R2, RIP_INTERFACE statement replaces OPTIONS INTERFACE of the V4R1 route table configuration. The RIP_INTERFACE statement is used to specify all routing options on a per interface basis. The RIP_INTERFACE statement now contains the functionality for defining routes and creating static routes. Prior to V4R2, these functions existed in the NET and HOST statements. For V4R2, the HOST and NET statements are no longer valid on the Work with RouteD Configuration (WRKRTDCFG) screen. This information is now being specified on the Add TCP/IP Route (ADDTCPRTE) and Change TCP/IP Route (CHGTCPRTE) commands.

The new supply value SUPPLY RIP2 and the new option DIST_ROUTES_IN are discussed in “Working with configuration V4R2 and later” on page 540.

14.4.2.3 AS/400 RIP V1 and V2 support: AS/400 V4R2

The following list describes the features of the RIP v1 and RIP v2 implementation in OS/400 V4R2, followed by a more detailed description:

- RIPv1 and RIPv2 is supported in OS/400 V4R2
  - RIPv1 broadcasts RIPv1 responses
  - RIPv2 multicasts RIPv2 responses
- RIPv2 multicast support does not imply that the AS/400 will be a multicast router.
- V4R2 implementation limitations
  - No SNMP MIB extension
  - No Frame Relay support
  - No Triggered RIP
  - No MD5 Authentication
  - No CIDR route aggregation
  - RIPv1 will not multicast RIPv2 responses
- Dead Router detection
- new RIP functions configured with Operations Navigator

Both RIP versions, RIP v1 and RIP v2, are supported in OS/400 V4R2. RIP v1 broadcasts RIP v1 responses. and RIP v2 multicasts RIP v2 responses.

RIP v2 uses multicasting to share routing information, but it does not mean that the AS/400 system is a multicast router.

The implementation of RIP has the following limitations in the V4R2 release:

- The RIP v2 SNMP MIB extension is not implemented (RFC1389, RFC1724).
- Frame Relay is not supported.
- Triggered RIP is not implemented (RFC2091, RFC2092).
- RIP v2 MD5 Authentication is not implemented (RFC2082).
- CIDR route aggregation is not implemented.
- RIP v1 does not multicast RIP v2 responses.

The AS/400 system is unique in that it has implemented “Dead Router” detection. When the IP forwarding code is unable to connect to the selected router, the route in the routing table is flagged with “Not Available” and the RouteD continues to find a usable route. To enhance this, AS/400 implementation supports duplicate routes to the same network. Each route is given a precedence (priority) to allow them to be easily ordered.

All new RIP functions are configured with the Operations Navigator.

The following V4R2 changes are made to both RIP v1 and RIP v2:

- With RIP v1 in V4R1, static routes could be defined within the RIP configuration file, but they could also be defined through the standard TCP/IP configuration. With V4R2, the definitions of static routes will be combined into a single configuration file. The only mechanism for adding a static route is to use the Add TCP/IP Route (ADDTCPRTE) command.
- RIP v1 only supports broadcast network interfaces: Token-Ring, Ethernet, and FDDI.

- With V4R2, RIP v1 and RIP v2 both run over Point-to-Point links, such as SLIP and PPP. This can be specified through the Operations Navigator on the PPP connection screen (TCP/IP settings tag) by clicking on the routing button. It is possible to add a static route through this interface, too.
- RIP v1 in V4R1 obtains the routing configuration information from QATOCRTDC.CONFIG file. RouteD needs to be restarted before these changes take effect.
- The RouteD configuration changes are dynamically reflected without restarting the server.
- Static route precedence with a duplicate route priority parameter. This specifies the duplicate route priority of the static route. Routes with a high duplicate route priority (DUPRTEPTY) are tried before routes with a low one. The valid range is 1 to 10. The default value being used is 5.

The Add TCP/IP Route (ADDTCPRTE) and Change TCP/IP Route (CHGTCPRTE) commands have new and changed parameters that integrate static routes into a single file. The new parameters are: BNDIFC, METRIC, REDST, DUPRTEPTY. See Figure 635.

```
                        Add TCP/IP Route (ADDTCPRTE)

Type choices, press Enter.

Route destination  . . . . . > ' '
Subnet mask  . . . . . > ' '
Type of service  . . . . . .   *NORMAL       *MINDELAY, *MAXTHRPUT...
Next hop . . . . . > ' '
Preferred binding interface  . .   *NONE
Maximum transmission unit  . . .   576           576-16388, *IFC
Route metric . . . . . .   1             1-16
Route redistribution . . . . . .   *NO           *NO, *YES
Duplicate route priority . . . .   5             1-10




                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 635. Parameters when adding a new static route

The parameters shown in Figure 635 are explained in the following list:

- **Route destination (RTEDEST)**:

  The default multicast route entry (*DFTMCAST) value is added to the Route Destination parameter. Allows the addition of default multicast routes. The default multicast routes are used to select the interface over which multicast packets should be sent when the application does not specify the interface.

- **Preferred binding interface (BNDIFC)**:

  This parameter is only useful when the destination has multiple IP interfaces. This allows a route to be bound to a specific IP interface. The binding is preferred and not absolute. Another active interface that uses the same line description will be used if the bound interface is not available. If the IP interface being specified is active, this route will bind to it. If it is not active, an

alternate and active IP interface that uses the same line description will be used. If the specified IP interface is not active, and no active alternate interfaces using the same line description can be found, any active IP interface on the same network will be used, regardless of line description. If no active IP interfaces on the same network as the specified interface can be found, the route will be disabled.

- **Route metric (METRIC)**:

  This parameter associates the “cost” assigned to this route. The metric cost of a route is a factor in determining the desirability of the route. A value of 1 is a route that is closest. A route with a value of 15 is relatively far away and a route with a metric of 16 is unreachable (infinity). Desirability decreases as the metric value (distance) increases. The default value for this parameter is 1.

- **Route redistribution**

  Specifies whether this static route information will be redistributed in the future. Traffic on this route can be reduced by specifying *NO for this parameter.

  – *YES is analogous to the RIP v1 specification of STATIC. This is advertised to the network.

  – *NO is analogous to the RIP v1 specification of PASSIVE. This is not advertised.

- **Duplicate route priority (DUPRTEPTY)**:

  Allows ordering of duplicate routes within an internal route table. If multiple routes are available, DUPRTEPTY is used to determine which duplicate route to try first. Higher priority values will allow a route to be chosen over a route that specifies a lower priority. The valid range is 1 to 10. The default value is 5.

14.4.3 Managing RIP support

To run the RouteD on the AS/400 system, the daemon should be started. This section provides information on how to start, stop, and configure the RouteD daemon on the AS/400 system.

14.4.3.1 Starting the RouteD daemon

This section describes how to start the RouteD daemon, both using the Operations Navigator and the green-screen interface.

Using Operations Navigator

1. Start Operations Navigator by clicking **Start->Programs->IBM AS400 Client Access->AS/400 Operations Navigator**. The AS/400 Operations Navigator window appears (Figure 636).

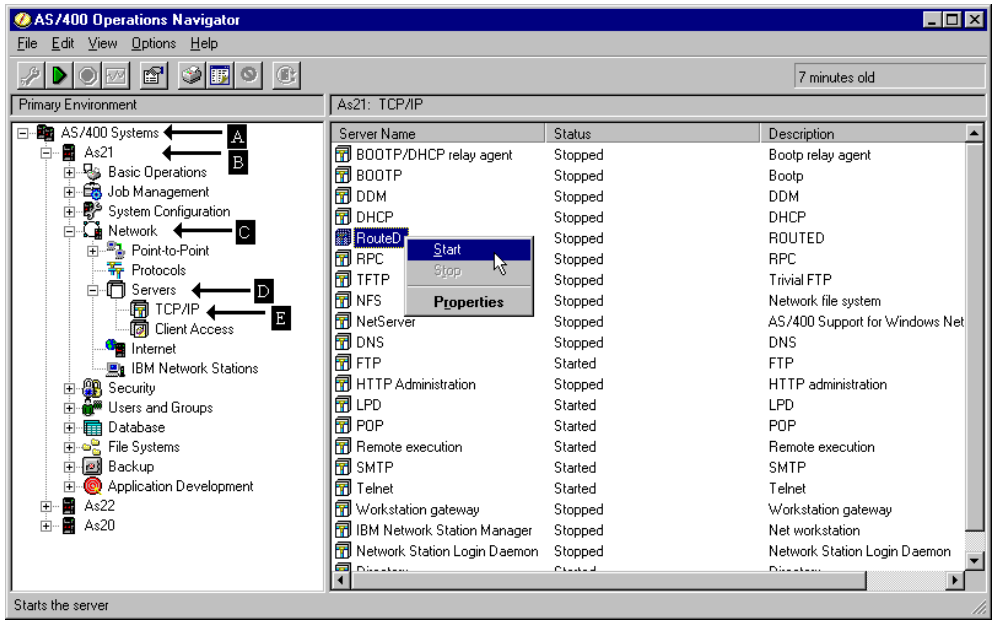


Figure 636. Starting the RouteD daemon using Operations Navigator

2. Double-click the AS/400 network icon (A).
3. Double-click the system icon (B) for the AS/400 system that you are configuring. The system components appear.
4. Double-click **Network** (C). The network components appear.
5. Double-click **Servers** (D). The available protocols appear.
6. Double-click **TCP/IP** (E). The available services appear in the right window.
7. Right-click on the **RouteD** item to see the menu. Click **Start** as shown in Figure 636.

Using the green-screen interface

Use the Start TCP/IP Server (`STRTCPSVR`) command specifying the RouteD server:

```
STRTCPSVR SERVER(*ROUTED)
```

14.4.3.2 Ending the RouteD daemon

This section describes how to stop the RouteD daemon, both using the Operations Navigator and the green-screen interface.

Using Operations Navigator

1. Start Operations Navigator by clicking **Start->Programs->IBM AS400 Client Access->AS/400 Operations Navigator**. The AS/400 Operations Navigator window appears (Figure 637 on page 532).

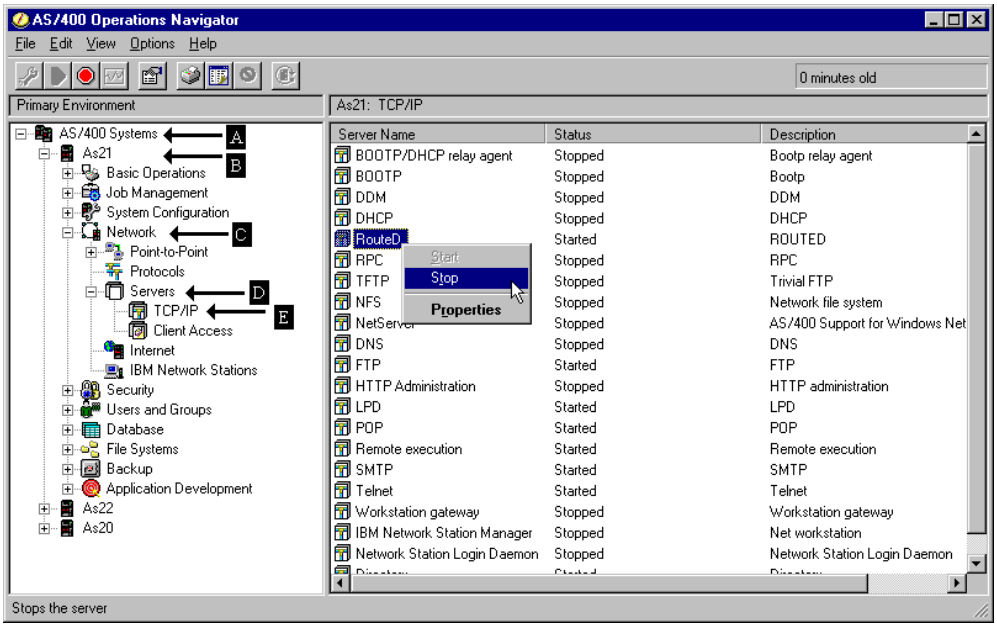


Figure 637. Ending the RouteD daemon using Operations Navigator

2. Double-click the AS/400 network (A).
3. Double-click the system icon (B) for the AS/400 system that you are configuring. The system components appear.
4. Double-click **Network** (C). The network components appear.
5. Double-click **Servers** (D). The available protocols appear.
6. Double-click **TCP/IP** (E). The available services appear in the right window.
7. Right-click on the **RouteD** item to see the menu. Select **Stop** as shown in Figure 637.

Using the green-screen interface

Use the End TCP/IP Server (ENDTCPSVR) command specifying the RouteD server:

```
ENDTCPSVR SERVER(*ROUTED)
```

14.4.3.3 Configuring the RouteD daemon

This section describes how to configure the RouteD daemon using the Operations Navigator interface.

Using Operations Navigator

1. Start Operations Navigator by clicking **Start->Programs->IBM AS400 Client Access->AS/400 Operations Navigator**. The AS/400 Operations Navigator window appears (Figure 638).

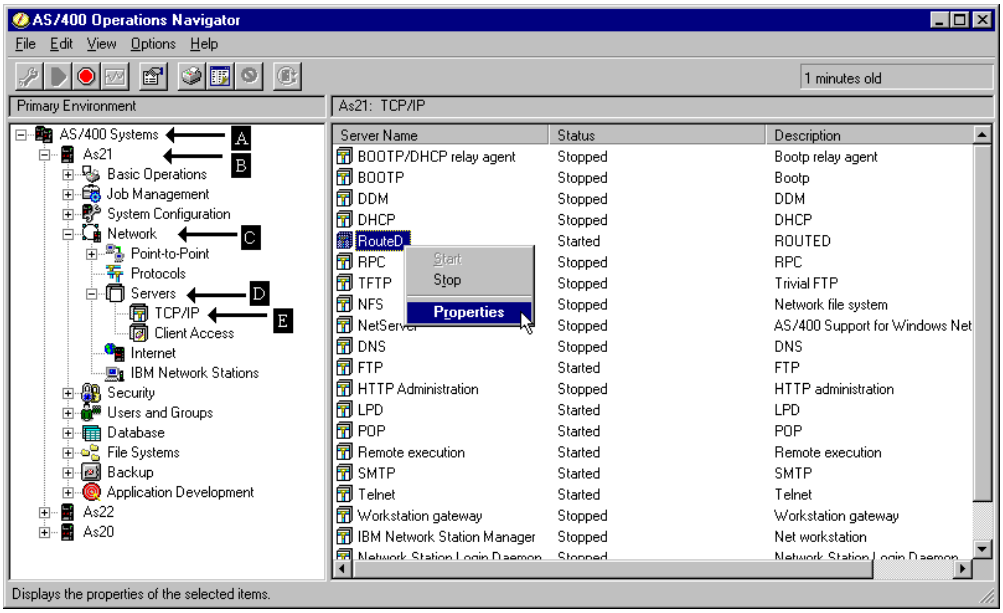


Figure 638. Configuring the RouteD daemon using Operations Navigator

2. Double-click on the AS/400 network (A).
3. Double-click the system icon (B) for the AS/400 system that you are configuring. The system components appear.
4. Double-click **Network** (C). The network components appear.
5. Double-click **Servers** (D). The available protocols appear.
6. Double-click **TCP/IP** (E). The available services appear in the right window.
7. Right-click on the **RouteD** item to see the menu. Select **Properties** as shown in Figure 638.
8. Specify if the RoutedD server automatically should start when TCP/IP is started. You can also specify if you want the RouteD daemon to act as an *active* or *passive* participant in the RIP network (Figure 639).

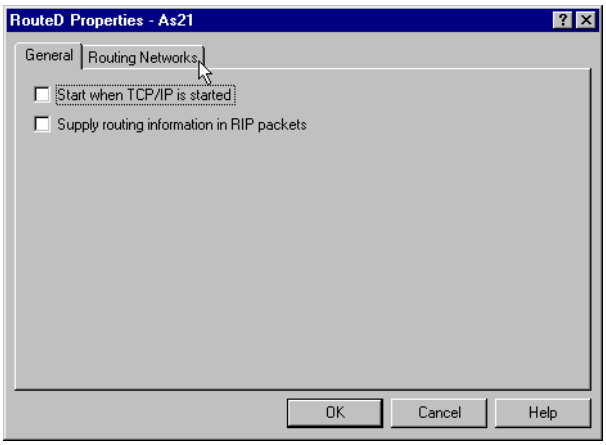


Figure 639. Configuring the General settings of the RouteD daemon

9. If you want to add new networks, select the **Routing Networks** tab, and click **New** (Figure 640 on page 534).

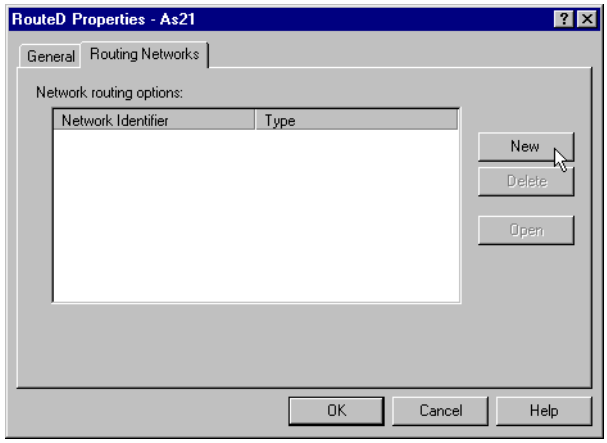


Figure 640. Configuring the Network settings of the RouteD daemon

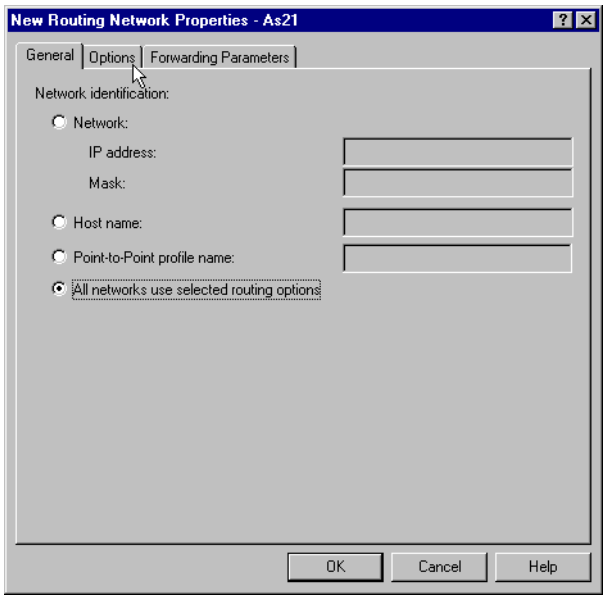


Figure 641. Adding a new network: General settings

10. In the **General** tab, you can specify how you want to identify networks for RouteD routing. You can identify them as a network, host name, or Point-to-Point Protocol (PPP) connection profile.

    - In the *network* option, specify that you want to identify the route as a network. If you click this, you must enter a network IP address and a network mask.
    - In the *host name* option, specify that you want to identify the route as a host by entering the host name.

- The *serial access profile name* option lets you specify that you want to identify the route as a PPP connection profile name. Enter the logical interface name that will be used to identify the PPP interface. An IP address will be dynamically assigned to this interface when the PPP connection becomes active.
- *All networks use selected routing options* specifies that you want all networks on the AS/400 system to use the selected routing options.

11. Select the **Options** tab (Figure 642 on page 536) to define routing information and route redistribution options for a specified peer gateway (this gateway can be reached directly).
    - The *Routing information* lets you specify the default routing protocol you want to use for the RouteD server. You can specify whether you want the RouteD server to send or receive Routing Information Protocol (RIP) traffic, as well as the version of RIP you want to use. The following options are possible:
      - **Passive (do not send or receive RIP traffic)**: This specifies that you do not want the RouteD server to send or receive any RIP traffic.
      - **Supply off (receive RIP1, RIP2; send no RIP traffic)**: This specifies that you want the RouteD server to receive all RIP traffics, but not to send any.
      - **Supply RIP1 (receive RIP1; send RIP1 traffic)**: This specifies that you want the RouteD server to receive all RIP traffics and sends only RIP version 1 (RIP1).
      - **Supply RIP2 (receive RIP2; send RIP2 traffic)**: This specifies that you want the RouteD server to receive all RIP2 traffics and send only RIP2. RIP2 are multicast. Any RIP2 packets received can contain an authentication record with the appropriate password based on the associated peer gateway.
    - *Route redistribution* lets you specify a route redistribution type for broadcasting routing tables. The basic reason for route redistribution types is to keep WANs from knowing about each other via the AS/400 system. For example, if your AS/400 system is connected to two WANs (WAN1 and WAN2) and one LAN, here is the general scenario the AS/400 system follows when the WANS are specified as *Limited* interfaces and the LAN is specified as a *Full* interface. The AS/400 system sends information about the routes it learned from WAN1 to the LAN. The AS/400 also sends information about the routes it learned from WAN2 to the LAN. The LAN, therefore, knows about both WAN1 and WAN2 routes. However, the AS/400 system does not send information about the other interfaces to the WAN. The result is that WAN1 knows about the LAN route, but not the WAN2 route. In the same way, WAN2 knows about the LAN route, but not the WAN1 route. For a LAN, the default is Full. For a WAN, the default is Limited. There may be reasons, however, why you may want to define a WAN interface as Full. For example, if you have a private network that is WAN-based, you may want to define your interfaces as Full to have the routing tables propagated so that WANs know about each other.
      - **Use default:** This specifies whether you want to use the default route redistribution types. Most of the time, you should select this option

because it is usually sufficient for your needs. For a LAN, the default is Full. For a WAN, the default is Limited.

  - **Limited:** When you select this option, the network route will not be sent to all routers. Typically, you choose this option for a WAN interface to control routing table propagation to the WAN.
  - **Full:** When you select this option, information about this route are sent to all routers. Typically, you choose this option for a LAN interface.

- **Metric (hop count):** This lets you select the route metric or a number associated with the cost of using this route. For example, a value of 1 indicates a route that is nearby, while a value of 15 indicates a route that is relatively far away. A value of 16 defines a route that is unreachable. This route metric number is added to routes received through the specified interface.
- **RIP community:** This specifies the community name for RIP2 authentication. Specifying a community name helps avoid problems with keeping IP address information accurate, if, for example, Point-to-Point Protocol (PPP) links are different every time a connection is made. This helps authenticate RIP2 responses with the peer gateway, if you are going to receive RIP2 on this interface. The RIP community name is the name of the community you expect to see from the peer gateway when RIP2 authentication is turned on.

  If you specify a community name, RIP2 authentication is indicated for this interface. The community name you specify must match the community name sent in all RIP2 message blocks for this interface. You can enter 1 to 16 characters.

  If you do not specify a RIP community, there is no authentication.

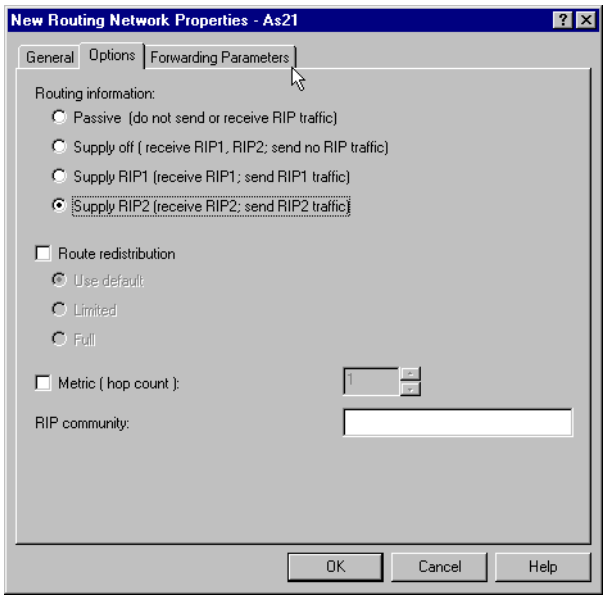


Figure 642. Adding a new network: Options

12. Select the **Forwarding Parameters** tab (Figure 643) to eliminate redistribution of network routes in the routing tables for the RouteD server. You can specify destination addresses for the blocking and forwarding of RIP traffic on the

interface. Additionally, you can specify a list of destination addresses that are not forwarded. You can also conditionally forward RIP traffic on the interface. You can change a particular IP address list using the Add and Remove buttons. To add an IP address to a list, click **Add**. To remove an IP address from a list, select it in the list, and click **Remove**. The possible parameters are:

- **Block network addresses**: Lists the network IP addresses that are blocked on the interface. The IP addresses you specify here indicate that you want the RouteD server to ignore these network addresses. The RouteD server does not include these addresses in its routing table. As a result, the addresses are not forwarded to other routers.
- **Forward network addresses**: Lists the network IP addresses that are forwarded only through the interface IP address you specify on the General tag page. If that interface address is not available, the RouteD server does not forward the route to another interface address.
- **Forward condition network addresses**: Lists the network IP addresses that are forwarded conditionally on the interface. If the gateway IP address you specify on the General page is down, the RouteD server forwards the route to another gateway address, based on the order you listed them on the Routing Peers page for RouteD Properties. The first address the RouteD server encounters that allows the route to be sent over the interface ends the RouteD server's processing of that route.
- **Do not forward network addresses**: Lists the network IP addresses you want the AS/400 system to know about and use, but does not forward them to other routers. Typically, these addresses are your company's internal addresses that you do not want redistributed beyond your company. The RouteD server does not forward these addresses to other routers.

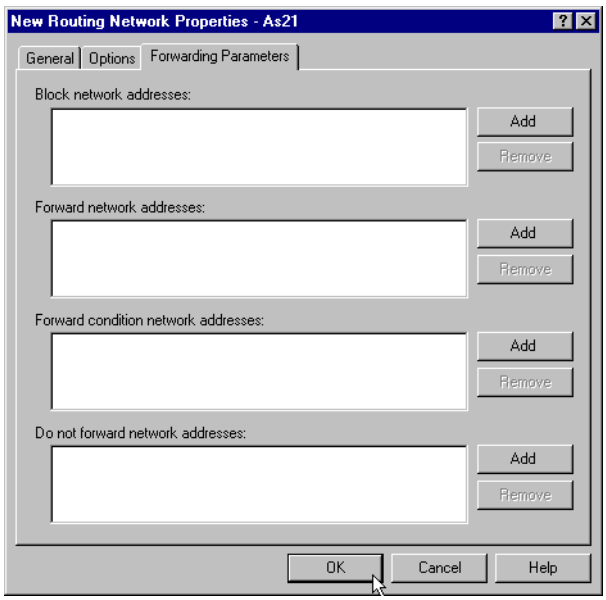


Figure 643. Adding a new network: Forwarding parameters

Using the green-screen interface

The RouteD server is configured using the Configure TCP/IP RouteD (CFGTCPRTD) command. This command displays the menu shown in Figure

644. *IOSYSCFG special authority is required to make changes to the RouteD attributes with the Change Routing RouteD Attributes (CHGRTDA) command.

```
                          Configure TCP/IP RouteD
                                                              System:   AS21
Select one of the following:

     1. Change RouteD attributes
     2. Work with RouteD configuration









Selection or command
===>

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
```

Figure 644. Using the Configure TCP/IP RouteD (CFGTCPRTD) command

Selecting option `1` enables the configuration of the general RouteD settings (Figure 645).

```
                      Change RouteD Attributes (CHGRTDA)

Type choices, press Enter.

Autostart  . . . . . .  . .   *NO          *SAME, *YES, *NO
Supply . . . . . . . . . . .  *NO          *SAME, *YES, *NO
```

Figure 645. Configuring the general RouteD settings

Two parameters are available to change the attributes.

- **Autostart servers parameter (AUTOSTART)**: Specifies whether to automatically start the RouteD server when TCP/IP is started by the STRTCP command. When RouteD is started by the STRTCPSVR command, the AUTOSTART parameter is ignored and the RouteD server is started, regardless of the value of this parameter. If STRTCPSVR SERVER (*ROUTED) is specified, and the RouteD server is already running, the start request will be denied.
- **Supply parameter (SUPPLY)**: Specifies whether RouteD should supply routing information in RIP packets over the network interfaces.
  - ***YES** means that the RouteD task supplies periodic RIP update packets to the attached networks. The supply of RIP packets over a particular interface may be overridden by an entry in the configuration file specifying that the supply over a particular interface is to be turned off.

– ***NO** means that the RouteD task receives and processes RIP packets normally, but does not supply periodic RIP update packets over any of the attached network interfaces. This effectively puts the RouteD server into *listen mode*.

Working with configuration V4R1

Use the Work with RouteD Configuration (WRKRTDCFG) command shown in Figure 646 to change the RouteD server configuration in the RouteD configuration file QATOCRTDC in QUSRSYS.

```
                      Work with RouteD Configuration
                                                          System:   AS21
Type options, press Enter.
  1=Add   2=Change   3=Copy   4=Remove   5=Display   13=Insert

      Sequence
Opt    Number    Entry

        00010    # * * * * * * * * * * * * * * * * * * * * * * * * * * * *   >
        00020    # RTD DEFAULT CONFIGURATION                                  >
        00030    # * * * * * * * * * * * * * * * * * * * * * * * * * * * *   >
        00040    #                                                           >
        00050    # RouteD Interface Definitions
        00060    # ---------------------------
        00070    # TCP/IP will learn about a route to network 10.0.0.0 th   >
        00080    # means external to RouteD, therefore do not allow Rout   >
        00090    # route to this network.
```

Figure 646. The RouteD configuration files from the green-screen interface

The command shows a list of RIP configuration entries. From this list, you can add, change, copy, remove, or display the individual entries. It is also possible to print the list. You do not have to add any entries in this file, but there may be some occasions where you want to put specific definitions on routes and networks. The entries in the file can supply additional routing information to the RouteD server.

*IOSYSCFG special authority is required to make changes to the RouteD configuration with the WRKRTDCFG command. We can find examples in member CONFIG in the file QATOCRTDC in library QUSRSYS.

In V4R1 of OS/400, there are three types of entries to the configuration file:

- # - comment
- net/host
- options

The net/host entry specifies how a remote gateway/host works in a network. The options entry specifies how a local interface works with networks. Each entry has to follow a particular syntax.

It is possible to specify gateway options and interface options. You can specify a gateway/host as being passive, external, or static:

Passive A passive gateway does not exchange routing information. Information about the passive gateway is maintained in the local routing tables indefinitely and is not advertised to the network, so that this gateway is

kept secret to other routers. Do not be confused with the general term passive in the RIP protocol, which can listen, but not speak.

External An external gateway parameter indicates that entries for this destination should never be added to the IP routing table via the RouteD server process. Nobody knows the existence of this gateway, including the local (or this) router.

Static A static gateway is similar to a passive gateway, with the exception that the information is included in the RIP packets that are broadcast to the neighboring routers. This entry is not updated even though a change has occurred to this gateway, unless the RouteD server is restarted.

Some of examples in the file QATOCRTDC in QUSRSYS are:

```
########################################################################
# Router 192.168.2.15 is used to reach host 7.1.2.3, but 192.168.2.15
# does not have a RouteD server, therefore the route is PASSIVE.
# host 7.1.2.3 mask 255.255.255.255 gateway 192.168.2.15 metric 1 passive
# No RIP packets will be broadcasted OR received over
# our network interface with address 192.168.150.1
# options interface 192.168.150.1 passive
########################################################################
```

Working with configuration V4R2 and later

An existing RouteD configuration file QUSRSYS/QATOCRTDC is automatically migrated at installation time of OS/400 V4R2. You can use the Work with RouteD Configuration (WRKRTDCFG) command shown in Figure 646 on page 539 to change the RouteD server configuration in the RouteD configuration file QATOCRTDC in QUSRSYS. We recommend that you use the Operations Navigator interface. Refer to 14.4.3.3, “Configuring the RouteD daemon” on page 532, for details.

In V4R2, the RIP_INTERFACE statement replaces OPTIONS INTERFACE from V4R1. The RIP_INTERFACE statement is used to specify all routing options on a per interface basis. The RIP_INTERFACE statement now contains the functionality for defining routes and creating static routes. Prior to V4R2, these functions existed in the NET and HOST statements. For V4R2, the HOST and NET statements are no longer valid on the WRKRTDCFG displays. This information is now being specified on the ADDTCPRTE and CHGTCPRTE commands.

It is possible to specify multiple interface options on a single entry in the configuration file. Possible options are: BLOCK, FORWARD, FORWARD.COND, and NOFORWARD. However, we recommend that you use multiple lines to specify multiple options for a given interface.

Interfaces on the AS/400 system can be specified as follows:

Network

Specified as an IP address and a mask or an IP address and a bit number. This bit number specifies which bit in the 0-n bits of the IP address (counting left to right) is the last bit of the IP address' network portion. If both the mask and bit number are missing, a network is calculated using the subnet mask of the specified interface (via the ADDTCPIFC command).

Interface name

This is the logical interface name used to identify a PPP interface which will have a dynamically assigned IP address at the time the PPP connection becomes active.

Hostname

This is the hostname of the AS/400 system.

*****

Refers to all interfaces on the AS/400 system. This is useful for setting defaults for all interfaces. These can be overridden by using the RIP_INTERFACE statement with different values for selected parameters.

In V4R2, SUPPLY RIP1, as well as the new SUPPLY RIP2, are possible. They indicate which version of the RIP protocol the system is using to send and receive routing information to and from neighboring routers. For SUPPLY RIP1, the system processes only RIP1 packets. For SUPPLY RIP2, the multicast address 224.0.0.9 is used to supply only RIP2 packets (see RFC1723).

DIST_ROUTES_IN controls how RouteD redistributes routes that are received from this RIP_INTERFACE network to WANs. The redistribution of routes to LANs are not affected by this.

- If LIMITED is specified, this means that routes received from the RIP_INTERFACE network will not be redistributed to other LIMITED interfaces. This parameter is for WAN only.
- The METRIC parameter specifies the value the system will add to routes received through the specified interface (1 through 15 is the valid range; 16 is infinity).

The COMMUNITY parameter specifies the community name used by this interface for authentication purposes (RFC1723). The rip_community_name (1 through 16 characters string) is valid for interfaces with a SUPPLY RIP2 value. The community name that is specified with the community option should match the community name sent in all RIP2 message blocks for this interface. If you do not specify the community option, authentication is not indicated for that interface.

Some of the examples in the file QATOCRTDC in QUSRSYS for V4R2 OS/400 are shown in Figure 647 on page 542.

```
# * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * #
# RID DEFAULT CONFIGURATION                                                   #
# * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * #
# RouteD Interface Definitions
# ----------------------------
# TCP/IP will learn about a route to network 10.0.0.0 through some
# means external to RouteD, therefore do not allow RouteD to accept a
# route to this network.
#
# RIP_INTERFACE * SUPPLY RIP1 METRIC 1 BLOCK 10.0.0.0 MASK 255.0.0.0
#
# The following is an example of a CL command, and should not be
# included in the RouteD configuration file:  Router 192.168.2.15 is
# used to reach host 7.1.2.3, but 192.168.2.15 does not have a RouteD
# server.  Do not redistribute this route to other routers.  The CL
# command is used, because host 7.1.2.3 is a static route.
#
#   ------ example CL command -------------------------------------
#  !                                                               !
#  !  ADDTCPRTE RTEDEST('7.1.2.3') SUBNETMASK('255.255.255.255')   !
#  !    NEXTHOP('192.168.2.15') METRIC(1) REDST(*NO)               !
#  !                                                               !
#   ---------------------------------------------------------------
#
# The following example is also an example of a CL command and should
# not be included in the RouteD configuration file:  We know of a
# route to network 11.0.0.0 through router 192.168.2.15 and we would
# like to include this information in our broadcasts to all other
# network interfaces.
#
#   ------ example CL command -------------------------------------
#  !                                                               !
#  !  ADDTCPRTE RTEDEST('11.0.0.0') SUBNETMASK('255.0.0.0')        !
#  !    NEXTHOP('192.168.2.15') METRIC(3) REDST(*YES)              !
#  !                                                               !
#   ---------------------------------------------------------------
#
# Do not allow RouteD to accept routes to network 10.0.0.0 through
# our network interface with address 192.168.2.2
# In addition, this is a very busy interface, so
# increment the metric to impose a higher cost of using it.
#
# RIP_INTERFACE 192.168.2.2 SUPPLY RIP2 METRIC 2 BLOCK 10.0.0.0
#
# Through network interface 192.168.5.4, we wish to
# listen to RIP packets, but we will not transmit any.
#
# RIP_INTERFACE 192.168.5.4 SUPPLY OFF
#
# No RIP packets will be broadcasted OR received over
# our network interface with address 192.168.150.1
#
# RIP_INTERFACE 192.168.150.1 PASSIVE
#
# Through point to point interface with name FOO, we wish to send
# RIP2 packets with authentication.
#
# RIP_INTERFACE FOO SUPPLY RIP2 COMMUNITY MY_COMMUNITY
#
# Through point to point interface with name FOO, we wish to send
# RIP2 packets without authentication.
#
# RIP_INTERFACE FOO SUPPLY RIP2
#
# Distribute all routes learned over a WAN containing host
# MYHOST.MYCOMPANY.COM to all other interfaces.
#
# RIP_INTERFACE MYHOST.MYCOMPANY.COM SUPPLY RIP2 DIST_ROUTES_IN FULL
#
# Do not allow RouteD to accept routes to private networks through
# network interface 1.2.0.0 (this example uses a bit number
# representation for the mask)
#
# RIP_INTERFACE 1.2.3.4 15 SUPPLY RIP1 BLOCK PRIVATE
#
# * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * #
```

Figure 647. Sample QUSRSYS/QATOCRTDC RIP control file

RIP support with PPP

Both RIP v1 and RIP v2 are supported on PPP links on an AS/400 system running V4R2 of OS/400. Refer to Chapter 4, “Configuring PPP and SLIP” on page 81, for more detailed information about PPP setup through the Operations Navigator.

On the TCP/IP settings page of the Point-to-Point profile properties dialog, it is possible to click Routing. This specifies whether Routing Information Protocol (RIP) traffic can be received or generated with this connection profile.

You have the following dynamic routing options:

None This specifies that no RIP traffic can be received or generated on this interface. Select None, if you want to use static routes.

RIP1 This specifies that Version 1 (RIP1) can be received and sent on this interface.

RIP2 This specifies that RIP2 can be received and sent on this interface.

When Dynamic routing is selected, you have the option of either full or limited route redistribution.

14.4.4 Sample configurations

Figure 648 shows how the RouteD configuration entries work in a sample network. The routers would know every routes in the all networks, including NetworkX, NetworkY, NetworkZ, NetworkA, and NetworkW.

14.4.4.1 Case 1 through case 3

The following sections show sample configurations on the Operations Navigator, which are equivalent to the RIP_INTERFACE statements. With the configurations, you can compare the configurations through the different user interfaces.

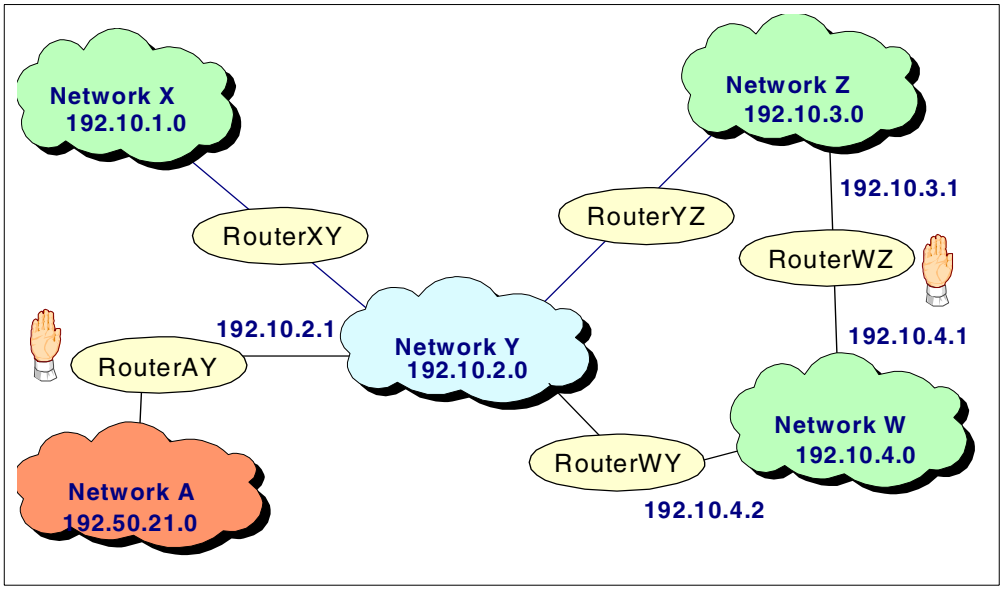


Figure 648. Sample network

Case 1

For case 1, refer to Figure 649. If the RouterAY has the following entry, none of the hosts in the networks can reach NetworkA:

```
RIP_INTERFACE 192.50.2.1 supply rip1 metric 1 noforward 192.50.21.0
```

The knowledge of that network is not advertised out.

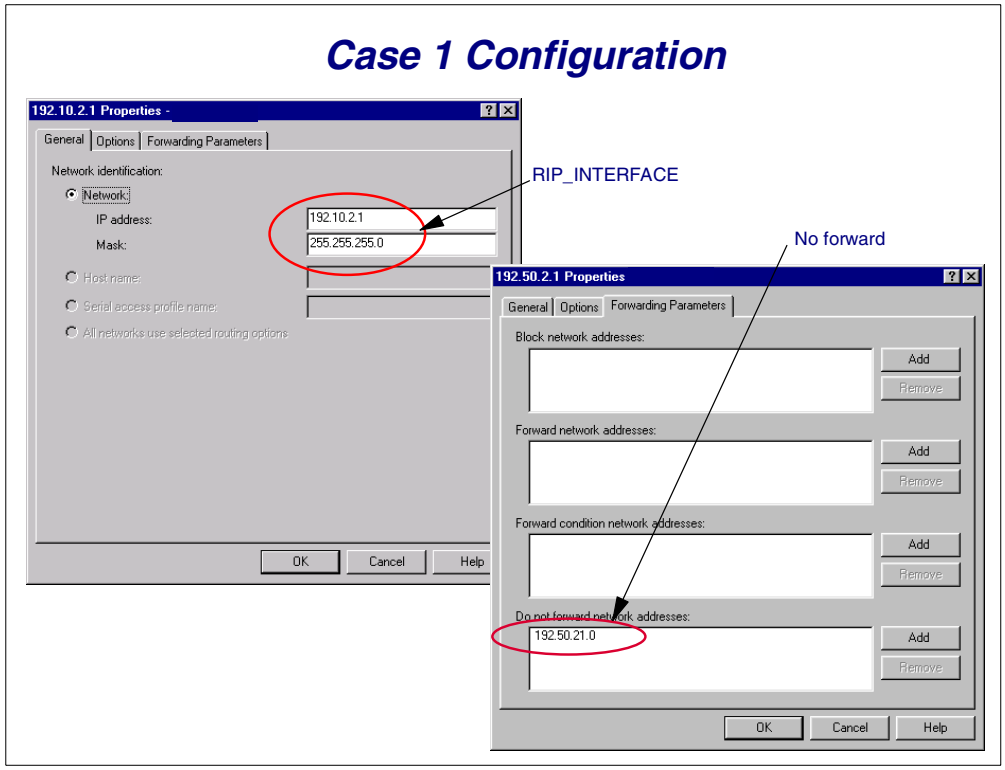


Figure 649. Blocking the192.50.21.0 network

Case 2

For case 2, refer to Figure 650. If the RouterWZ has the following entry, none of the IP packets will go from NetworkZ through the RouterWZ to the NetworkW:

```
RIP_INTERFACE 192.10.3.1 supply rip1 metric 1 noforward 192.10.4.0
```

The knowledge of that network is not advertised out. NetworkW is still reachable, because the RouterWY may advertise the route to that network.

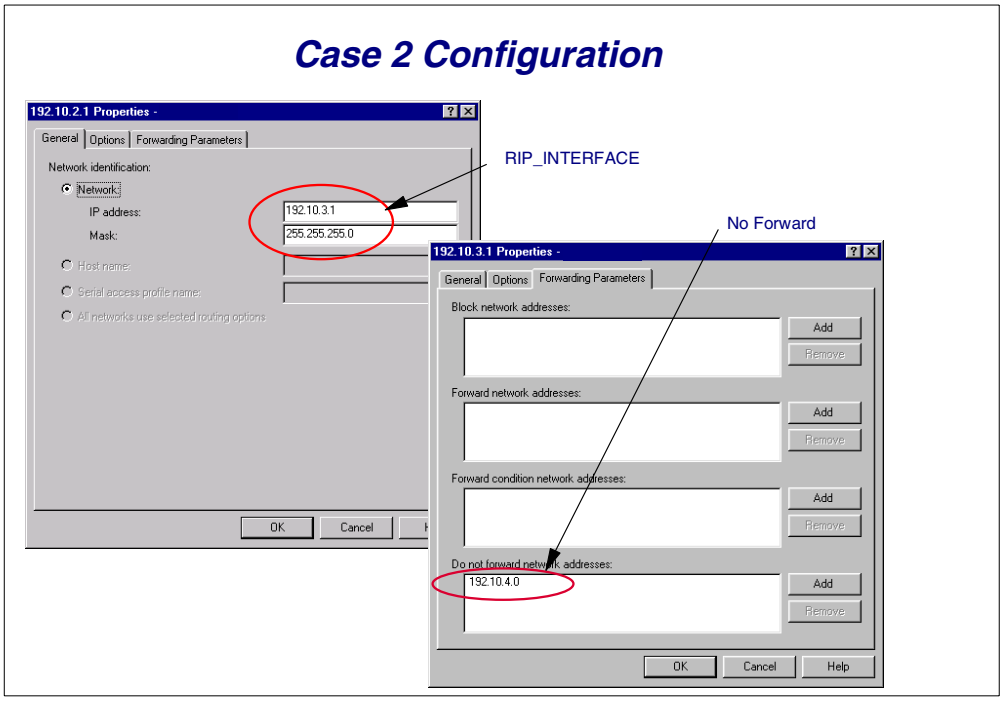


Figure 650. Blocking the 192.10.4.0 network

Case 3

For case 3, refer to Figure 651 on page 546. If the RouterAY has the following entry, none of the routers know about the route going through the RouterAY to reach the NetworkA:

```
RIP_INTERFACE 192.50.21.1 passive
```

No IP packets should be routed to this RouterAY. This does not prevent IP packets routed to this router, because a host in the NetworkY may have an entry to point this router as the next hop address:

```
ADDTCPRTE RTEDEST('192.50.21.5') SUBNETMASK('255.255.255.255') +
NEXTHOP('192.10.2.1')
```

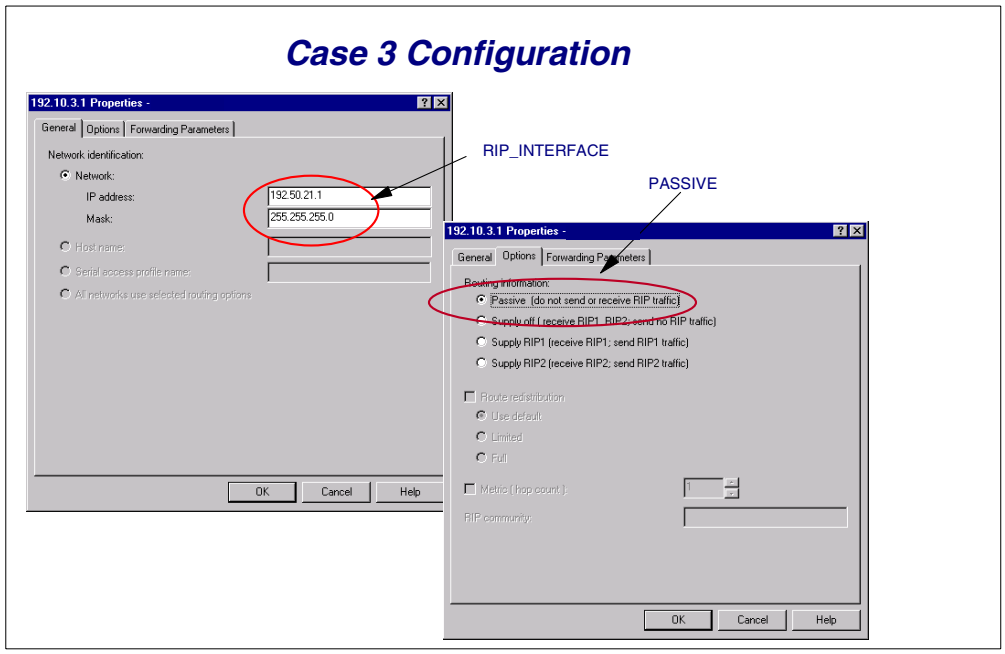


Figure 651. Blocking the 192.50.21.0 network

14.4.5 Routing information

The current routing table on the AS/400 system can be shown by the use of the Work with TCP/IP Network Sts (NETSTAT) command. From the command entry, enter:

```
NETSTAT OPTION(*CNN)
```

Figure 652 and Figure 653, show the routing table. Notice the *Route Source* column, showing the origin of the entry, *RIP (learned from the RouteD daemon) or *CFG (static routing entry added by the system administrator).

```
                    Display TCP/IP Route Information
                                                    System:   SYSTEM01
Type options, press Enter.
  5=Display details

     Route            Type of        Route  Route      Route
Opt  Destination      Service         MTU   Type       Source
     8.7.24.0         *NORMAL         1989  *SUBNET    *RIP
     8.7.23.0         *NORMAL         1989  *SUBNET    *RIP
     8.7.22.0         *NORMAL         1989  *SUBNET    *RIP
     8.7.21.0         *NORMAL         1989  *SUBNET    *RIP
     8.7.20.0         *NORMAL         1989  *SUBNET    *RIP
     8.7.19.0         *NORMAL         1989  *SUBNET    *RIP
     8.7.18.0         *NORMAL         1989  *SUBNET    *RIP
     8.7.17.0         *NORMAL         1989  *SUBNET    *RIP
     8.7.16.0         *NORMAL         1989  *SUBNET    *RIP
     8.7.15.0         *NORMAL         1989  *SUBNET    *RIP
     8.7.14.0         *NORMAL         1989  *SUBNET    *RIP
     8.7.11.0         *NORMAL         1989  *SUBNET    *RIP
     8.7.10.0         *NORMAL         1989  *SUBNET    *RIP
```

Figure 652. TCP/IP routing table: NETSTAT Command (Part 1)

```
                    Display TCP/IP Route Information
                                                      System:   SYSTEM01
Type options, press Enter.
  5=Display details

     Route            Type of         Route  Route      Route
Opt  Destination      Service          MTU   Type       Source
     8.7.2.0          *NORMAL          1989  *SUBNET    *RIP
     8.7.1.0          *NORMAL          1989  *SUBNET    *RIP
     127.0.0.0        *NORMAL           576  *DIRECT    *CFG
     192.168.3.0      *NORMAL         16388  *DIRECT    *CFG
     *DFTROUTE        *NORMAL           576  *DFTROUTE  *CFG
```

Figure 653. TCP/IP Routing Table: NETSTAT Command (Part 2)

14.5 The future of routing on the AS/400 system

As of OS/400 V4R4, the only dynamic routing protocol available is RIP. As briefly described in 14.2.2, “Dynamic routing” on page 521, the OSPF protocol is a better choice of dynamic routing protocol if dealing with larger networks. Currently, it is unknown if the AS/400 system will support OSPF in forthcoming releases of the operating system.

When dealing with larger networks and requiring OSPF as routing protocol, the use of routers is advised. These routers are able to run all well-known routing protocols, including RIP and OSPF. Running RIP on the AS/400 system and both RIP and OSPF on the routers will enable the AS/400 system to participate in the complete network, since the routers will propagate routes from the OSPF protocol to the RIP protocol and hereby advertise the routes discovered by OSPF to the AS/400 system.

14.6 Scenarios

This section shows several scenarios of routing in relation to the AS/400 system. Both static and dynamic routing scenarios are shown.

The diagrams showing the scenarios uses the CIDR method of specifying a network mask. For example, the network 10.1.1.0 with subnet mask 255.255.255.0 can also be specified as network 10.1.1.0/24. The 24 identifies the number of significant bits in the IP address to use as the network. The 255.255.255.0 subnet mask identifies that the three first bytes of the mask should be used. Since each byte consists of 8 bits, this equals the 24 bits used in the CIDR method.

14.6.1 Static routing

The following three scenarios all show the use of static routing on the AS/400 system. Examples of using the AS/400 system and other platforms are shown.

14.6.1.1 Scenario 1: AS/400 as a router between LANs

Figure 654 shows the use of the AS21 AS/400 system as a router between two networks. This scenario is typical in a situation where a site has both an Ethernet LAN and a Token-Ring LAN. By using the AS21 AS/400 system as a router

between the two types of LAN, the hosts on each side can access resources on the other LAN.

On the Ethernet LAN, the AS/400 system AS22 can connect to the Windows NT system NTPL on the Token-Ring by using AS21 as an intermediate system.

AS21 acts as a router when IP datagram forwarding is enabled on AS21. This way, data packets received on one interface can be routed to other interfaces, provided an active interfaces exists. If the datagram forwarding is *not* enabled, no routing takes place.

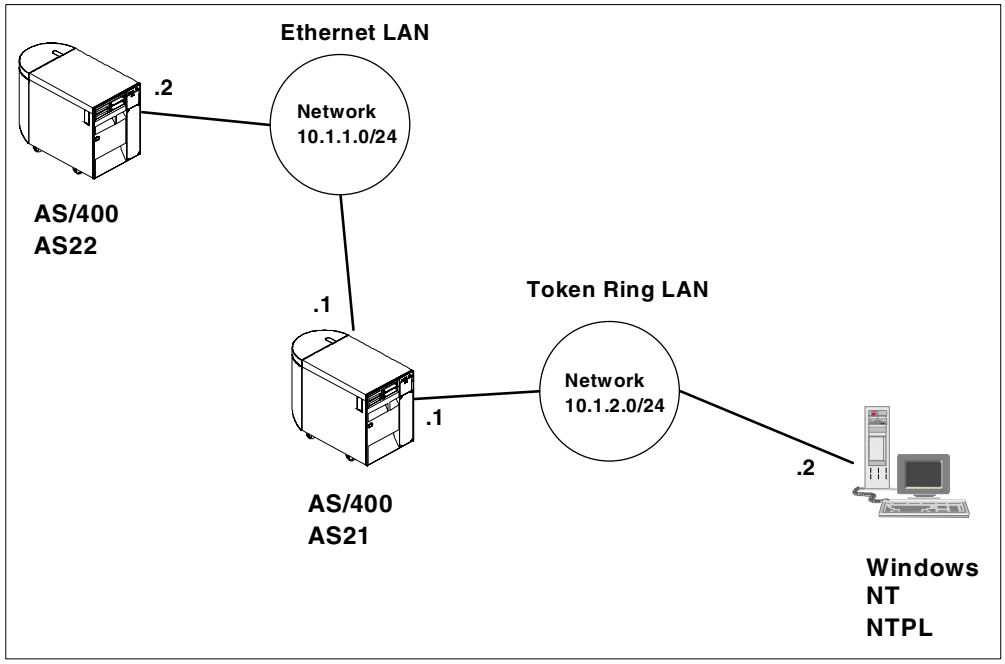


Figure 654. Scenario 1: AS/400 AS21 as a router between networks

14.6.1.2 Scenario 2: AS/400 system at a central site and a remote site

Figure 655 shows the use of the AS21 AS/400 system located at a central site, servicing a remote location. The router hides the possible existence of a WAN connecting the two networks. This scenario does not include the WAN connection, but this is irrelevant to the configuration of routing on the two AS/400 systems. They only know the existence of the router. It is the job of the router to connect the two networks.

The scenario is also useful when creating networks that consist of different network types, such as Ethernet and Token-Ring. The most elegant solution is to use a router to connect the dislike network types.

AS21 will be using the default-route feature, since only one remote location exists. AS22 will be using a default-route feature as well.

The router acts as the intermediate system between the two networks, the Token-Ring and the Ethernet.

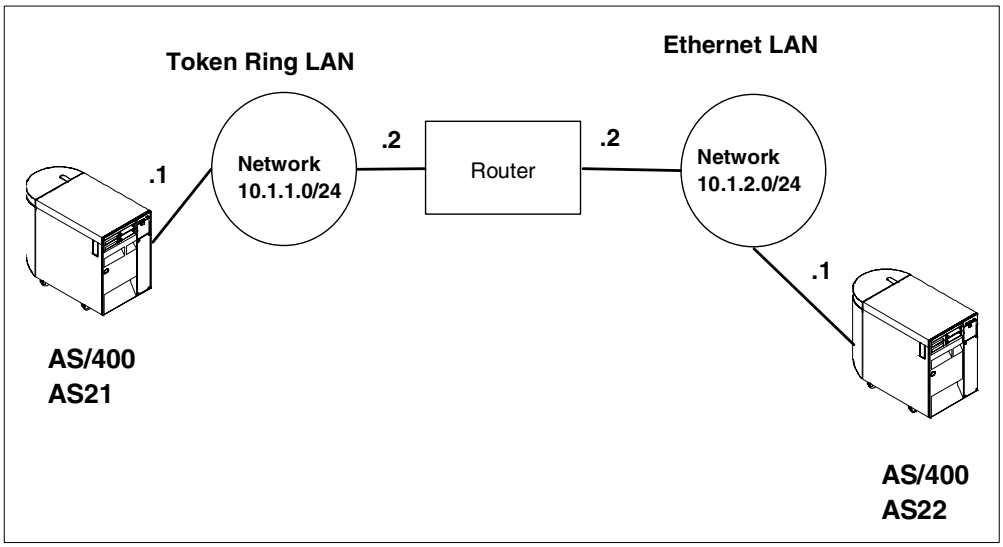


Figure 655. Scenario 2: Central site with a remote network

14.6.1.3 Scenario 3: AS/400 at central site and several remote sites

Figure 656 shows the use of the AS21 AS/400 system located at a central site servicing two remote locations. This scenario is typical in a situation where a company has the DP department located centrally and systems located at other locations.

The AS/400 system AS21 will use explicit routing entries, since two different routers will be used as gateways, depending on what is to be accessed. A default route is not necessary.

The AS/400 systems AS22 and AS23 are using default-routing, since the only partner is the centrally located AS21.

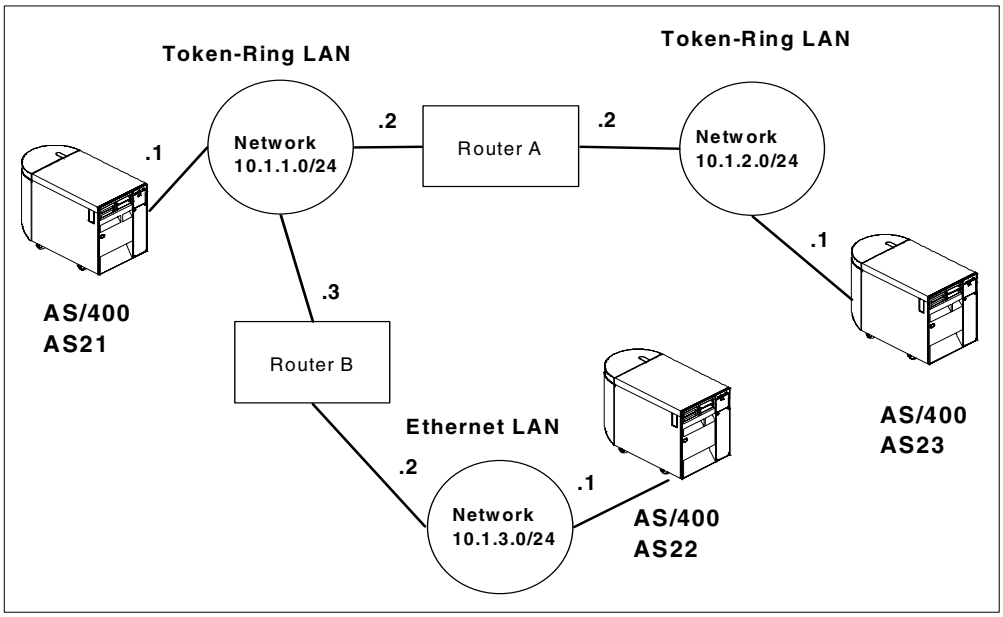


Figure 656. Scenario 3: Central site with multiple remote networks

14.6.2 Dynamic routing

The following two scenarios show the use of dynamic routing on the AS/400 system. The examples also show the AS/400 system using RIP.

14.6.2.1 Scenario 4: AS/400 using RIP to build a complete network map

Figure 657 shows the use of the AS/400 system AS21 and AS22 using RIP. The two AS/400 systems receive RIP messages and hereby automatically learn the complete network topology within a few minutes. The AS/400 system AS22 initially does not know the existence of the 10.1.1.0/24 network, but learns this from the RIP messages received from the AS/400 system AS21 via the 10.1.2.0/24 network.

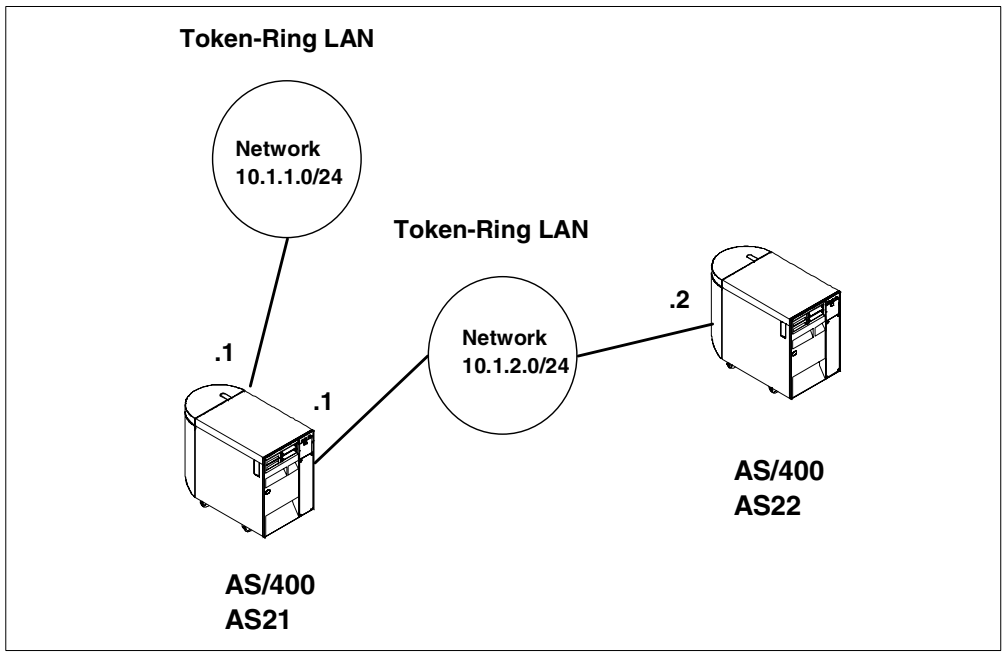


Figure 657. Scenario 4: Using Dynamic Routing (RIP) to connect several networks

14.6.2.2 Scenario 5: AS/400 using (RIP) to hide part of a network

Figure 658 shows the use of the two AS/400 systems AS21 and AS22 using RIP. The two AS/400 systems receive RIP messages and, therefore, automatically learn the complete network topology within few minutes.

Note that the complete network topology is exchanged, since the 10.1.3.0/24 network is hidden by AS22 by using the RIP NOFORWARD function.

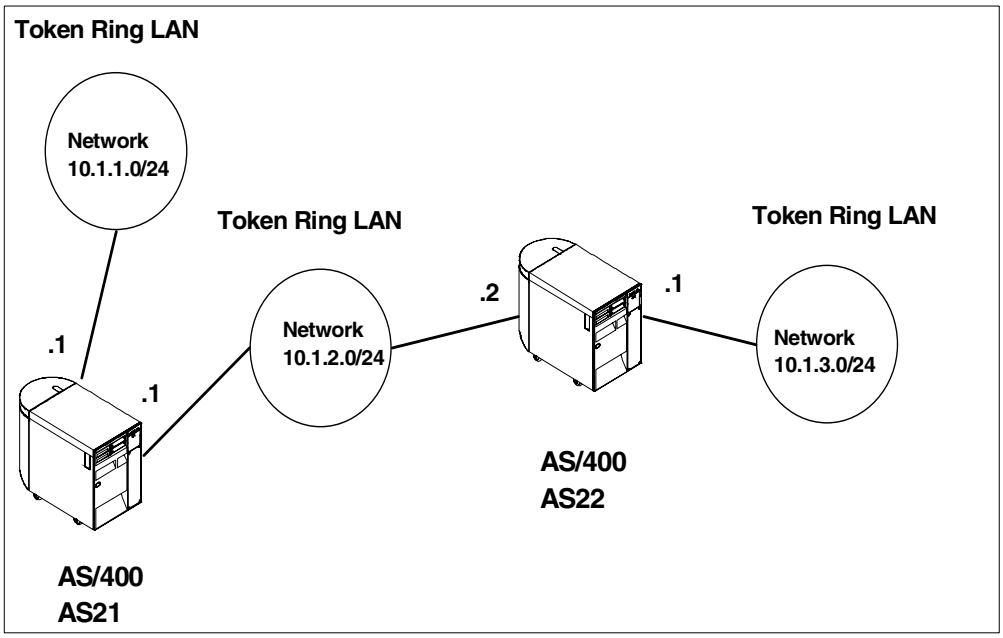


Figure 658. Scenario 5: Using Dynamic Routing (RIP) to omit a network

14.7 Configuring the scenarios

This section details the configuration of the scenarios described in 14.6, “Scenarios” on page 547. Each scenario is documented through a task overview, the actual configuration, and the testing of the scenario.

14.7.1 The AS/400 system as a router between LANs

The scenario shows the use of the AS/400 system acting as an intermediate system between dislike LANs.

14.7.1.1 Task overview

The task includes these actions:

1. Connect the two AS/400 systems and the Windows NT system to the networks. The AS21 system should be connected to both the Ethernet and the Token-Ring networks and the AS22 system to the Ethernet network only. The Windows NT NTPL system should be connected to the Token-Ring network only.
2. Create and start the IP interfaces on the two AS/400 systems AS21 and AS22.
3. Configure the AS22 system to use the AS21 system as a default router.
4. Configure the AS21 system to allow forwarding of IP datagrams across its IP interfaces.
5. Configure the basic TCP/IP setup on the Windows NT NTPL system. Use the AS/400 AS21 system as a default router.
6. Test the connection by using the FTP function from the NTPL system to start an FTP session on the AS22 system.

14.7.1.2 Configuring the scenario

Refer to Chapter 2, “TCP/IP basic installation and configuration” on page 7, to create the basic setup of the IP interfaces on the two AS/400 systems. Use the following IP addresses on the AS/400 systems:

- AS21: 10.1.1.1/24 on the 10.1.1.0/24 network (Ethernet network)
- AS21: 10.1.2.1/24 on the 10.1.2.0/24 network (Token-Ring network)
- AS22: 10.1.1.2/24 on the 10.1.1.0/24 network (Ethernet network)

1. Configure the default route on the AS22 system using the Configure TCP/IP (CFGTCP) command:

   ```
   CFGTCP
   ```

   a. Enter option `2` on the menu (Figure 659).

```
CFGTCP                          Configure TCP/IP
                                                          System:   AS22
Select one of the following:

     1. Work with TCP/IP interfaces
     2. Work with TCP/IP routes
     3. Change TCP/IP attributes
     4. Work with TCP/IP port restrictions
     5. Work with TCP/IP remote system information

    10. Work with TCP/IP host table entries
    11. Merge TCP/IP host table
    12. Change TCP/IP domain information

    20. Configure TCP/IP applications
    21. Configure related tables
    22. Configure point-to-point TCP/IP


Selection or command
===> 2

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
```

Figure 659. Configuring the default route on the AS22 system

   b. Add the `*DFTROUTE` routing entry to the routing table (Figure 660).

```
                          Work with TCP/IP Routes
                                                                 System:   AS22
Type options, press Enter.
  1=Add   2=Change   4=Remove   5=Display

     Route            Subnet           Next             Preferred
Opt  Destination      Mask             Hop              Interface
 1   *DFTROUTE        *NONE            10.1.1.1




                                                                        Bottom
F3=Exit      F5=Refresh   F6=Print list   F11=Display type of service
F12=Cancel   F17=Top      F18=Bottom
```

*Figure 660. Adding the *DFTROUTE routing entry to the routing table on AS22*

c. Use the default values supplied by the command (Figure 661).

```
                        Add TCP/IP Route (ADDTCPRTE)

Type choices, press Enter.

Route destination  . . . . . > *DFTROUTE
Subnet mask  . . . . . > *NONE
Type of service  . . . . . *NORMAL       *MINDELAY, *MAXTHRPUT...
Next hop . . . . . > '10.1.1.1'
Preferred binding interface  . .  *NONE
Maximum transmission unit  . . .  576          576-16388, *IFC
Route metric . . . . . .  1            1-16
Route redistribution . . . . . .  *NO          *NO, *YES
Duplicate route priority . . . .  5            1-10









                                                                        Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 661. Selecting the default values for the routing entry

The complete routing table is shown in Figure 662 on page 554.

```
                              Work with TCP/IP Routes
                                                                  System:   AS22
Type options, press Enter.
  1=Add   2=Change   4=Remove   5=Display

     Route            Subnet           Next             Preferred
Opt  Destination      Mask             Hop              Interface

     *DFTROUTE        *NONE            10.1.1.1         *NONE

                                                                        Bottom
F3=Exit      F5=Refresh   F6=Print list   F11=Display type of service
F12=Cancel   F17=Top      F18=Bottom
TCP/IP route added successfully.
```

Figure 662. Routing table on the AS22 system

2. Configure the IP datagram forwarding on the AS21system. This can be done using either the Operations Navigator interface or the green-screen interface.

   If you want to use Operations Navigator, use the following procedure:

   a. Start Operations Navigator by clicking **Start->Programs->IBM AS400 Client Access->AS/400 Operations Navigator**. The AS/400 Operations Navigator window appears (Figure 663).

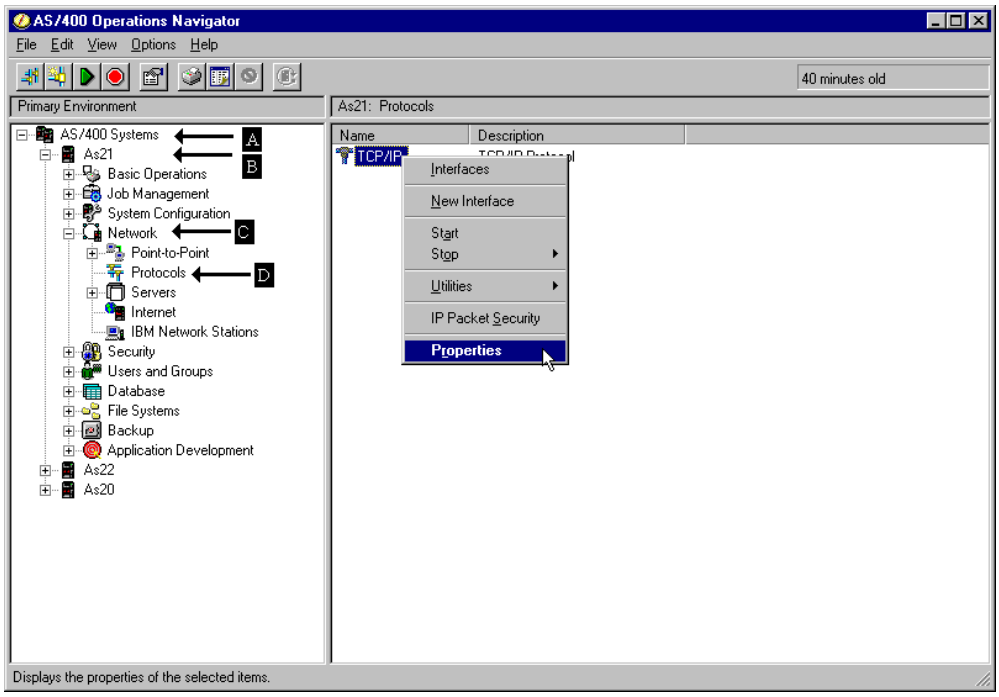


Figure 663. Configuring the TCP/IP settings on AS21 system

b. Double-click **AS/400 Network** (A).

c. Double-click the system icon (B) for the AS21 AS/400 system that you are configuring. The system components appear.

d. Double-click **Network** (C). The network components appear.

e. Double-click **Protocols** (D). The available protocols appear.

f. Place the mouse on the **TCP/IP** item in the right window, right-click, and select the **Properties** as shown in Figure 663.

g. Select the **Settings** tab. Make sure that there is a check in the IP datagram forwarding field. Click **OK** to confirm the setting (Figure 664).

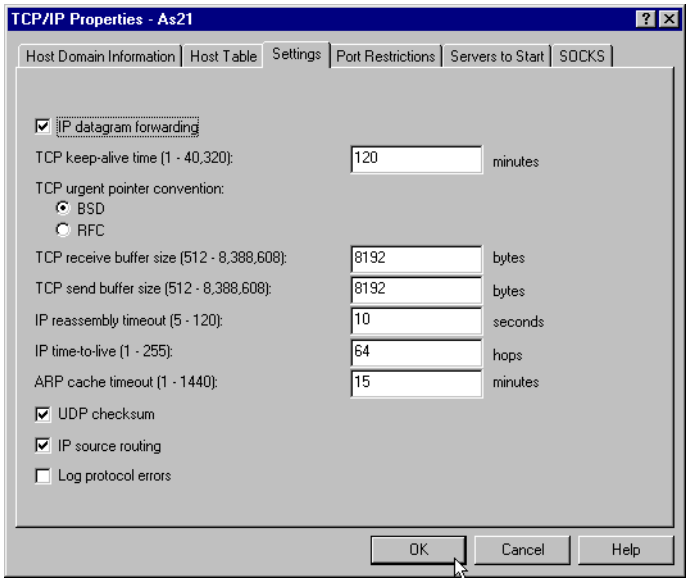


Figure 664. Enabling the IP datagram forwarding on the AS21 system

3. If you want to use the green-screen interface, use the following procedure:

   a. Use the Configure TCP/IP (CFGTCP) command to get the menu shown in Figure 665 on page 556.

```
CFGTCP                          Configure TCP/IP
                                                          System:   AS21
Select one of the following:

     1. Work with TCP/IP interfaces
     2. Work with TCP/IP routes
     3. Change TCP/IP attributes
     4. Work with TCP/IP port restrictions
     5. Work with TCP/IP remote system information

    10. Work with TCP/IP host table entries
    11. Merge TCP/IP host table
    12. Change TCP/IP domain information

    20. Configure TCP/IP applications
    21. Configure related tables
    22. Configure point-to-point TCP/IP


Selection or command
===> 3

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
```

Figure 665. Configuring the TCP/IP settings on AS21 green-screen interface

b. Enter menu option `3` (Change TCP/IP attributes). Change the IP Datagram forwarding parameter to `*YES` as shown in Figure 666.

```
                    Change TCP/IP Attributes (CHGTCPA)

Type choices, press Enter.

TCP keep alive . . . . . .   120          1-40320, *SAME, *DFT
TCP urgent pointer . . . . .  *BSD         *SAME, *BSD, *RFC
TCP receive buffer size  . . .  8192       512-8388608, *SAME, *DFT
TCP send buffer size . . . . .  8192       512-8388608, *SAME, *DFT
UDP checksum . . . . . .   *YES          *SAME, *YES, *NO
IP datagram forwarding . . . . .  *YES     *SAME, *YES, *NO
IP source routing  . . . . . .  *YES       *SAME, *YES, *NO
IP reassembly time-out . . . . .  10       5-120, *SAME, *DFT
IP time to live  . . . . . .   64          1-255, *SAME, *DFT
ARP cache timeout  . . . . . .  15         1-1440, *SAME, *DFT
Log protocol errors  . . . . . .  *NO      *SAME, *YES, *NO




                                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

*Figure 666. Changing the IP datagram forwarding parameter to *YES*

4. Configure TCP/IP on the Windows NT NTPL system, follow these steps:

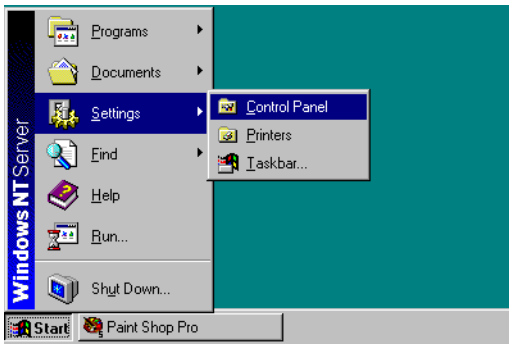


Figure 667. Selecting Control Panel from the Settings menu

a. On the Windows menu, click **Start->Settings->Control Panel** as shown in Figure 667. The display shown in Figure 668 appears.

Figure 668. Selecting the Network icon from the Control panel

b. Double-click **Network**. The display shown in Figure 669 on page 558 appears.

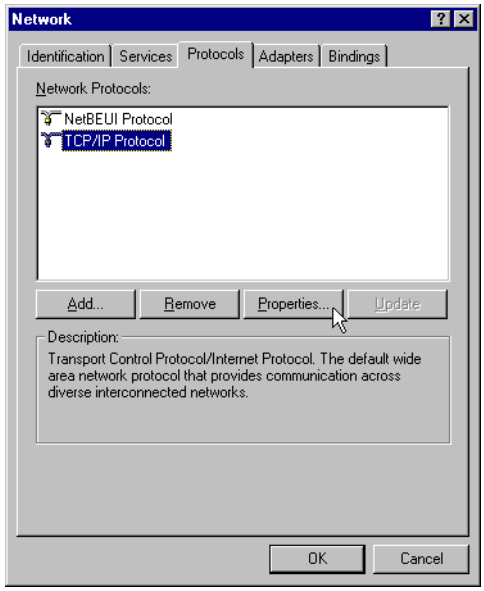


Figure 669. Selecting Properties for the TCP/IP Protocol

c. Click **Protocols**. A list of available protocols is displayed. Select **TCP/IP protocol**, and click **Properties**. The display shown in Figure 670 appears.

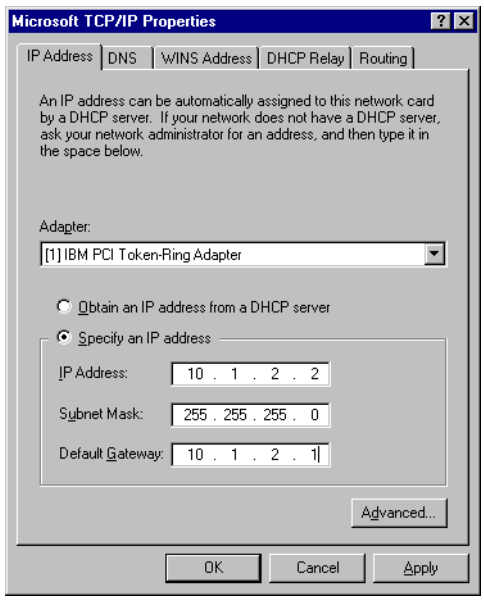


Figure 670. Configure IP address to the NTPL system and Default Gateway (AS21)

d. Specify the IP address information in the relevant fields. Click **OK**. Restart the PC if your are asked to do so.

14.7.1.3 Testing the scenario

The test of the scenario is done on the NTPL system. Start a command entry on the PC and run the PING command from the PC. If any reply is received, the connection is working properly (Figure 671).

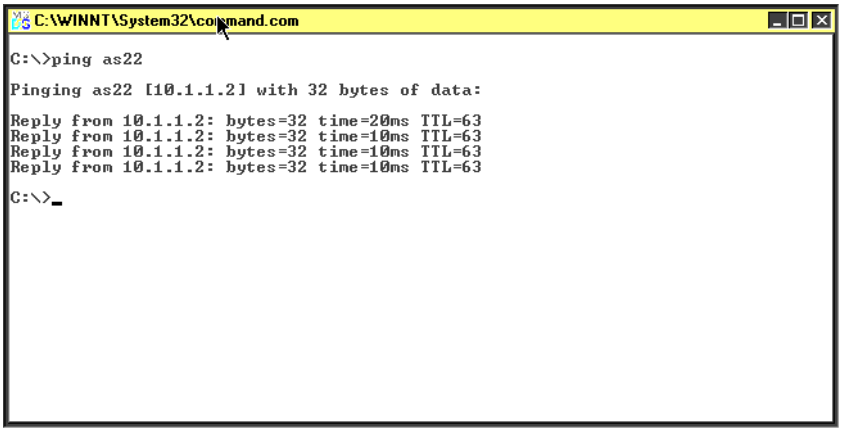


Figure 671. Testing the connection to AS22 using the PING command from the PC

To verify which route is used from the NTPL system to the AS22 system, use the PC Traceroute (TRACERT) command. This command shows the intermediate systems that the IP data packets use to reach the destination AS22 system. As Figure 672 shows, the scenario correctly uses 10.1.2.1 (the AS21system) as an intermediate hop on its way to AS22.

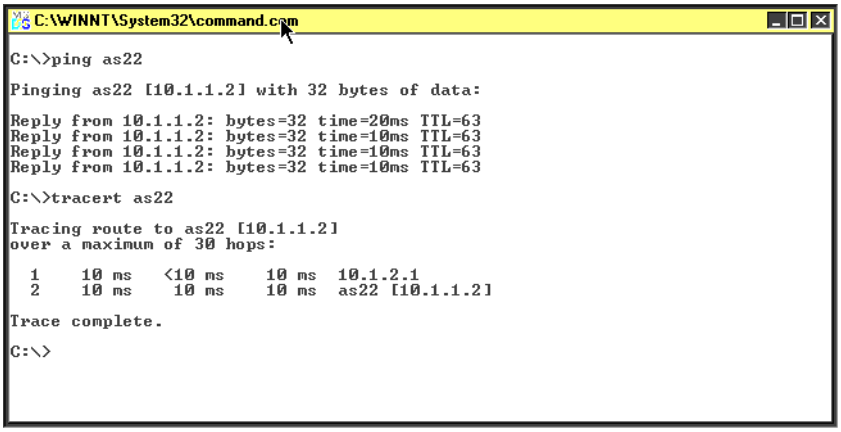


Figure 672. Testing the connection to the AS22 system using the TRACERT command

The final test is to start an FTP session to the AS22 system. Refer to Chapter 6, “Using FTP on the AS/400 system” on page 253, for more information on configuration and use of the FTP function. The FTP server on the AS22 system must be active to test the scenario.

As Figure 673 on page 560 shows, the connection to the AS22 system is successful. The FTP session is established, prompting the user for a user ID.

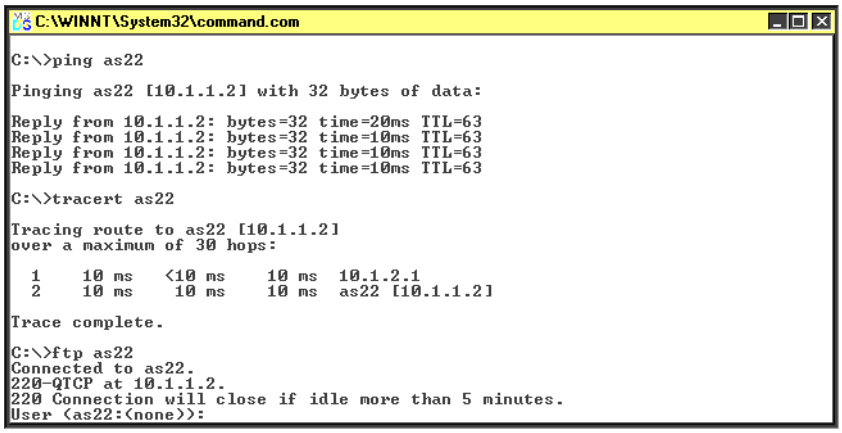


Figure 673. Testing the connection to the AS22 system using the FTP Command

14.7.2 The AS/400 system at a central site and at a remote site

The scenario shows the use of the AS/400 system at two separate networks, connected by a dedicated router acting as an intermediate system between dislike LANs.

14.7.2.1 Task overview

This task accomplishes these events:

1. Connect the two AS/400 systems to networks. Connect the AS21 system to the Token-Ring network and the AS22 system to the Ethernet network.
2. Create and start the IP interfaces on the two AS/400 systems AS21 and AS22.
3. Configure a static default route on both the AS21 and the AS22 system.
4. Connect the router to the two networks.
5. Configure two IP interfaces on the router: one for the Token-Ring and one for the Ethernet.
6. Test the connection by using Telnet.

14.7.2.2 Configuring the scenario

Refer to Chapter 2, “TCP/IP basic installation and configuration” on page 7, to create the basic setup of the IP interfaces on the two AS/400 systems. Use the following IP addresses on the AS/400 systems:

- AS21: 10.1.1.1/24 on the 10.1.1.0/24 network
- AS22: 10.1.2.1/24 on the 10.1.2.0/24 network

1. The configuration of the static default route entry on both systems is done using the green-screen interface as shown in the following steps. The configuration of the AS21 system is shown first, followed by the configuration of the AS22 system.
   a. On the AS21, go to the command entry, and enter `CFGTCP` to get to the Configure TCP/IP menu.
   b. Enter menu option `2` to work with the TCP/IP routes (Figure 674).

```
CFGTCP                          Configure TCP/IP
                                                              System:   AS21
Select one of the following:

     1. Work with TCP/IP interfaces
     2. Work with TCP/IP routes
     3. Change TCP/IP attributes
     4. Work with TCP/IP port restrictions
     5. Work with TCP/IP remote system information

    10. Work with TCP/IP host table entries
    11. Merge TCP/IP host table
    12. Change TCP/IP domain information

    20. Configure TCP/IP applications
    21. Configure related tables
    22. Configure point-to-point TCP/IP


Selection or command
===> 2

F3=Exit    F4=Prompt    F9=Retrieve    F12=Cancel
```

Figure 674. Adding routes on the AS21 system

c. Add the default route to the routing table. Notice that the *DFTROUTE entry has a subnet mask of *NONE (Figure 675).

```
                          Work with TCP/IP Routes
                                                              System:   AS21
Type options, press Enter.
  1=Add   2=Change   4=Remove   5=Display

     Route            Subnet           Next             Preferred
Opt  Destination      Mask             Hop              Interface
 1   *DFTROUTE        *NONE            10.1.1.2

                                                                      Bottom
F3=Exit      F5=Refresh   F6=Print list   F11=Display type of service
F12=Cancel   F17=Top      F18=Bottom
```

*Figure 675. Adding the *DFTROUTE entry to the routing table on AS21*

d. Press Enter. The Add TCP/IP Route (ADDTCPRTE) command appears, allowing you to specify additional parameters. Press Enter to select the default values (Figure 676 on page 562).

```
                        Add TCP/IP Route (ADDTCPRTE)

Type choices, press Enter.

Route destination  . . . . . > *DFTROUTE
Subnet mask  . . . . . > *NONE
Type of service  . . . . . .   *NORMAL       *MINDELAY, *MAXTHRPUT...
Next hop . . . . . > '10.1.1.2'
Preferred binding interface  . .   *NONE
Maximum transmission unit  . . .   576          576-16388, *IFC
Route metric . . . . . .   1            1-16
Route redistribution . . . . . .   *NO          *NO, *YES
Duplicate route priority . . . .   5            1-10

                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 676. Parameters on the ADDTCPRTE command

After successfully adding the entry (Figure 677), the AS21 routing table is ready.

```
                          Work with TCP/IP Routes
                                                          System:   AS21
Type options, press Enter.
  1=Add   2=Change   4=Remove   5=Display

     Route            Subnet           Next             Preferred
Opt  Destination      Mask             Hop              Interface

     *DFTROUTE        *NONE            10.1.1.2         *NONE

                                                                      Bottom
F3=Exit      F5=Refresh   F6=Print list   F11=Display type of service
F12=Cancel   F17=Top      F18=Bottom
TCP/IP route added successfully.
```

Figure 677. The complete routing table on AS21

2. On the AS22 system, the configuration is almost identical, except for the IP address of the default route. Go to the command entry, and enter `CFGTCP` to get to the Configure TCP/IP menu (Figure 678).

```
CFGTCP                          Configure TCP/IP
                                                              System:   AS22
Select one of the following:

     1. Work with TCP/IP interfaces
     2. Work with TCP/IP routes
     3. Change TCP/IP attributes
     4. Work with TCP/IP port restrictions
     5. Work with TCP/IP remote system information

    10. Work with TCP/IP host table entries
    11. Merge TCP/IP host table
    12. Change TCP/IP domain information

    20. Configure TCP/IP applications
    21. Configure related tables
    22. Configure point-to-point TCP/IP


Selection or command
===> 2

F3=Exit    F4=Prompt    F9=Retrieve    F12=Cancel
```

Figure 678. Configuring the Routing table on AS22

a. Add the default route to the routing table. Notice that the *DFTROUTE entry has a subnet mask of *NONE (Figure 679).

```
                          Work with TCP/IP Routes
                                                              System:   AS22
Type options, press Enter.
  1=Add   2=Change   4=Remove   5=Display

     Route            Subnet           Next             Preferred
Opt  Destination      Mask             Hop              Interface
 1   *DFTROUTE        *NONE            10.1.2.2

                                                                        Bottom
F3=Exit      F5=Refresh   F6=Print list   F11=Display type of service
F12=Cancel   F17=Top      F18=Bottom
```

*Figure 679. Adding the *DFTROUTE entry on AS22*

b. Press Enter. The Add TCP/IP Route (ADDTCPRTE) command appears, allowing you to specify additional parameters. Press Enter to select the default values (Figure 680 on page 564).

```
                        Add TCP/IP Route (ADDTCPRTE)

Type choices, press Enter.

Route destination  . . . . . > *DFTROUTE
Subnet mask  . . . . . > *NONE
Type of service  . . . . . *NORMAL       *MINDELAY, *MAXTHRPUT...
Next hop . . . . . > '10.1.2.2'
Preferred binding interface  . .  *NONE
Maximum transmission unit  . . .  576          576-16388, *IFC
Route metric . . . . . 1            1-16
Route redistribution . . . . . *NO          *NO, *YES
Duplicate route priority . . . .  5            1-10

                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 680. Additional parameters on the ADDTCPRTE command

c. After successfully adding the entry (Figure 681), the AS22 routing table is ready.

```
                          Work with TCP/IP Routes
                                                          System:   AS22
Type options, press Enter.
  1=Add   2=Change   4=Remove   5=Display

     Route            Subnet           Next             Preferred
Opt  Destination      Mask             Hop              Interface

     *DFTROUTE        *NONE            10.1.2.2         *NONE

                                                                      Bottom
F3=Exit      F5=Refresh   F6=Print list   F11=Display type of service
F12=Cancel   F17=Top      F18=Bottom
TCP/IP route added successfully.
```

*Figure 681. The routing table on AS22 (*DFTROUTE Added)*

3. The configuration of the router is not shown in this configuration.

a. The following interfaces should be configured on the router:
   – Token-Ring interface: 10.1.1.2/24
   – Ethernet interface: 10.1.2.2/24

b. No static or dynamic routing should be configured on the router, because the router knows the existence of both the Token-Ring and the Ethernet networks due to its two configured interfaces.

c. After the configuration of the router has been completed, connect the router to both the Token-Ring and the Ethernet networks.

14.7.2.3 Testing the scenario

Testing the scenario can be done by using the Start TCP/IP TELNET (TELNET) command from the AS21 (10.1.1.1) system to start a TN5250 session at the AS22 (10.1.2.1) system. See Figure 682 and Figure 683 on page 566.

```
MAIN                              AS/400 Main Menu
                                                              System:   AS21
Select one of the following:

     1. User tasks
     2. Office tasks
     3. General system tasks
     4. Files, libraries, and folders
     5. Programming
     6. Communications
     7. Define or change the system
     8. Problem handling
     9. Display a menu
    10. Information Assistant options
    11. Client Access/400 tasks

    90. Sign off

Selection or command
===> telnet '10.1.2.1'

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=Information Assistant
F23=Set initial menu
```

Figure 682. Testing the scenario using the TELNET command

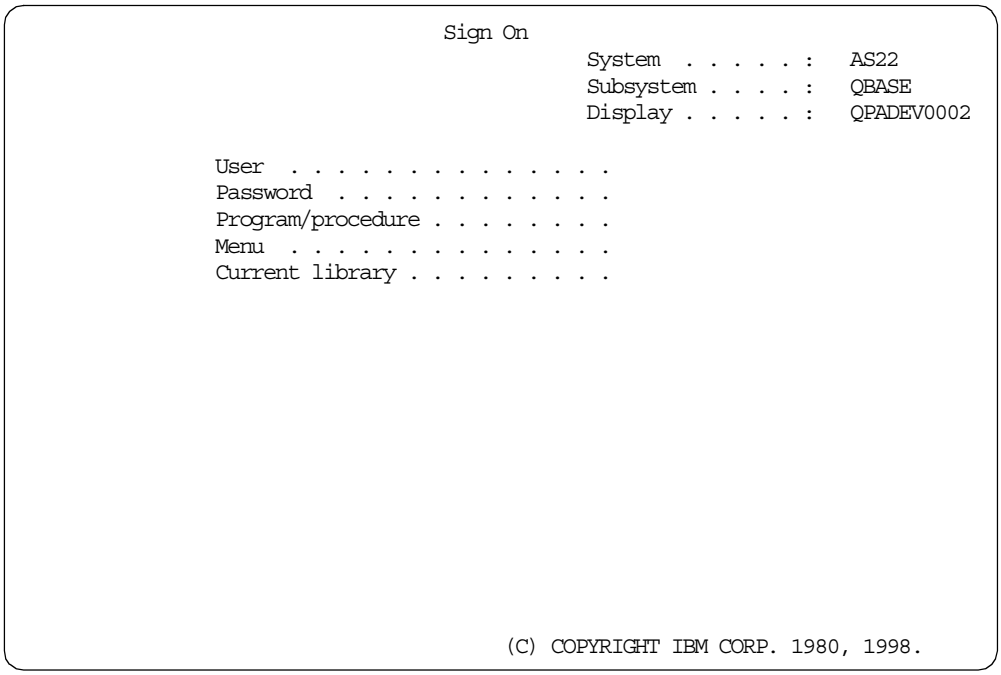


Figure 683. Successful establishment of connection to AS22 using TELNET

14.7.3 The AS/400 system at the central site and remote sites

This scenario shows the use of AS/400 systems connected by several dedicated routers. The two remote AS/400 systems connect to the centrally located AS/400 system.

14.7.3.1 Task overview

This task covers these steps:

1. Connect the three AS/400 systems to networks.
2. Create and start the IP interfaces on the three AS/400 systems AS21, AS22, and AS23.
3. Configure two routing entries at the AS21 system, pointing to each of the remote networks.
4. Configure a default route on both the AS22 and the AS23 system, both pointing back at the centrally located AS21 via the routers.
5. Connect the two routers to the networks.
6. Configure the IP interfaces on the routers
7. Test the connections by using Telnet and FTP.

14.7.3.2 Configuring the scenario

Refer to Chapter 2, “TCP/IP basic installation and configuration” on page 7, to create the basic setup of the IP interfaces on the three AS/400 systems. Use the following IP addresses on the AS/400 systems:

- AS21: 10.1.1.1/24 on the 10.1.1.0/24 network
- AS22: 10.1.3.1/24 on the 10.1.3.0/24 network
- AS23: 10.1.2.1/24 on the 10.1.2.0/24 network

1. The configuration of the routing entries at AS21 system is done using the green-screen interface as shown in the following steps:

   a. On the AS21, go to the command entry, and enter `CFGTCP` to get to the Configure TCP/IP menu.

   b. Enter menu option `2` to work with the TCP/IP routes (Figure 684).

```
CFGTCP                          Configure TCP/IP
                                                          System:   AS21
Select one of the following:

     1. Work with TCP/IP interfaces
     2. Work with TCP/IP routes
     3. Change TCP/IP attributes
     4. Work with TCP/IP port restrictions
     5. Work with TCP/IP remote system information

    10. Work with TCP/IP host table entries
    11. Merge TCP/IP host table
    12. Change TCP/IP domain information

    20. Configure TCP/IP applications
    21. Configure related tables
    22. Configure point-to-point TCP/IP


Selection or command
===> 2

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
```

Figure 684. Adding routing entries to the AS21 system

   c. Add the routing entry for the 10.1.2.0/24 network (Figure 685 and Figure 686 on page 568).

```
                          Work with TCP/IP Routes
                                                              System:   AS21
Type options, press Enter.
  1=Add   2=Change   4=Remove   5=Display

     Route            Subnet           Next             Preferred
Opt  Destination      Mask             Hop              Interface
 1   10.1.2.0         255.255.255.0    10.1.1.2




                                                                       Bottom
F3=Exit      F5=Refresh   F6=Print list   F11=Display type of service
F12=Cancel   F17=Top      F18=Bottom
```

Figure 685. Adding a routing entry to the AS23 system (Part 1)

```
                        Add TCP/IP Route (ADDTCPRTE)

Type choices, press Enter.

Route destination  . . . . . > '10.1.2.0'
Subnet mask  . . . . . > '255.255.255.0'
Type of service  . . . . . *NORMAL       *MINDELAY, *MAXTHRPUT...
Next hop . . . . . > '10.1.1.2'
Preferred binding interface  . .  *NONE
Maximum transmission unit  . . .  576          576-16388, *IFC
Route metric . . . . . 1            1-16
Route redistribution . . . . . *NO          *NO, *YES
Duplicate route priority . . . .  5            1-10









                                                                       Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 686. Adding a routing entry to the AS23 system (Part 2)

d. Add the routing entry for the 10.1.3.0/24 network (Figure 687 and Figure 688).

```
                          Work with TCP/IP Routes
                                                              System:   AS21
Type options, press Enter.
  1=Add   2=Change   4=Remove   5=Display

     Route            Subnet           Next             Preferred
Opt  Destination      Mask             Hop              Interface
 1   10.1.3.0         255.255.255.0    10.1.1.3
     10.1.2.0         255.255.255.0    10.1.1.2

                                                                      Bottom
F3=Exit      F5=Refresh   F6=Print list   F11=Display type of service
F12=Cancel   F17=Top      F18=Bottom
TCP/IP route added successfully.
```

Figure 687. Adding a routing entry to the AS22 System (Part 1)

```
                          Add TCP/IP Route (ADDTCPRTE)

Type choices, press Enter.

Route destination  . . . . . > '10.1.3.0'
Subnet mask  . . . . . > '255.255.255.0'
Type of service  . . . . . *NORMAL       *MINDELAY, *MAXTHRPUT...
Next hop . . . . . > '10.1.1.3'
Preferred binding interface  . .  *NONE
Maximum transmission unit  . . .  576          576-16388, *IFC
Route metric . . . . . .  1            1-16
Route redistribution . . . . . .  *NO          *NO, *YES
Duplicate route priority . . . .  5            1-10

                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 688. Adding a routing entry to the AS22 system (Part 2)

The complete routing table on the AS21 system is shown in Figure 689 on page 570.

```
                          Work with TCP/IP Routes
                                                              System:   AS21
Type options, press Enter.
  1=Add   2=Change   4=Remove   5=Display

     Route            Subnet           Next             Preferred
Opt  Destination      Mask             Hop              Interface

     10.1.2.0         255.255.255.0    10.1.1.2         *NONE
     10.1.3.0         255.255.255.0    10.1.1.3         *NONE

                                                                    Bottom
F3=Exit      F5=Refresh   F6=Print list   F11=Display type of service
F12=Cancel   F17=Top      F18=Bottom
```

Figure 689. The complete routing table on the AS21 system

2. The configuration of the routing entries at AS22 is done using the green-screen interface as shown in the following steps:
   a. On the AS22, go to the command entry, and enter `CFGTCP` to get to the Configure TCP/IP menu.
   b. Enter menu option `2` to work with the TCP/IP routes (Figure 690).

```
CFGTCP                          Configure TCP/IP
                                                              System:   AS22
Select one of the following:

     1. Work with TCP/IP interfaces
     2. Work with TCP/IP routes
     3. Change TCP/IP attributes
     4. Work with TCP/IP port restrictions
     5. Work with TCP/IP remote system information

    10. Work with TCP/IP host table entries
    11. Merge TCP/IP host table
    12. Change TCP/IP domain information

    20. Configure TCP/IP applications
    21. Configure related tables
    22. Configure point-to-point TCP/IP


Selection or command
===> 2

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
```

Figure 690. Configuring the routing entries on the AS22 System

c. Add a default routing entry on the AS22 system (Figure 691 and Figure 692).

```
                          Work with TCP/IP Routes
                                                              System:   AS22
Type options, press Enter.
  1=Add   2=Change   4=Remove   5=Display

     Route            Subnet           Next             Preferred
Opt  Destination      Mask             Hop              Interface
 1   *DFTROUTE        *NONE            10.1.3.2

                                                                        Bottom
F3=Exit      F5=Refresh   F6=Print list   F11=Display type of service
F12=Cancel   F17=Top      F18=Bottom
```

Figure 691. Adding a default routing entry on the AS22 system (Part 1)

```
                          Add TCP/IP Route (ADDTCPRTE)

Type choices, press Enter.

Route destination  . . . . . > *DFTROUTE
Subnet mask  . . . . . > *NONE
Type of service  . . . . . *NORMAL       *MINDELAY, *MAXTHRPUT...
Next hop . . . . . > '10.1.3.2'
Preferred binding interface  . .  *NONE
Maximum transmission unit  . . .  576          576-16388, *IFC
Route metric . . . . . .  1            1-16
Route redistribution . . . . . .  *NO          *NO, *YES
Duplicate route priority . . . .  5            1-10

                                                                        Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 692. Adding a default routing entry at the AS22 system (Part 2)

The complete routing table on the AS22 system is shown in Figure 693 on page 572.

```
                          Work with TCP/IP Routes
                                                               System:   AS22
Type options, press Enter.
  1=Add   2=Change   4=Remove   5=Display

     Route            Subnet           Next             Preferred
Opt  Destination      Mask             Hop              Interface

     *DFTROUTE        *NONE            10.1.3.2         *NONE

                                                                       Bottom
F3=Exit      F5=Refresh   F6=Print list   F11=Display type of service
F12=Cancel   F17=Top      F18=Bottom
TCP/IP route added successfully.
```

Figure 693. The complete routing table on the AS22 system

3. The configuration of the routing entries at AS23 is done using the green-screen interface as shown here:
   a. On the AS23, go to the command entry, and enter `CFGTCP` to get to the Configure TCP/IP menu.
   b. Enter menu option `2` to work with the TCP/IP routes (Figure 694).

```
CFGTCP                         Configure TCP/IP
                                                               System:   AS23
Select one of the following:

     1. Work with TCP/IP interfaces
     2. Work with TCP/IP routes
     3. Change TCP/IP attributes
     4. Work with TCP/IP port restrictions
     5. Work with TCP/IP remote system information

    10. Work with TCP/IP host table entries
    11. Merge TCP/IP host table
    12. Change TCP/IP domain information

    20. Configure TCP/IP applications
    21. Configure related tables
    22. Configure point-to-point TCP/IP


Selection or command
===> 2

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
```

Figure 694. Configuring a default routing entry on the AS23 system

c. Add a default routing entry to the AS23 system (Figure 695). Place a `1` in the option field and complete the other information shown. Press Enter. The display shown in Figure 696 appears.

```
                          Work with TCP/IP Routes
                                                              System:   AS23
Type options, press Enter.
  1=Add   2=Change   4=Remove   5=Display

     Route            Subnet           Next             Preferred
Opt  Destination      Mask             Hop              Interface
 1   *DFTROUTE        *NONE            10.1.2.2

                                                                        Bottom
F3=Exit      F5=Refresh   F6=Print list   F11=Display type of service
F12=Cancel   F17=Top      F18=Bottom
```

Figure 695. Adding a default routing entry on the AS23 system (Part 1)

d. Enter any additional information that you require, and press Enter.

```
                        Add TCP/IP Route (ADDTCPRTE)

Type choices, press Enter.

Route destination  . . . . . > *DFTROUTE
Subnet mask  . . . . . > *NONE
Type of service  . . . . . *NORMAL       *MINDELAY, *MAXTHRPUT...
Next hop . . . . . > '10.1.2.2'
Preferred binding interface  . .  *NONE
Maximum transmission unit  . . .  576          576-16388, *IFC
Route metric . . . . . .  1            1-16
Route redistribution . . . . . .  *NO          *NO, *YES
Duplicate route priority . . . .  5            1-10

                                                                        Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 696. Adding a default routing entry on the AS23 system (Part 2)

The complete routing table at the AS23 system is shown in Figure 697 on page 574.

```
                          Work with TCP/IP Routes
                                                              System:   AS23
Type options, press Enter.
  1=Add   2=Change   4=Remove   5=Display

     Route            Subnet           Next             Preferred
Opt  Destination      Mask             Hop              Interface

     *DFTROUTE        *NONE            10.1.2.2         *NONE









                                                                      Bottom
F3=Exit      F5=Refresh   F6=Print list   F11=Display type of service
F12=Cancel   F17=Top      F18=Bottom
TCP/IP route added successfully.
```

Figure 697. The complete routing table on the AS23 system

4. The configuration of the routers is not shown in this configuration.
   a. The following interfaces should be configured on the routers:

      Router A

      – Token-Ring interface 1: 10.1.1.2/24
      – Token-Ring interface 2: 10.1.2.2/24

      Router B

      – Ethernet interface: 10.1.3.2/24
      – Token-Ring interface: 10.1.1.3/24

   b. No static or dynamic routing should be configured on the routers.
   c. After the configuration of the routers has been completed, connect the routers to both the Token-Ring and the Ethernet networks.

14.7.3.3 Testing the scenario

Testing the scenario is performed by starting a Telnet session from the AS23 system to the AS21 system and starting a FTP session from AS22 system to the AS21 system. Follow these steps:

1. Start a Telnet session from the AS23 system to the AS21 system as shown in Figure 698.

```
CFGTCP                          Configure TCP/IP
                                                              System:   AS23
Select one of the following:

     1. Work with TCP/IP interfaces
     2. Work with TCP/IP routes
     3. Change TCP/IP attributes
     4. Work with TCP/IP port restrictions
     5. Work with TCP/IP remote system information

    10. Work with TCP/IP host table entries
    11. Merge TCP/IP host table
    12. Change TCP/IP domain information

    20. Configure TCP/IP applications
    21. Configure related tables
    22. Configure point-to-point TCP/IP


Selection or command
===> telnet '10.1.1.1'

F3=Exit    F4=Prompt    F9=Retrieve    F12=Cancel
```

Figure 698. Starting a Telnet session from AS23 to AS21 (10.1.1.1)

```
                                  Sign On
                                              System  . . . . . :   AS21
                                              Subsystem . . . . :   QINTER
                                              Display . . . . . :   QPADEV0003

                User  . . . . . .
                Password  . . . . . .
                Program/procedure . . . . . .
                Menu  . . . . . .
                Current library . . . . . .









                                        (C) COPYRIGHT IBM CORP. 1980, 1998.
```

Figure 699. The Telnet session at the destination AS21 system

2. Start an FTP session from the AS22 system to the AS21 system as shown in Figure 700 and Figure 701 on page 576.

```
CFGTCP                          Configure TCP/IP
                                                              System:   AS22
Select one of the following:

     1. Work with TCP/IP interfaces
     2. Work with TCP/IP routes
     3. Change TCP/IP attributes
     4. Work with TCP/IP port restrictions
     5. Work with TCP/IP remote system information

    10. Work with TCP/IP host table entries
    11. Merge TCP/IP host table
    12. Change TCP/IP domain information

    20. Configure TCP/IP applications
    21. Configure related tables
    22. Configure point-to-point TCP/IP


Selection or command
===> ftp '10.1.1.1'

F3=Exit    F4=Prompt    F9=Retrieve    F12=Cancel
```

Figure 700. Starting the FTP session to the AS21 system (10.1.1.1)

```
                            File Transfer Protocol

Previous FTP subcommands and messages:
  Connecting to remote host 10.1.1.1 using port 21.
  220-QTCP at 10.1.1.1.
  220 Connection will close if idle more than 5 minutes.









Enter login ID (itscid29):
===>



F3=Exit      F6=Print       F9=Retrieve
F17=Top    F18=Bottom    F21=CL command line
```

Figure 701. The FTP session at the remote AS21 system (10.1.1.1)

14.7.4 The AS/400 system using RIP to build a complete network map

The following section describes, in detail, the configuration and the testing of scenario 4.

14.7.4.1 Task overview

This scenario involves these tasks:

1. Connect the two AS/400 systems to the Token-Ring LANs.
2. Create and start the IP interfaces on the two AS/400 systems AS21 and AS22.
3. Configure RIP on both AS/400 systems.
4. Start the RouteD daemons on both AS/400 systems.
5. Watch the routing entries on both systems as the RIP messages propagate.
6. Test the connection by using Telnet.

14.7.4.2 Configuring the scenario

Refer to Chapter 2, “TCP/IP basic installation and configuration” on page 7, to create the basic setup of the IP interfaces. Use the following IP addresses on the AS/400 systems:

- AS21: 10.1.1.1/24 on the 10.1.1.0/24 network
- AS21: 10. 1.2.1/24 on the 10.1.2.0/24 network
- AS22: 10.1.2.2/24 on the 10.1.2.0/24 network

Configure and start RIP on the AS21 system. Follow these steps:

1. Start Operations Navigator by clicking **Start -> Programs -> IBM AS400 Client Access ->AS/400 Operations Navigator**. The AS/400 Operations Navigator window appears (Figure 702).

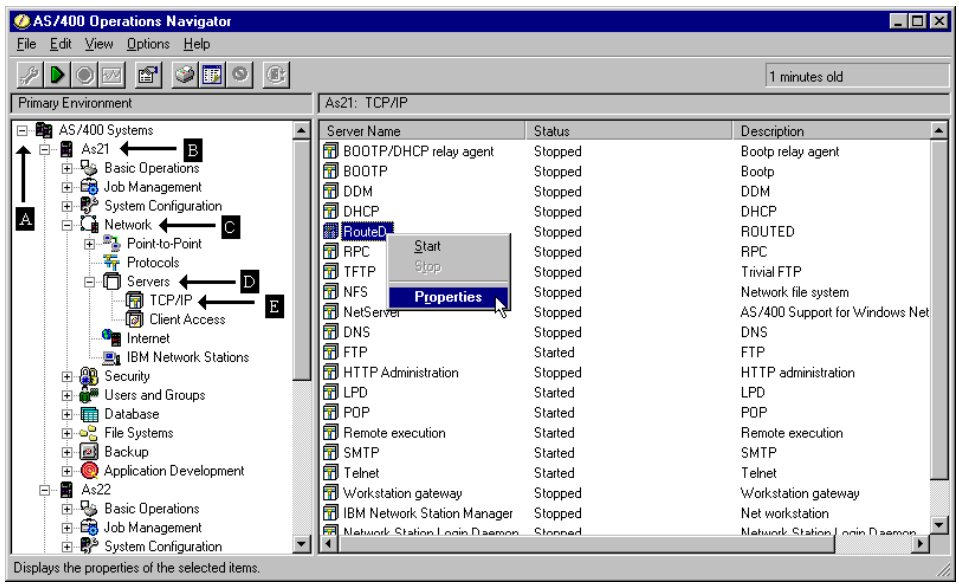


Figure 702. Starting the configuration of RIP on the AS21 system

2. Double-click the AS/400 network (A).
3. Double-click the system icon (B) for the AS21 AS/400 system that you are configuring. The system components appear.
4. Double-click **Network** (C). The network components appear.
5. Double-click **Servers** (D). The available protocols appear.
6. Double-click **TCP/IP** (E). The available services appear in the right window.
7. Use the right mouse-button to show the menu. Select **Properties** as shown in Figure 702.

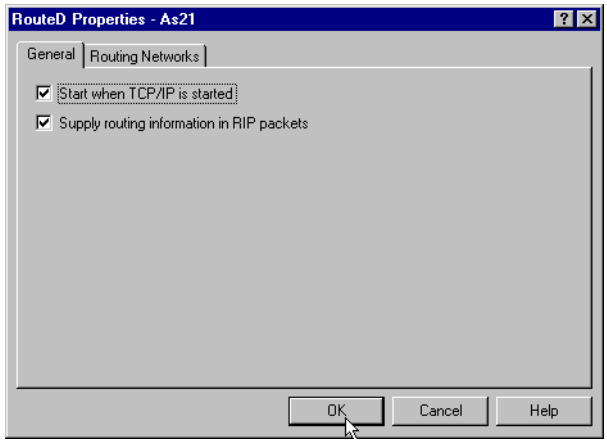


Figure 703. Enabling RIP on AS21 system

8. Select both check boxes as shown in Figure 703, and click the **OK** button
9. Start the RouteD daemon. Right-click and select **Start** (Figure 702 on page 577). If the daemon was already started, restart the server by selecting **Stop** and then **Start**.

The configuration of the AS22 system is identical to the AS21 system. Repeat steps 1 through 9 in step 1 to configure RIP on the AS22 system. This completes the configuration of RIP on both systems

14.7.4.3 Testing the scenario

The scenario can be tested by issuing the Start TCP/IP TELNET (TELNET) command at the AS22 system, connecting to the 10.1.1.1 interface at the AS21 system.

To view the routing entries at both systems, use the Work with TCP/IP Network Status (NETSTAT) command `NETSTAT OPTION(*RTE)`. This command shows the dynamically added routing entries that the AS22 system receives from the AS21 system. Please allow the systems to exchange the RIP messages. Wait two to three minutes for the routing tables to become stable.

Figure 704 shows the routing table at the AS21 system. No RIP routes are added to this system, since the AS21 has interfaces on both the 10.1.1.0/24 and the 10.1.2.0/24 networks.

```
                    Display TCP/IP Route Information
                                                              System:   AS21
Type options, press Enter.
  5=Display details

     Route            Subnet           Next              Route
Opt  Destination      Mask             Hop             Available
     10.1.2.0         255.255.255.0    *DIRECT           *YES
     10.1.1.0         255.255.255.0    *DIRECT           *YES
     127.0.0.0        255.0.0.0        *DIRECT           *YES

F3=Exit   F5=Refresh     F6=Print list   F9=Command line
F11=Display route type   F12=Cancel      F13=Sort by column   F24=More keys
```

Figure 704. Routing table on the AS21 system

Figure 705 shows the routing table on the AS22 system. It also shows the dynamically added RIP route, destination 10.1.1.0, and the address of the router connection to that network (10.1.2.1 is the AS21 interface at the 10.1.2.0/24 network).

```
                    Display TCP/IP Route Information
                                                              System:   AS22
Type options, press Enter.
  5=Display details

     Route            Subnet           Next              Route
Opt  Destination      Mask             Hop             Available
     10.1.2.0         255.255.255.0    *DIRECT           *YES
     10.1.1.0         255.255.255.0    10.1.2.1          *YES
     127.0.0.0        255.0.0.0        *DIRECT           *YES

F3=Exit   F5=Refresh     F6=Print list   F9=Command line
F11=Display route type   F12=Cancel      F13=Sort by column   F24=More keys
```

Figure 705. Routing table on the AS22 system

The actual test of the routing setup is performed by issuing the command shown in Figure 706 on page 580.

```
MAIN                          AS/400 Main Menu
                                                          System:   AS22
Select one of the following:

     1. User tasks
     2. Office tasks
     3. General system tasks
     4. Files, libraries, and folders
     5. Programming
     6. Communications
     7. Define or change the system
     8. Problem handling
     9. Display a menu
    10. Information Assistant options
    11. Client Access/400 tasks

    90. Sign off

Selection or command
===> telnet '10.1.1.1'

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=Information Assistant
F23=Set initial menu
```

Figure 706. Starting a Telnet session from the AS22 system to the AS21 system

The connection to the AS22 system is created as shown Figure 707.

```
                              Sign On
                                          System  . . . . . :   AS21
                                          Subsystem . . . . :   QINTER
                                          Display . . . . . :   QPADEV0004

              User  . . . . . . . . . . . . .
              Password  . . . . . . . . . . .
              Program/procedure . . . . . . .
              Menu  . . . . . . . . . . . . .
              Current library . . . . . . . .




                                   (C) COPYRIGHT IBM CORP. 1980, 1998.
```

Figure 707. Completed Telnet session from the AS22 system to the AS21 system

14.7.5 The AS/400 system using RIP to hide part of a complete network

The following section describes, in detail, the configuration and the testing of scenario 5.

14.7.5.1 Task overview

This task involves these steps:

1. Connect the two AS/400 systems to the Token-Ring LANs.
2. Create and start the IP interfaces on the two AS/400 systems AS21 and AS22.

3. Configure RIP on both AS/400 systems. Do not expose the 10.1.3.0/24 network from the AS22 system. We use the NOFORWARD option to do this.
4. Start the RouteD daemons on both AS/400 systems.
5. Watch the routing entries on both systems as the RIP messages propagate.
6. Test the connection by using Telnet

14.7.5.2 Configuring the scenario

Refer to Chapter 2, “TCP/IP basic installation and configuration” on page 7, to create the basic setup of the IP interfaces. Use the following IP addresses on the AS/400 systems:

- AS21: 10.1.1.1/24 on the 10.1.1.0/24 network
- AS21: 10.1.2.1/24 on the 10.1.2.0/24 network
- AS22: 10.1.2.2/24 on the 10.1.2.0/24 network
- AS22: 10.1.3.1/24 on the 10.1.3.0/24 network

Follow these steps:

1. Start Operations Navigator by clicking **Start->Programs->IBM AS400 Client Access->AS/400 Operations Navigator**. The AS/400 Operations Navigator window appears (Figure 708).

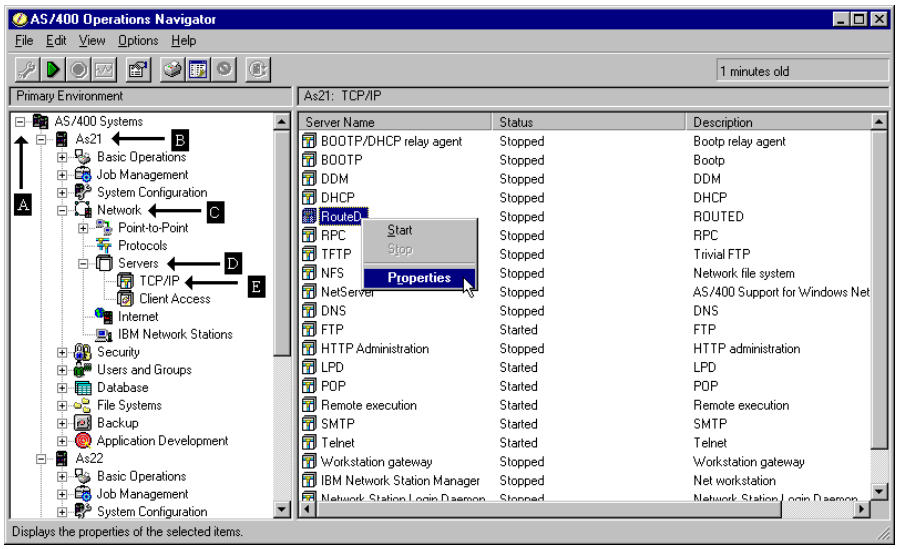


Figure 708. Starting the configuration of RIP on the AS21 system

2. Double-click the AS/400 network (A).
3. Double-click the system icon (B) for the AS21 AS/400 system that you are configuring. The system components appear.
4. Double-click **Network** (C). The network components appear.
5. Double-click **Servers** (D). The available protocols appear.
6. Double-click **TCP/IP** (E). The available services appear in the right window.
7. Use the right mouse-button to show the menu. Select **Properties** as shown in Figure 708.

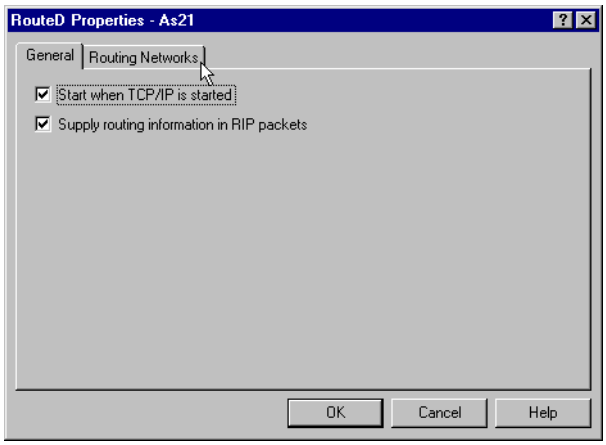


Figure 709. Enabling RIP on the AS21 system

8. Select both check boxes as shown in Figure 709, and click the **Routing Networks** tab. The display shown in Figure 710 appears.

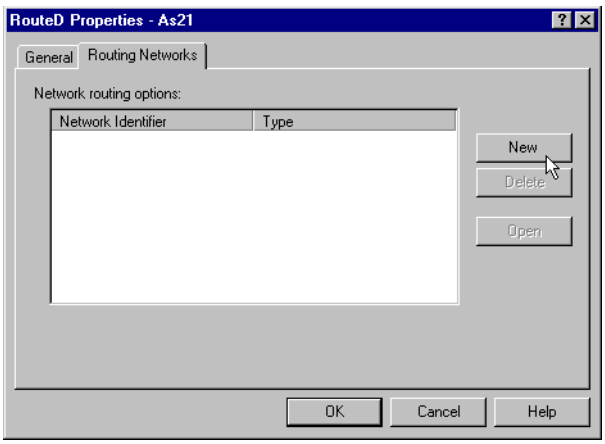


Figure 710. Adding a new network to the RIP configuration on the AS21 system

9. Select the **Add** button to add a new network. The display shown in Figure 711 appears.

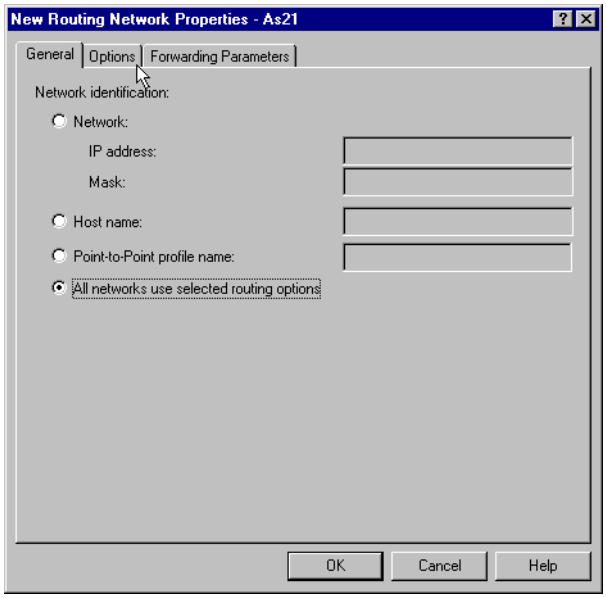


Figure 711. Selecting all the interfaces

10. Make sure that the **All networks use selected routing options** entry is selected as shown in Figure 711. Select the **Options** tab. The display shown in Figure 712 appears.

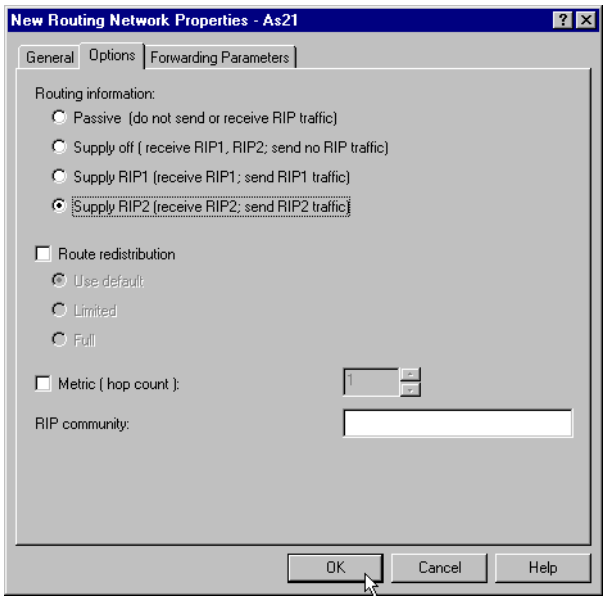


Figure 712. Supplying RIP2 on the network

11. Select **Supply RIP2 (receive RIP2; send RIP2 traffic)** as shown in Figure 712. Click **OK** to confirm the options. The display shown in Figure 713 on page 584 appears. Notice that the new entry is added in the window.

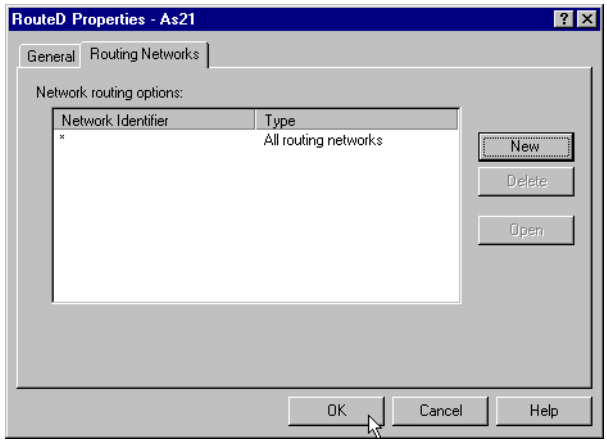


Figure 713. Confirming the configuration by clicking Ok

12.Click **OK** to confirm the complete configuration.

13.Restart the RouteD daemon on AS21 as shown in Figure 714.

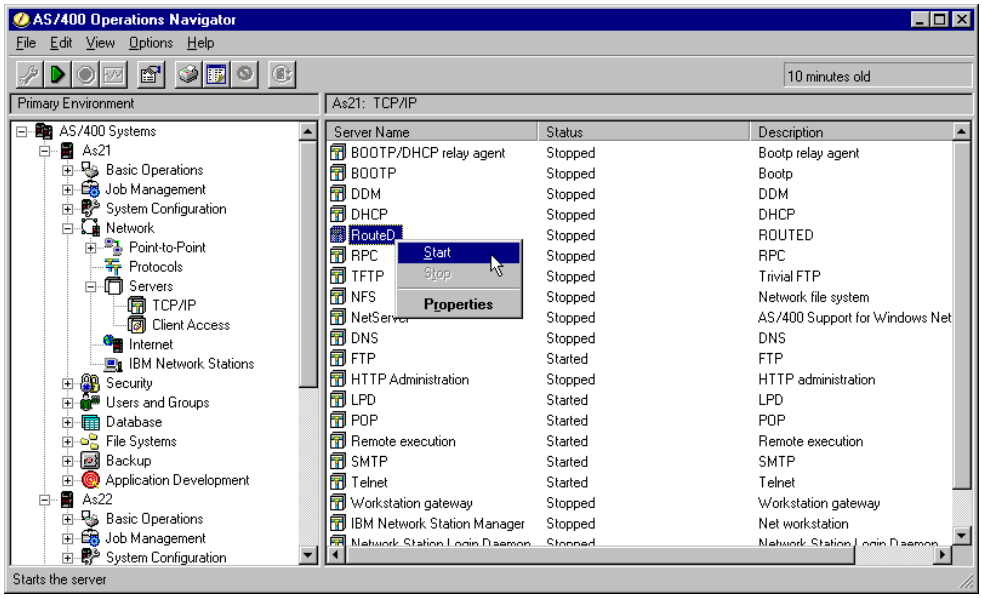


Figure 714. Restarting the RouteD daemon on the AS21 system

The configuration of the AS22 AS/400 system is almost identical. These are the steps necessary to create the configuration. Follow the steps shown in Figure 715 to get to the TCP/IP server window.

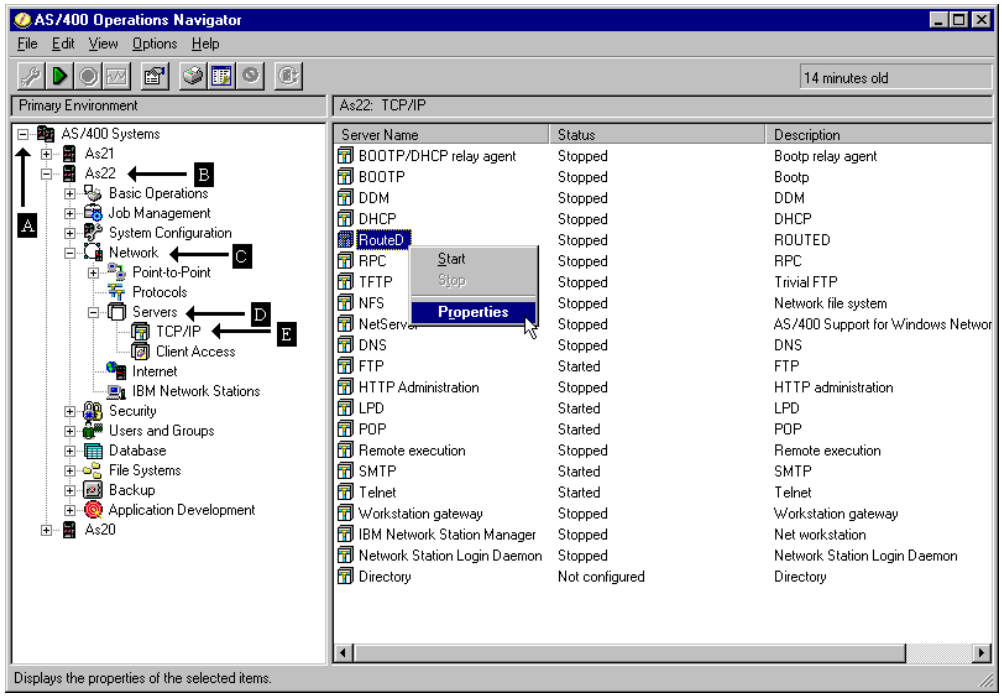


Figure 715. Selecting Properties

1. Right-click RouteD. Select **Properties** from the pop-up menu. The display shown in Figure 716 appears.

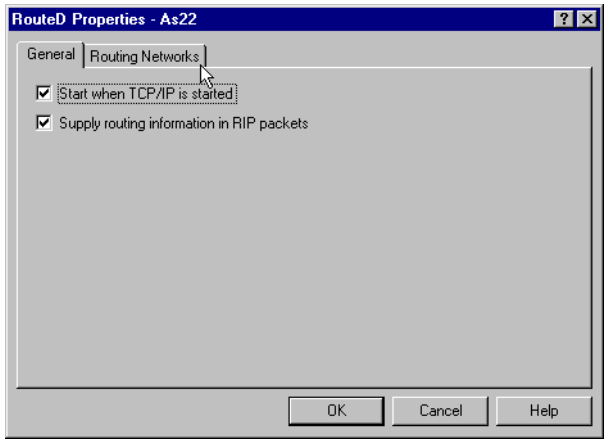


Figure 716. Selecting all options on the General tab

2. Select both check boxes. Click the **Routing Networks** tab. The display shown in Figure 717 on page 586 appears.

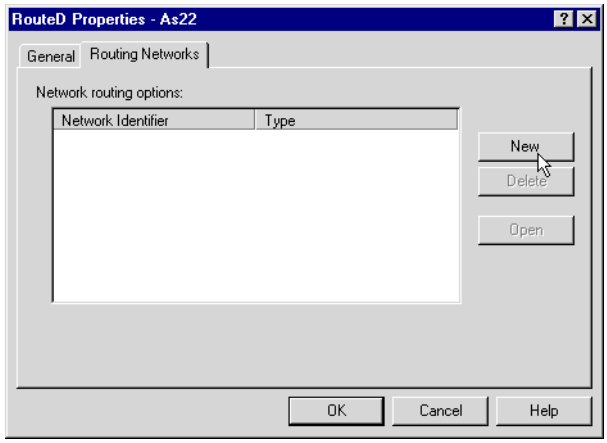


Figure 717. Adding a network entry using the Add button

3. Click **New** to add a new network. The display shown in Figure 718 appears.

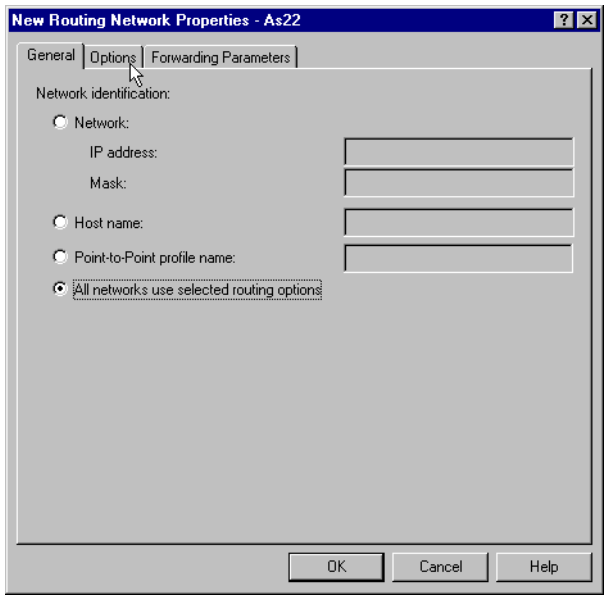


Figure 718. Selecting all networks and the Options tab

4. Select **All Networks use selected routing options**. Click **Options**. The display shown in Figure 719 appears.

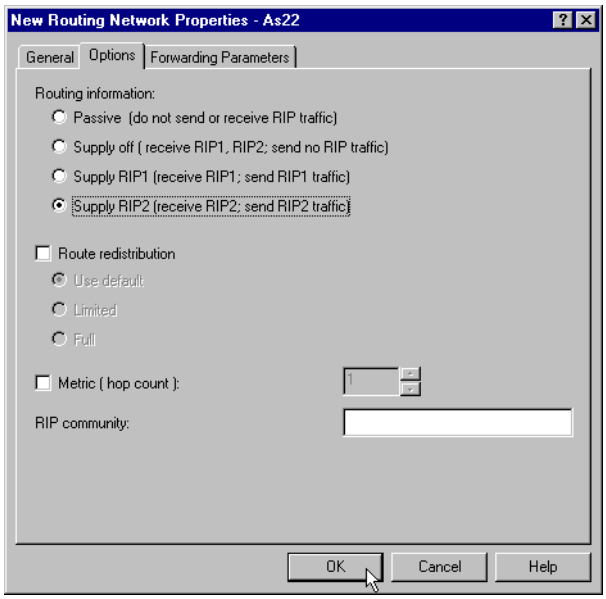


Figure 719. Selecting RIP 2

5. Select **Supply RIP2**. Click **OK**. The display shown in Figure 720 appears. Notice that the entry for all networks (*) is now listed in the Routing Networks window.

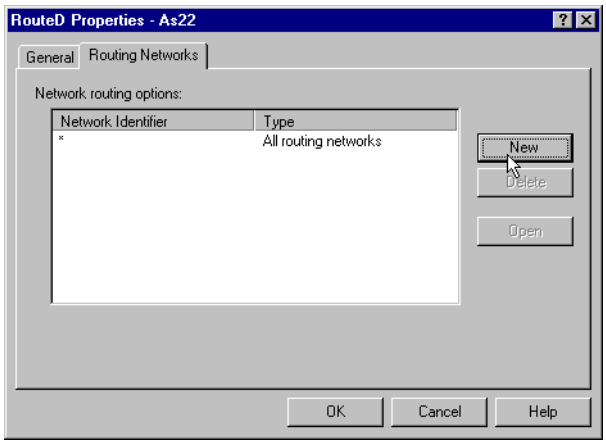


Figure 720. Clicking New to add a net entry for the 10.1.3.0/24 network

6. Add another new network. Click **New**. The display shown in Figure 721 on page 588 appears.

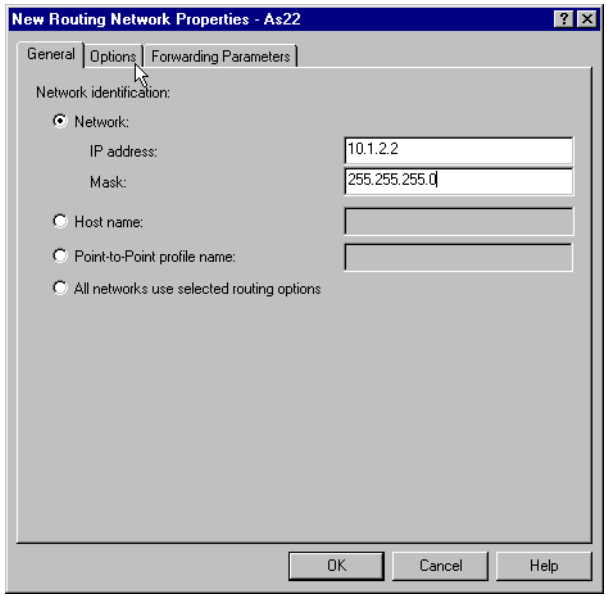


Figure 721. New Routing Network Properties: AS22

7. Enter the address of the IP interface that propagates RIP packets from the hidden network (10.1.2.2) and the correct mask (255.255.255.0). Click **Options**. The display shown in Figure 722 appears.

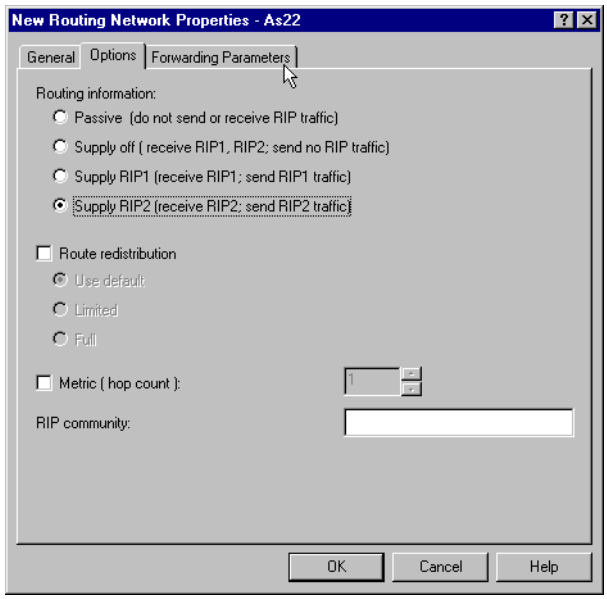


Figure 722. Selecting RIP 2 as the default

8. Select **RIP2**. Click **Forwarding Parameters**. The display shown in Figure 723 appears.

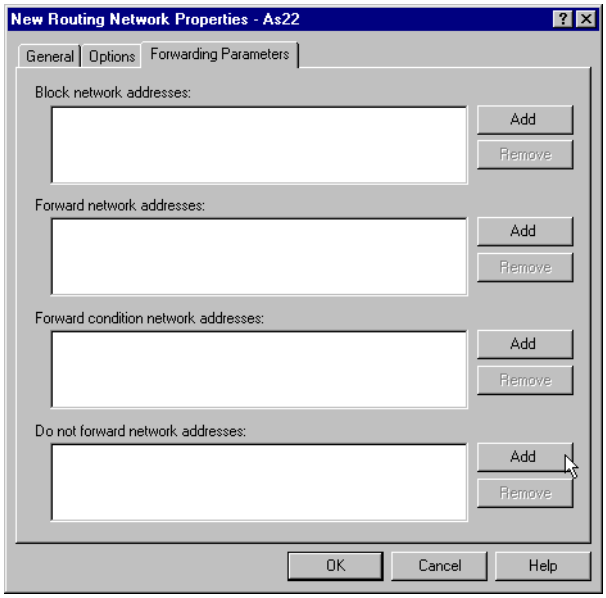


Figure 723. Using the Add button in the do not forward part of the window

9. Click the **Forwarding Parameters** tab, and click the **Add** button beside the Do not forward network addresses section. An input field opens in the window. Type the network address to hide (10.1.3.0) and press enter. The window shown in Figure 724 appears.

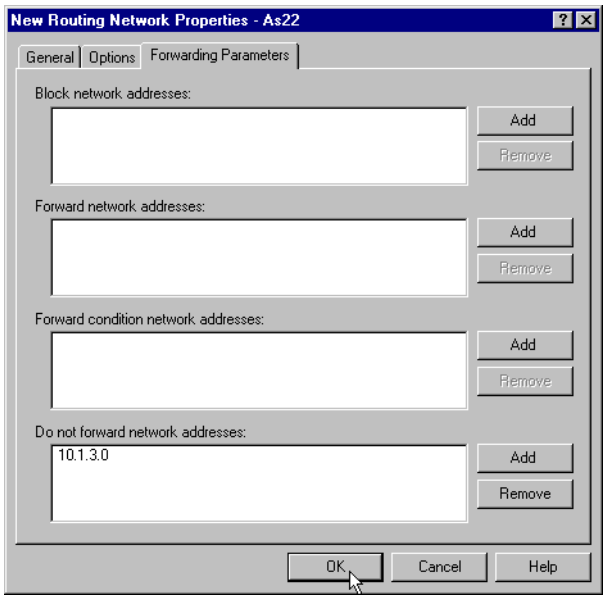


Figure 724. Entering the network that should be hidden behind 10.1.2.2/24

10. Click **OK** to confirm the “Do not forward network addresses” entry. The display shown in Figure 725 on page 590 appears.

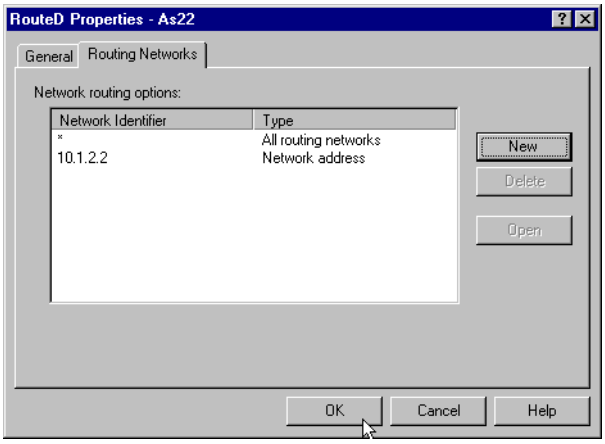


Figure 725. Confirming the configuration by clicking OK

11. Click **OK** to confirm the configuration. You are returned to the TCP/IP servers window of Operations Navigator (Figure 726).

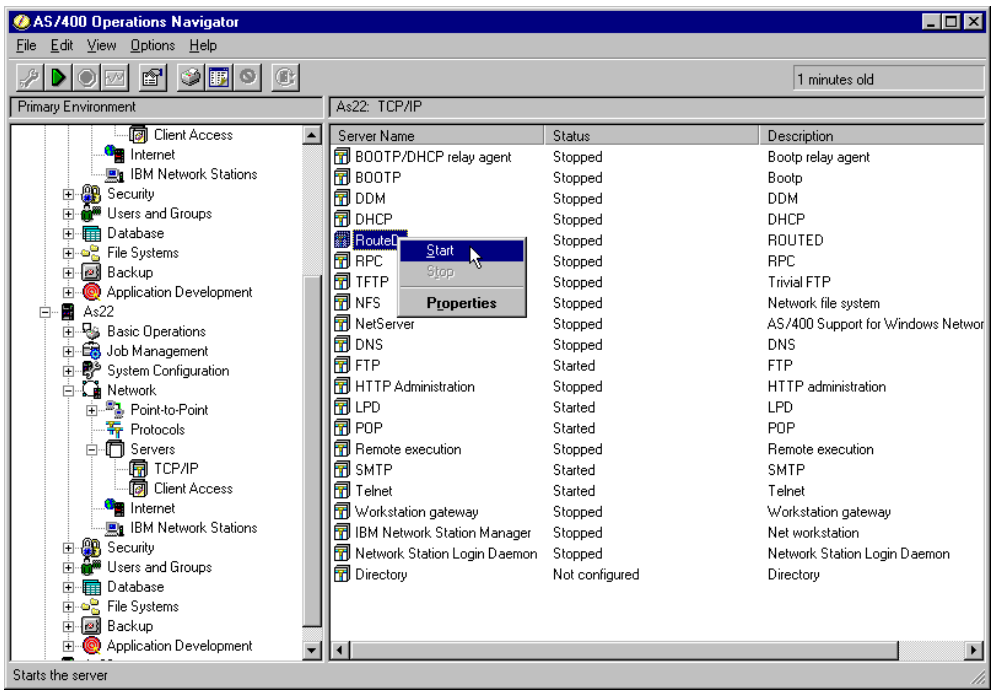


Figure 726. Starting the RouteD daemon to activate the RIP protocol

12. Right-click the **RouteD** icon, and select **Start** from the pop-up menu. This starts the RouteD daemon.

14.7.5.3 Testing the scenario

The scenario can be tested by issuing the Verify TCP/IP Connection (PING) command at AS21, to test the connection to the 10.1.3.1 interface at the AS22 system.

To view the routing entries at both systems, use the Work with TCP/IP Network Status (NETSTAT) command `NETSTAT OPTION(*RTE)`. This command shows the

dynamically added routing entries that the AS22 system receives from the AS21 system and vice versa. Please allow the systems to exchange the RIP messages. Wait two to three minutes for the routing tables to become stable.

Figure 727 shows the routing table at the AS21 system. No RIP routes are added to this system from the 10.1.3.0/24 network, since AS22 does not forward information about this network.

```
                    Display TCP/IP Route Information
                                                          System:   AS21
Type options, press Enter.
  5=Display details

     Route            Subnet           Next              Route
Opt  Destination      Mask             Hop             Available
     10.1.2.0         255.255.255.0    *DIRECT           *YES
     10.1.1.0         255.255.255.0    *DIRECT           *YES
     10.1.2.0         255.255.255.0    10.1.2.2          *YES
     127.0.0.0        255.0.0.0        *DIRECT           *YES

F3=Exit   F5=Refresh     F6=Print list   F9=Command line
F11=Display route type   F12=Cancel      F13=Sort by column   F24=More keys
```

Figure 727. Routing table on the AS21 system

Figure 728 shows the routing table at the AS22 system. The complete network is visible from AS22, since AS21 does not hide any of the networks.

```
                    Display TCP/IP Route Information
                                                          System:   AS22
Type options, press Enter.
  5=Display details

     Route            Subnet           Next              Route
Opt  Destination      Mask             Hop             Available
     10.1.3.0         255.255.255.0    *DIRECT           *YES
     10.1.2.0         255.255.255.0    *DIRECT           *YES
     10.1.2.0         255.255.255.0    10.1.2.1          *YES
     10.1.1.0         255.255.255.0    10.1.2.1          *YES
     127.0.0.0        255.0.0.0        *DIRECT           *YES

F3=Exit   F5=Refresh     F6=Print list   F9=Command line
F11=Display route type   F12=Cancel      F13=Sort by column   F24=More keys
```

Figure 728. Routing table on the AS22 system

Testing if access from AS21 to the 10.1.3.0/24 networks is possible is easily done by using the Verify TCP/IP Connection (PING) command. Note that the connection is made to the 10.1.3.0/24 network, the hidden interface of AS22. See Figure 729 and Figure 730 on page 592.

```
MAIN                          AS/400 Main Menu
                                                          System:   AS21
Select one of the following:

     1. User tasks
     2. Office tasks
     3. General system tasks
     4. Files, libraries, and folders
     5. Programming
     6. Communications
     7. Define or change the system
     8. Problem handling
     9. Display a menu
    10. Information Assistant options
    11. Client Access/400 tasks

    90. Sign off

Selection or command
===> PING RMTSYS('10.1.3.1')

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=Information Assistant
F23=Set initial menu
```

Figure 729. Testing the connection using the PING command

```
                          Display All Messages
                                                          System:   AS21
Job . . :   QPADEV0002    User . . :   ITSCID29      Number . . . :   001106

    No response from host within 1 seconds for connection verification 5.
    Connection verification statistics: 0 of 5 successful (0 %).
3 > wrkjob
3 > PING RMTSYS('10.1.3.1')
    Verifying connection to host system 10.1.3.1.
    No response from host within 1 seconds for connection verification 1.
    No response from host within 1 seconds for connection verification 2.
    No response from host within 1 seconds for connection verification 3.
    No response from host within 1 seconds for connection verification 4.
    No response from host within 1 seconds for connection verification 5.
    Connection verification statistics: 0 of 5 successful (0 %).
                                                                      More...
Press Enter to continue.

F3=Exit   F5=Refresh   F12=Cancel   F17=Top   F18=Bottom
```

Figure 730. No access to the 10.1.3.0/24 network: Hiding network successful

Chapter 15. Using virtual IP addresses

This chapter explains the use of virtual IP addresses. It provides information about how to configure virtual IP addresses on the AS/400 system.

15.1 What a virtual IP address is

A virtual IP address, or circuitless connection, is an IP interface that is defined on the system without being associated with a physical hardware adapter. These addresses can always be active on the system. These addresses can be used as the “system” IP address. These addresses are always reached indirectly through a real TCP/IP interface and do not respond to Address Resolution Protocol (ARP) requests. For other systems to reach the virtual IP address, they must have a route defined to reach the address. The AS/400 system accepts IP packets on any interface and processes the packet if the IP address is defined on any interface on the system. This provides a way to assign one or more addresses to the system without needing to bind the address to a physical interface. This can be used when you want to run multiple occurrences of a Domino Web server bound to different addresses, or other services, such as HTTP servers, that need to bind to default ports.

Virtual IP address support was added in V4R3 of the OS/400 operating system. This feature can be used when consolidating multiple systems into one large system.

15.2 Configuring virtual IP addresses

The addresses that you set up as virtual IP addresses cannot be a part of any real network segment in your network. Choose a network address range that is unused in your environment. Figure 731 shows a sample network that we discuss in this section.

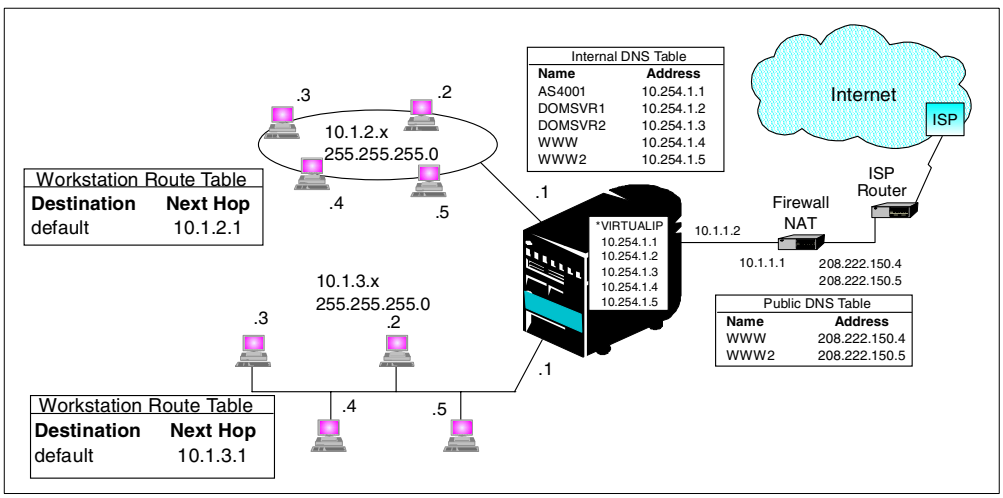


Figure 731. Sample network using virtual IP addresses

In this example, the workstations all point to the AS/400 system as their next hop gateway. The firewall uses Network Address Translation (NAT) to change the

public address 208.222.150.4 to 10.254.1.4, and the public address 208.222.150.5 to 10.254.1.5. The firewall has a route entry that directs all "10." traffic to AS/400 interface 10.1.1.2. When a packet arrives at the AS/400 system, it goes through the packet processing. If the destination address matches any address defined on the system (including virtual IP addresses), the system processes the packet.

15.2.1 Task summary

To set up the virtual IP address, we perform the following steps:

1. Select a network address to use as virtual IP addresses.
2. Define the virtual IP addresses on the system.
3. Add the route entries to any systems that need to access the system using the virtual IP addresses.
4. Add the system names to the DNS.
5. Start the interfaces that have the virtual IP address defined.
6. Test the connectivity.

15.2.2 Selecting a network address to use as virtual IP addresses

The first step in configuring virtual IP addresses is to select an address range to use as virtual IP addresses. This subnet of addresses must not be used anywhere else in the network. The address range cannot be a part of an existing subnet. The addresses in this range cannot respond to an ARP request. In our example, we selected a range that should be well out of the way of the rest of the network. Our sample network shown in Figure 731 on page 593 uses the 10. network for private addresses. After checking our network documentation, we determined that the 10.255.1 subnet was not in use. Because it is such a high address range, it should be out of the way of future growth.

15.2.3 Defining the virtual IP addresses on the system

Once you determine the address range that you are going to use, you need to create the TCP/IP interfaces on the system that will use the addresses. To add the virtual IP addresses on the system, you can either use the AS/400 command ADDTCPIFC, or you may use Operations Navigator. To add the first interface, we entered the command:

```
ADDTCPIFC INTNETADR('10.254.1.1') LIND(*VIRTUALIP) SUBNETMASK(*HOST)
MTU(16388)
```

We specified a subnet mask of *HOST so that we can use other addresses in the network range as virtual addresses on other systems. You must also specify a Maximum Transmission Unit (MTU) size because there is not a physical line description for the command to use to determine the frame size. The MTU size does not impact performance, because the interface is virtual. The route and physical interface taken out of the system determines the real MTU size.

To add the second interface, we used Operations Navigator. Refer to Figure 732 as you perform the following procedure to start the add process:

1. Double-click the system name **AS07** (A).
2. Double-click **Network** (B).

3. Click **Protocols** (C).
4. In the right window, right-click **TCP/IP**, and select **New Interface->Circuitless** (D). An information screen displays (not shown). Click **Next**. The display shown in Figure 733 on page 596 appears without the completed values.

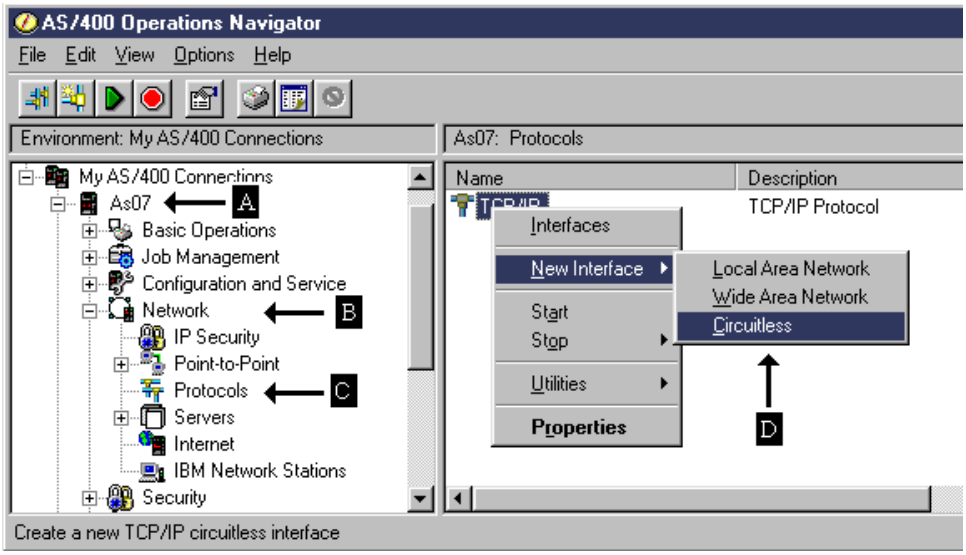


Figure 732. Adding a virtual IP address

5. Enter the virtual IP address, a name for the interface (used in Operations Navigator only), and the subnet mask value as shown in Figure 733 on page 596. Click **Next**. The Start TCP/IP Interface window (not shown) is displayed. Select your start values for the interface, and click **Next**. The New TCP/IP Interface Summary window appears (Figure 734 on page 596).

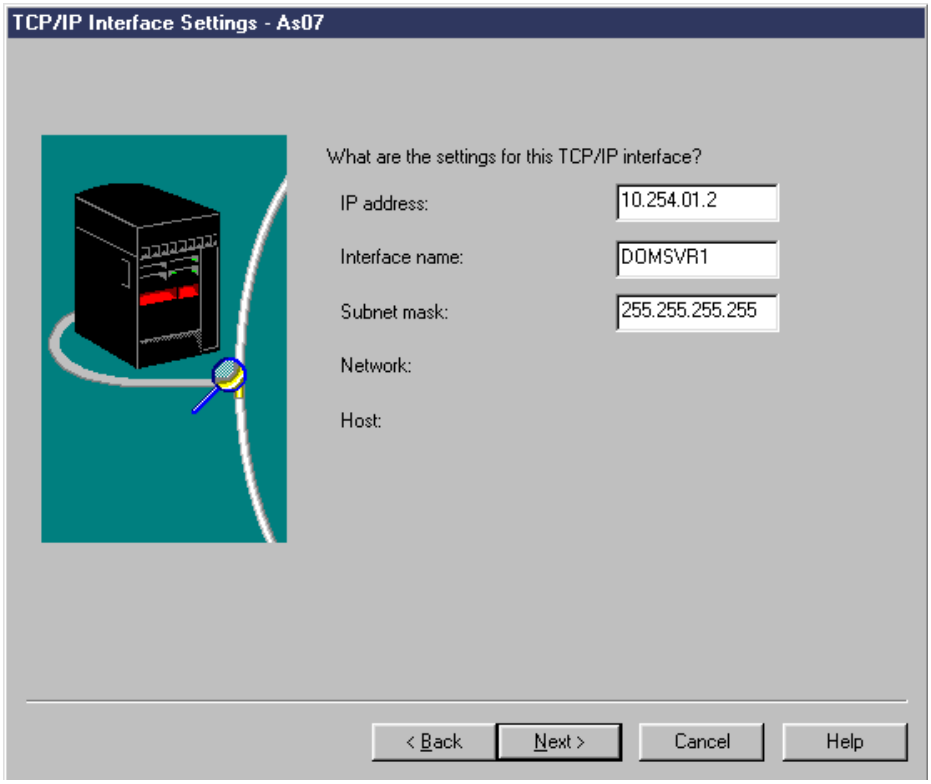


Figure 733. Specifying the IP address information

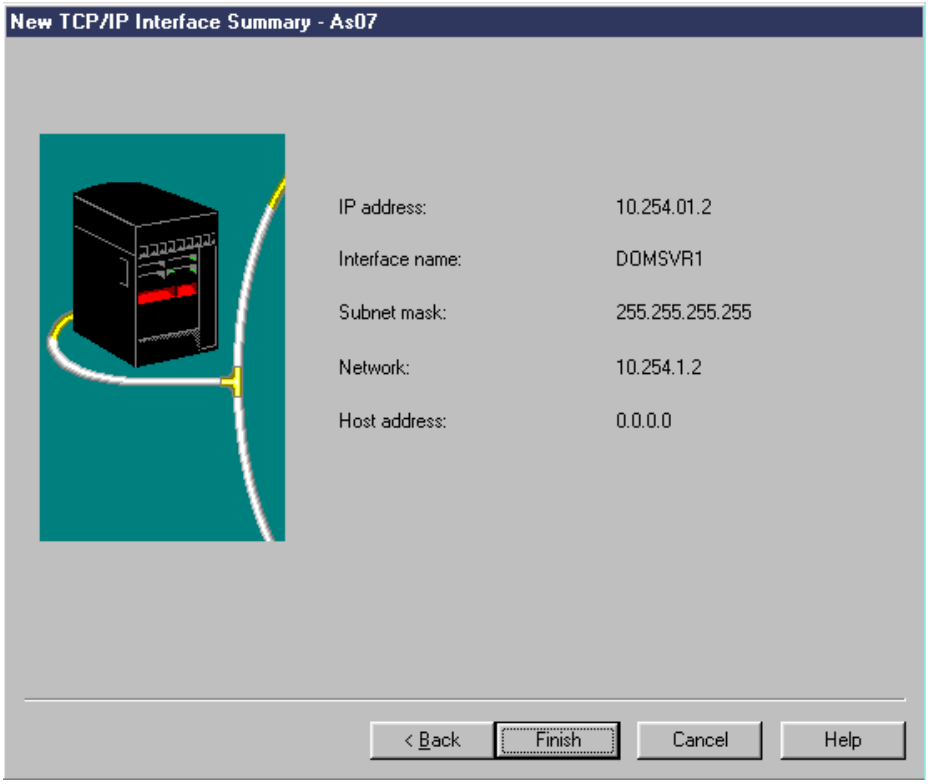


Figure 734. New TCP/IP Interface Summary display

6. Verify that the values shown are correct. If they are correct, click **Finish**. If they are incorrect, use the **Back** button to backup and correct the error. When you click **Finish**, you are given an opportunity to test the interface. Click **Test now**. After the test, click **OK** to exit the add function.

The virtual IP address is now added. Repeat the steps as needed to add the remaining IP addresses. Figure 735 shows the Work with TCP/IP Interfaces (`CFGTCP` option `1`) display after all the adds are complete for the interfaces.

```
                        Work with TCP/IP Interfaces
                                                              System:   AS07
Type options, press Enter.
  1=Add   2=Change   4=Remove   5=Display   9=Start   10=End

     Internet         Subnet             Line      Line
Opt  Address          Mask             Description  Type

 __
 __  10.1.1.2         255.255.255.0    ETHLINE1     *ELAN
 __  10.1.2.1         255.255.255.0    TRNLINE      *TRLAN
 __  10.1.3.1         255.255.255.0    ETHLINE2     *ELAN
 __  10.254.1.1       255.255.255.255  *VIRTUALIP   *NONE
 __  10.254.1.2       255.255.255.255  *VIRTUALIP   *NONE
 __  10.254.1.3       255.255.255.255  *VIRTUALIP   *NONE
 __  10.254.1.4       255.255.255.255  *VIRTUALIP   *NONE
 __  10.254.1.5       255.255.255.255  *VIRTUALIP   *NONE


                                                                    More...
F3=Exit      F5=Refresh   F6=Print list   F11=Display interface status
F12=Cancel   F17=Top      F18=Bottom
```

Figure 735. TCP/IP interfaces with all addresses added

15.2.4 Adding the route entries

Go to each system that needs to access the virtual IP addresses and add the correct routing entry. For most of the systems, this will consist of a default route with a next hop that points to the real AS/400 interface. In some cases, a more specific route entry may be needed.

This information is added to a Windows workstation by specifying a gateway entry in the TCP/IP properties of the network configuration. The information can also be passed in the DHCP configuration information that is passed to a workstation that is using DHCP to determine its TCP/IP address. Using DHCP is the recommended approach, because it puts all the TCP/IP configuration for the workstations in a central location.

You need to make the appropriate route entries in any routers that need to point to these virtual IP addresses. You may distribute the route to the virtual IP addresses to other systems and routers in the network using RIPV2. This is done by starting the router daemon on the AS/400 system using the command:

```
STRTCPSVR SERVER(*ROUTED)
```

15.2.5 Adding the system names to the DNS

Add the system names to the DNS server in the internal network. Refer to 8.9, “Setting up a simple DNS server” on page 306, for an example of the DNS

configuration. For detailed instructions, refer to *AS/400 TCP/IP Autoconfiguration: DNS and DHCP Support,* SG24-5147.

15.2.6 Starting the interfaces

If you did not choose to start the TCP/IP interfaces as you created them, you need to start them now. To start the interfaces, go to the Work with TCP/IP Interfaces (`CFGTCP` option `1`) as shown in Figure 735 on page 597. Enter a `9` in the option area in front of each interface you want to start. Press Enter. The interfaces should start.

15.2.7 Testing the connectivity

After all the interfaces are started, test the connectivity to ensure that everything is working. The easiest way to do this is by using the `PING` command.

Go to a workstation and PING one of the virtual IP addresses. You should receive a successful completion message. If you do not, check the route information in the workstation and any routers that may be in the network. If the PING works using an address, try it using the name you assigned to the virtual IP address. If the PING by address worked, but the PING by name fails, you have a DNS problem.

Repeat the PING test for each virtual address you defined. Perform a PING using the address first and then a PING using the name of each interface.

After all the PING tests in the internal network work, go to the external network and try accessing a server. A PING test may not work from the outside, because most firewalls block the PING command.

15.3 Virtual IP addresses and e-mail

Virtual IP addresses work well in situations where you need unique TCP/IP addresses to bind to applications. One example of this is when you set up multiple Domino servers on the same system. The recommendation is to define a new address for each new server. While you can add multiple IP addresses to a physical interface, this can lead to problems at times when a request comes in with one address but is responded to with another address.

Another problem can result. If the physical interface is varied off, the IP addresses associated with the interface are not available. With a virtual IP address, the interface can be active as long as the system is active. This may result in higher availability.

Chapter 16. OS/400 multicasting support

This chapter covers the basics of multicasting and how the AS/400 system can be used in this environment. It also introduces the multicasting in general, describing the use of multicasting and the IGMP protocol. The use of multicasting on the AS/400 system is shown using the RIP server (RouteD), which currently is the only application that supports multicasting. Other applications can be created by developers. This chapter contains information and an example of a multicast application.

Please refer to *TCP/IP Tutorial and Technical Overview,* GG24-3376, for detailed information on multicasting in general.

16.1 Introduction to multicasting

To understand multicasting, this section provides an introduction to multicasting support in general.

16.1.1 Unicast, broadcast, and multicast

The transmission of IP datagrams from one system to another system or several other systems can be done in several ways:

- Unicast
- Broadcast
- Multicast

Each of the three ways of transmitting datagrams is described in the following sections. Figure 736 shows the different transmission types.

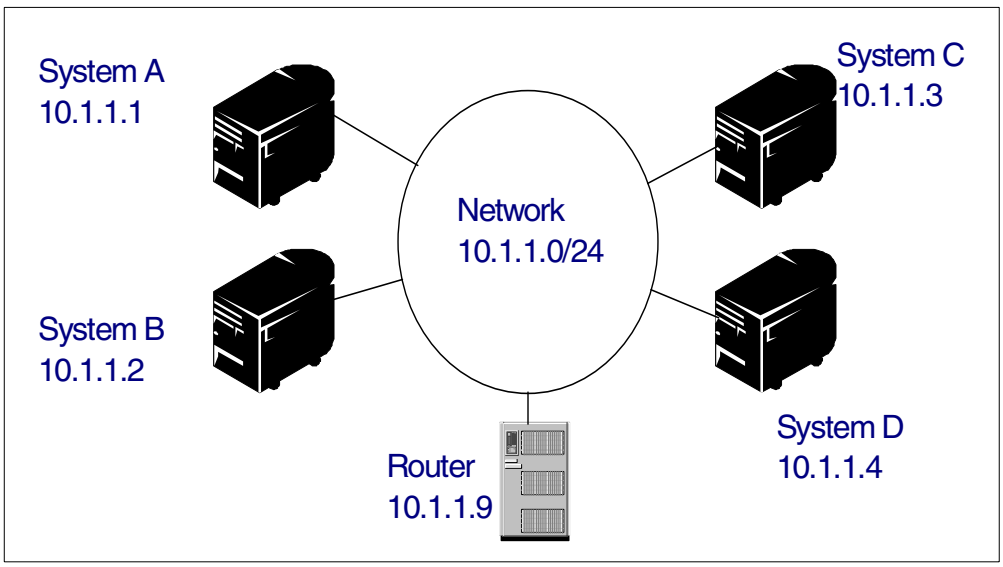


Figure 736. A Simple network using Unicast, Multicast, and Broadcast

16.1.1.1 Unicast

If System A wants to send an IP datagram to System B, the datagram is sent using unicasting, only *one* receiver of the datagram. A typical example is an

Telnet session between two systems. The source IP address is the System A (10.1.1.1), and the destination address is the System B (10.1.1.2).

16.1.1.2 Broadcast

If System A wants to send an IP datagram to all the systems on the 10.1.1.0/24 network, the datagram is sent using broadcasting. An example is when RIP v1 routing tables are propagated across the network. The routing table from System A is sent to all hosts on the subnet. Depending on what kind of broadcasting is wanted, four types of broadcasting exist:

- Limited broadcast address
- Network directed broadcast address
- Subnet directed broadcast address
- All-subnet-directed broadcast

Limited broadcast address

The address 255.255.255.255 (all bits “1” in all parts of the IP address). All hosts on the local network will recognize this address, but routers will not forward it. BOOTP and DHCP are an exception to this. Routers can be configured to forward BOOTP and DHCP broadcasts packets, a BOOTP/DHCP relay agent.

Network directed broadcast address

If the network number is a valid network number, the network is not subnetted, and the host number is all ones (for example, 128.2.255.255), the address refers to all the hosts on the specified network (for example, 128.2.0.0). Routers should normally forward these packets.

Subnet directed broadcast address

If the network number is a valid network number, the subnet is a valid subnet number, and the host is all ones, the address refers to all hosts on the specified subnet. The actual broadcast is performed by the router, which receives the datagram into the subnet.

All-subnet-directed broadcast

If the network is a valid number, the network is subnetted, and the local part is all ones (for example, 128.2.255.255), the address refers to all hosts on all subnets in the specified network. In principle, routers may propagate broadcasts for all subnets, but are not required to do so. In practice, they do not. There are few circumstances where such a broadcast would be desirable.

16.1.1.3 Multicast

Unfortunately, all the broadcast types have one major disadvantage. Apart from network selection, there is no way to select which hosts will receive the broadcast. All hosts on the subnet will receive it, and have to process it. The received frames travel up the protocol stack to the relevant level, where the frame possibly is discarded if the particular host is not interested in the data. This causes a significant overhead in the hosts that spend unnecessary time processing unwanted frames. Multicasting avoids this overhead by using preassigned groups of IP addresses: Class D addresses.

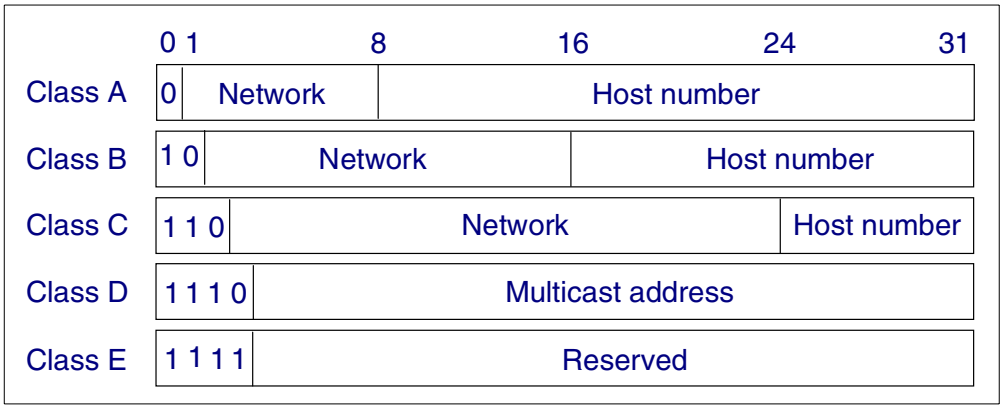


Figure 737. The IP address classes: Class D is multicast addresses

Multicast group addresses are in the range of 224.0.0.0 to 239.255.255.255 (class D in Figure 737). For each multicast address, there is a set of zero or more hosts listening to it. This set is called the *host group*. Multicasting is defined in RFC 1112 and IP addresses are defined in RFC1166.

Any host can send packets to a host group. There are no requirements for a host to be a member of a group to send a packet to that group.

The addition of IP multicasting enables new types of application solutions that take advantage of the benefits associated with a one-to-many type of transmission. For example, consider a chain of retail stores where there are hundreds or even thousands of stores with an AS/400 system in each store. Now suppose that the headquarters wants to distribute an updated price list file to each store. Instead of transferring the file several times, once for each store, a multicast solution would enable the AS/400 system to transmit the updated file a single time, with each store receiving it simultaneously.

Please note that only the UDP transport protocol supports multicasting, since UDP is a connectionless protocol. The TCP transport protocol is a connection-oriented protocol and, therefore, only supports unicasting.

16.1.2 Host groups

One host can be a member of one or more groups, as illustrated in Figure 738 on page 602. In the example, host 1 and 2 are members of group A; host 2, 3, and 4 are members of group B; and host 4 and 5 are members of group C. Host 2 and 4 are both members of two groups, while host 1, 3, and 5 are members of one group. The group names A through C are only used for simplicity. In real life, the group name will be a class D IP address.

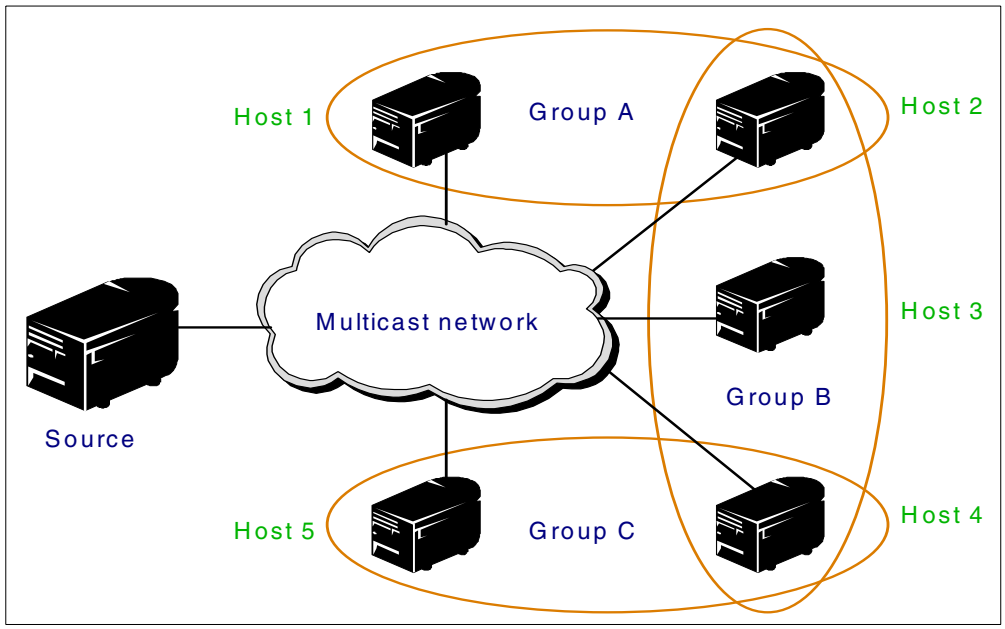


Figure 738. Multicasting network with several host groups

Multicasting on a single physical network is simple. To join a group, a process running on a host must somehow inform its network device drivers that it wants to be a member of the specified group. The device driver itself then maps the multicast address to the physical address and informs the adapter to enable the reception for that address.

16.1.2.1 Permanent and transient addresses

The IP class D host groups can be permanent or transient. The following sections describe both.

Permanent address

A permanent group has a well-known, administratively assigned IP address. The addresses shown below are part of the list of addresses out of the permanent group. The complete and up-to-date list can be found in RFC1700. A permanent group exists even if it has no members. However, the membership of a host group is not permanent. The host may leave or join the group at will.

Examples of permanent addresses (from RFC1700):

```
224.0.0.0  Base Address (Reserved)                  [RFC1112,JBP]
224.0.0.1  All Systems on this Subnet               [RFC1112,JBP]
224.0.0.2  All Routers on this Subnet                       [JBP]
224.0.0.3  Unassigned                                       [JBP]
224.0.0.4  DVMRP    Routers                         [RFC1075,JBP]
224.0.0.5  OSPFIGP  OSPFIGP All Routers            [RFC1583,JXM1]
224.0.0.6  OSPFIGP  OSPFIGP Designated Routers     [RFC1583,JXM1]
224.0.0.7  ST Routers                              [RFC1190,KS14]
224.0.0.8  ST Hosts                                [RFC1190,KS14]
224.0.0.9  RIP2 Routers                                 [GSM11]
224.0.0.10 IGRP Routers                        [Dino Farinacci]
224.0.0.11 Mobile-Agents                         [Bill Simpson]
224.0.0.12-224.0.0.255 Unassigned                           [JBP]

224.0.1.0  VMTP Managers Group                     [RFC1045,DRC3]
224.0.1.1  NTP      Network Time Protocol          [RFC1119,DLM1]
224.0.1.2  SGI-Dogfight                                    [AXC]
224.0.1.3  Rwhod                                           [SXD]
224.0.1.4  VNP                                          [DRC3]
```

```
224.0.1.5  Artificial Horizons - Aviator                    [BXF]
224.0.1.6  NSS - Name Service Server                       [BXS2]
224.0.1.7  AUDIONEWS - Audio News Multicast                [MXF2]
224.0.1.8  SUN NIS+ Information Service                    [CXM3]
224.0.1.9  MTP Multicast Transport Protocol                 [SXA]
224.0.1.10 IETF-1-LOW-AUDIO                                [SC3]
224.0.1.11 IETF-1-AUDIO                                    [SC3]
224.0.1.12 IETF-1-VIDEO                                    [SC3]
224.0.1.13 IETF-2-LOW-AUDIO                                [SC3]
224.0.1.14 IETF-2-AUDIO                                    [SC3]
224.0.1.15 IETF-2-VIDEO                                    [SC3]
224.0.1.16 MUSIC-SERVICE                        [Guido van Rossum]
224.0.1.17 SEANET-TELEMETRY                        [Andrew Maffei]
224.0.1.18 SEANET-IMAGE                            [Andrew Maffei]
224.0.1.19 MLOADD                                         [Braden]
224.0.1.20 any private experiment                           [JBP]
224.0.1.21 DVMRP on MOSPF                               [John Moy]
224.0.1.22 SVRLOC                               <veizades@ftp.com>
224.0.1.23 XINGTV                                 <hgxing@aol.com>
224.0.1.24 microsoft-ds                     <arnoldm@microsoft.com>
224.0.1.25 nbc-pro                        <bloomer@birch.crd.ge.com>
224.0.1.26 nbc-pfn                        <bloomer@birch.crd.ge.com>
224.0.1.27-224.0.1.255  Unassigned                          [JBP]
224.0.2.1  "rwho" Group (BSD) (unofficial)                  [JBP]
224.0.2.2  SUN RPC PMAPPROC_CALLIT                         [BXE1]
224.0.3.000-224.0.3.255 RFE Generic Service                [DXS3]
224.0.4.000-224.0.4.255 RFE Individual Conferences         [DXS3]
224.0.5.000-224.0.5.127 CDPD Groups                  [Bob Brenner]
224.0.5.128-224.0.5.255 Unassigned                          [IANA]
224.0.6.000-224.0.6.127 Cornell ISIS Project           [Tim Clark]
224.0.6.128-224.0.6.255 Unassigned                          [IANA]
224.1.0.0-224.1.255.255  ST Multicast Groups        [RFC1190,KS14]
224.2.0.0-224.2.255.255  Multimedia Conference Calls         [SC3]
224.252.0.0-224.255.255.255 DIS transient groups     [Joel Snyder]
232.0.0.0-232.255.255.255  VMTP transient groups    [RFC1045,DRC3]
```

Transient address

Any group that is not permanent is transient and is available for dynamic assignment as needed. Transient groups use addresses that are not reserved for permanent groups. Transient groups exist only as long as they have members.

16.1.2.2 Link layer considerations

The link layer of the protocol stack is responsible for transmitting the IP datagrams across the network. Two functions are performed at the link layer. One function maps the IP destination address (a multicast address) to a corresponding multicast MAC address (Ethernet, Token-Ring, FDDI). The second function is a hardware filtering mechanism, which is desirable in the link layer to have an efficient implementation. Otherwise, the host would have to examine each datagram and perform the filtering.

At the link layer, the IP multicast group destination address is mapped to a link layer multicast address. For point-to-point networks, such as PPP and the loopback interface, there is no mapping needed, since there is only one possible destination.

The following mapping for the various network physical layers is supported:

- **Ethernet/FDDI**

  For Ethernet/FDDI, a 4-byte IP multicast address is mapped to a 6-byte Ethernet/FDDI destination MAC group address by placing the low order 23 bits of the IP address into the low order 23 bits of the group address 01-00-5E-00-00-00. For example, 224.255.0.2 maps to 01-00-5E-7F-00-02. Further information can be found in RFC1060 and RFC1188.

- **Token-Ring**

  For Token-Ring, due to the current implementation of controller chips, it is not as straightforward. Due to these limitations, mapping the IP multicast address to the Token-Ring destination MAC address is done by mapping all IP multicast addresses to the Token-Ring C0:00:00:04:00:00 functional address. RFC1469 describes this in detail.

- **Packet-switched (X.25/Frame Relay)**

  Multicast does not map well with all link layers, such as X.25 and Frame Relay. For example, a switched network, such as X.25 or Frame Relay, does not lend itself to multicast applications, because there is no mechanism for transmitting a single packet to all systems in the network. This is basically the same as if you would pick up your phone, dial, and get a connection to all phones in the world.

16.1.2.3 Multicasting across several physical networks

Multicasting is not limited to a single physical network. There are two aspects to multicasting across physical networks:

- A mechanism for deciding how widespread the multicast is. Remember that, unlike unicast addresses and broadcast addresses, multicast addresses cover the entire TCP/IP network.
- A mechanism for deciding whether a multicast datagram needs to be forwarded to a particular network.

The first problem is easily solved. The multicast has a time to live (TTL) value like all other IP packets, which is decreased with each hop to a new network. When the TTL field reaches zero, the datagram can go no further, and will be discarded.

The second aspect, deciding whether a router should forward a multicast datagram is done using a combination of multicast routing protocols, such as MOSPF (multicast OSPF), MSDP (meast source distribution), and the Internet Group Management Protocol (IGMP).

16.1.3 Internet Group Management Protocol (IGMP)

RFC1112 defines three levels of multicasting support:

- **Level 0:** No support for multicasting
- **Level 1:** Support for sending, but not receiving, multicast IP datagrams
- **Level 2:** Full support for IP multicasting

Level 2 requires implementation of the IGMP protocol. OS/400 V4R2 supports level 2. In short terms, the IGMP protocol works in the following way.

To join a group, the host sends a report on an interface. The report is addressed to the multicast group of interest. Multicast routers on the same subnet receive the report and set a flag to indicate that at least one host on that subnet is a member of that group (Figure 739). No host has to join all host groups (224.0.0.1). Membership in this group is automatic. Multicast routers have to listen to all multicast addresses, that is all groups, to detect such reports.

Multicast routers regularly, but infrequently, send out a query to the all hosts multicast address. Each host who still wishes to be a member of one or more

groups replies once for each group of interest. Each reply is sent after a random delay to ensure that IGMP does not cause burst traffic on the network.

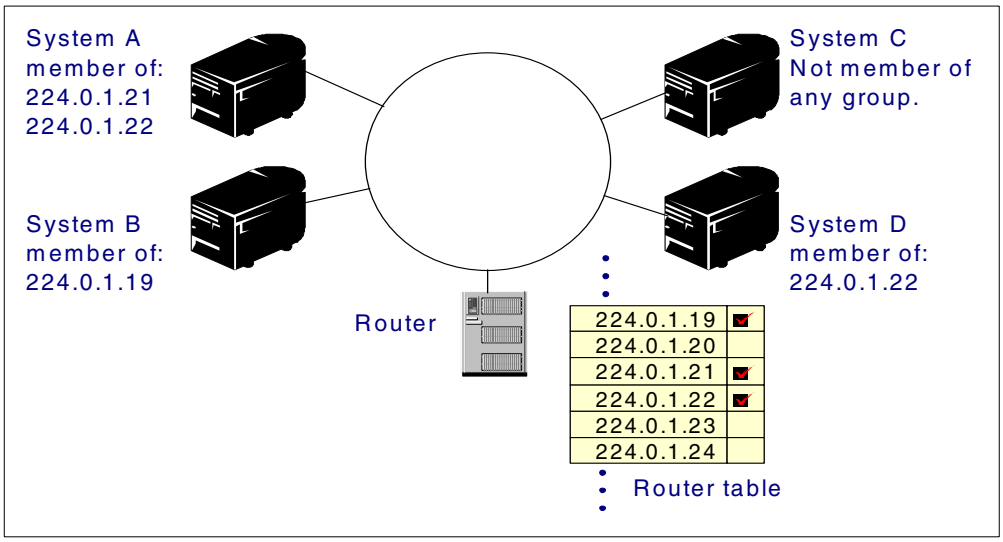


Figure 739. IGMP table on the router in the multicast network

16.1.4 Multicast routers and multicast routing protocols

Since IGMP only performs the communication between the receiving hosts and the multicast router in the subnet, routing of the packets between multicast routers must be managed by a multicast routing protocol. There are several algorithms for packet routing between multicast routers, which are all described in *TCP/IP Tutorial and Technical Overview,* GG24-3376.

Note: The AS/400 system cannot act as a multicast-capable router.

16.2 OS/400 multicasting implementation

OS/400 V4R2 has full multicasting level 2 support. It allows for sending and receiving multicast IP datagrams, as well as joining and leaving host groups.

IP multicast on the AS/400 system for non-multicast capable links, such as X.25 and twinax, is not supported. IP multicasting is also not supported on Frame Relay, FDDI/SDDI, and ATM networks.

From the administrator point of view, multicasting is not very visible. The configuration using IGMP is done automatically. Only a few commands let you view the configuration. The multicast support is more interesting from a developers point of view.

16.2.1 Commands supporting multicasting

A few commands let you exploit the multicasting environment. The following sections offer a brief description of the commands.

16.2.1.1 Verify TCP/IP Connection (PING/VFYTCPCNN)

This command allows ICMP echo to a multicast group destination address and displays the information from multicast PING responses. The new parameter is

IPTTL (IP time to live). This gives the user control over how far (for example, number of hops) a ICMP echo packet will travel.

16.2.1.2 Add TCP/IP Route (ADDTCPRTE)

This command contains a new special value for the RTEDEST (Route destination) parameter. The value *DFTMCAST allows the addition of default multicast routes. The default multicast routes are used to select the interface, over which multicast packets should be sent when the application does not specify the interface.

16.2.1.3 Work with TCP/IP Network Status (WRKTCPSTS/NETSTAT)

This command now displays the multicast group associated with each interface. The format of the group display varies depending on the TCP/IP interface selected, hardware support, not multicast capable, etc.

```
                  Work with TCP/IP Network Status
                                                          System:   AS22
Select one of the following:

     1. Work with TCP/IP interface status
     2. Display TCP/IP route information
     3. Work with TCP/IP connection status









Selection or command
===> 1

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
```

Figure 740. The NETSTAT command main menu

To determine if an interface supports multicast, enter option `14` on the NETSTAT display (Figure 741). If the selected interface supports multicasting, there will be at least one host group entry for the All Hosts Group 224.0.0.1 (Figure 742 and Figure 744 on page 608). Otherwise, the interface does not support multicast (Figure 743 on page 608).

```
                Work with TCP/IP Interface Status
                                                              System:   AS22
Type options, press Enter.
  5=Display details   8=Display associated routes   9=Start   10=End
  12=Work with configuration status   14=Display multicast groups

     Internet         Network            Line      Interface
Opt  Address          Address          Description  Status
14   10.1.1.1         10.1.1.0         FSIOPETH01   Active
14   127.0.0.1        127.0.0.0        *LOOPBACK    Active
14   192.168.1.1      192.168.1.0      FSIOPTRN02   Active









                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F11=Display line information   F12=Cancel
F13=Sort by column    F24=More keys
```

Figure 741. Defined interfaces on the system: Accessing the multicast groups

```
                        Display Multicast Host Groups
                                                              System:   AS22
Interface internet address  . . . . . . . . . . :   10.1.1.1


Host Group       Hardware Address        Host Group       Hardware Address
224.0.0.1        01:00:5E:00:00:01









                                                                      Bottom
F3=Exit   F5=Refresh   F6=Print   F9=Command line   F11=Hide hardware address
F12=Cancel
```

Figure 742. All hosts multicast group and the Ethernet multicast address

```
                        Display Multicast Host Groups
                                                              System:   AS22
Interface internet address  . . . . . . . . . :   127.0.0.1


Host Group          Host Group          Host Group          Host Group

  (No multicast host groups)

                                                                        Bottom
F3=Exit   F5=Refresh   F6=Print   F9=Command line   F12=Cancel
```

*Figure 743. No multicast group are defined for the *LOOPBACK (127.0.0.1) interface*

```
                        Display Multicast Host Groups
                                                              System:   AS22
Interface internet address  . . . . . . . . . :   192.168.1.1


Host Group       Hardware Address       Host Group       Hardware Address
224.0.0.1        C0:00:00:04:00:00

                                                                        Bottom
F3=Exit   F5=Refresh   F6=Print   F9=Command line   F11=Hide hardware address
F12=Cancel
```

Figure 744. All hosts multicast group and the Token-Ring multicast address

16.2.2 OS/400 applications using multicasting

Starting with OS/400 V4R2, the only application making use of the multicasting support is RIP v2 (RouteD). Chapter 14, “Using routing with the AS/400 system” on page 519, has a detailed description of RIP. As shown in “Permanent address” on page 602, the RIP v2 protocol uses the 224.0.0.9 multicast host group. Figure 745 shows an example of a system where the RouteD server is running. Both the All Hosts group and the RIP v2 group are active on this interface.

```
                        Display Multicast Host Groups
                                                                 System:   AS22
Interface internet address  . . . . . :   192.168.1.1


Host Group       Hardware Address        Host Group       Hardware Address
224.0.0.1        C0:00:00:04:00:00
224.0.0.9        C0:00:00:04:00:00

                                                                          Bottom
F3=Exit   F5=Refresh   F6=Print   F9=Command line   F11=Hide hardware address
F12=Cancel
```

Figure 745. Multicast host groups assigned when RouteD is active

16.2.3 AS/400 hardware considerations

The older 2626 Token-Ring IOP requires manual configuration to receive multicast datagrams. In particular, the Token-Ring address C00000040000 must be specified on the functional address parameter for a Token-Ring line description. To add this address to a line description named TRNLINE, use the following command:

```
CHGLINTRN LIND(TRNLINE) FCNADR(C00000040000)
```

Notice that the line must be varied off to change the line description.

The 2617 Ethernet IOP also requires manual intervention to receive multicast datagrams. The Ethernet group addresses to be received need to be specified on the group address parameter (GRPADR for an Ethernet line description). A 4-byte IP multicast is mapped to a 6-byte Ethernet group address by placing the low order 23 bits of the IP multicast address into the low order 23 bits of the Ethernet group address 01005E000000. For example, to receive multicast datagrams with a destination address of 224.255.0.2, the GRPADR parameter for the 2617 Ethernet line description must include 01005E7F0002.

Table 31 on page 610 shows the level of IOP support for the various communication IOPs.

The Assist IOPs provide support for the following IP and IGMP protocol functions:

- Mapping the IP destination Class-D address to the physical layers destination multicast address (RFC1112, RFC1188, RFC1469). For Ethernet and FDDI, the low order 23 bits of the IP address are placed into the low order 23 bits of the Ethernet multicast address 01-00-5E-00-00-00. For Token-Ring, all IP multicast addresses are mapped into the Token-Ring functional address C0-00-00-04-00-00.

- Performing where appropriate a multicast packet filtering function. Incoming datagrams destined to groups to which the host does not belong are discarded without generating any error report or log entry. This requires the IOP to have knowledge of which multicast group addresses the host joins and leaves on a given interface. Note that IP multicast packets may arrive with a MAC-level destination address that is not a MAC level multicast address (for example, it may be broadcast or a directed interface). These packets should be processed as if they were sent to the MAC level multicast address. In other words, do not discard an IP multicast packet just because it was sent using something other than a MAC-level multicast address.
- Selecting the correct Ethernet standard based on the configuration of the line description for Ethernet lines. Multicast packets will either be transmitted as Ethernet version 2, IEEE 802.3, or both, depending on how the line is configured.
- The number of IP group addresses supported by an Assist IOP is limited to 32. If more than 32 groups are joined on any particular interface, the host informs the IOP that inbound multicast packet filtering should be disabled, at which point the filtering function is taken over by the host IP code.

Table 31. Hardware features and multicast support

Feature	Description	Multicast assist
2617	SPD Ethernet adapter	Assist
2619	SPD 16/4 Mbps Token-Ring Adapter/HP	Assist
2626	SPD 16/4 Mbps Token-Ring Adapter/A	Non Assist
2723	PCI Ethernet IOA	Assist
2724	PCI 16/4 Mbps Token-Ring IOA	Assist
2838	PCI 100/10 Mbps Ethernet IOA	Assist
6149	PCI 16/4 Mbps Token-Ring IOA	Assist
6181	PCI Ethernet IOA	Assist
9174	Ethernet IOA	Non Assist
9175	Token-Ring IOA	Non Assist
FSIOP-1		Assist
FSIOP-3		Assist
SLIP/PPP		Non Assist

16.3 Developing multicasting applications

Currently, only the RIP v2 multicast is used on the AS/400 system, but multicasting still enables you to make your own applications. This section provides you with basic programming information.

The socket APIs include the verbs setsockopt() and getsockopt() for setting and getting various socket options. These verbs have been enhanced to accept the new parameter values for IP multicast.

An application program can send or receive multicast datagrams by using the Sockets API and connectionless, SOCK_DGRAM type sockets. Multicasting is a one-to-many transmission method. You cannot use connection-oriented sockets of type SOCK_STREAM for multicasting. When a socket of type SOCK_DGRAM is created, an application can use the setsockopt() function to control the multicast characteristics associated with that socket. The setsockopt() function accepts the following IPPROTO_IP level flags:

- **IP_ADD_MEMBERSHIP**: Joins the multicast group specified.
- **IP_DROP_MEMBERSHIP**: Leaves the multicast group specified.
- **IP_MULTICAST_IF**: Sets the interface over which outgoing multicast datagrams should be sent.
- **IP_MULTICAST_TTL**: Sets the time to live (TTL) in the IP header for outgoing multicast datagrams.
- **IP_MULTICAST_LOOP**: Specifies whether a copy of an outgoing multicast datagram should be delivered to the sending host as long as it is a member of the multicast group.

For additional information about sockets, including some sample programs, refer to *Sockets Programming,* SC41-5422. The book *System API Reference,* SC41-5801, documents the sockets API.

Chapter 17. Configuration and use of REXEC

This chapter covers the set up of Remote Execution (REXEC) on the AS/400 system. It briefly describes the REXEC server and client protocol. Furthermore, the basic functions of managing and configuring REXEC on the AS/400 system are shown. Several scenarios show the AS/400 system as a REXEC server and REXEC client in relation to other platforms.

For further information on the REXEC function in general, refer to *TCP/IP Tutorial and Technical Overview,* GG24-3376.

17.1 Description of Remote Execution (REXEC)

REXEC is a protocol that allows execution of commands on remote systems across TCP/IP networks. Figure 746 shows the components used in the REXEC protocol.

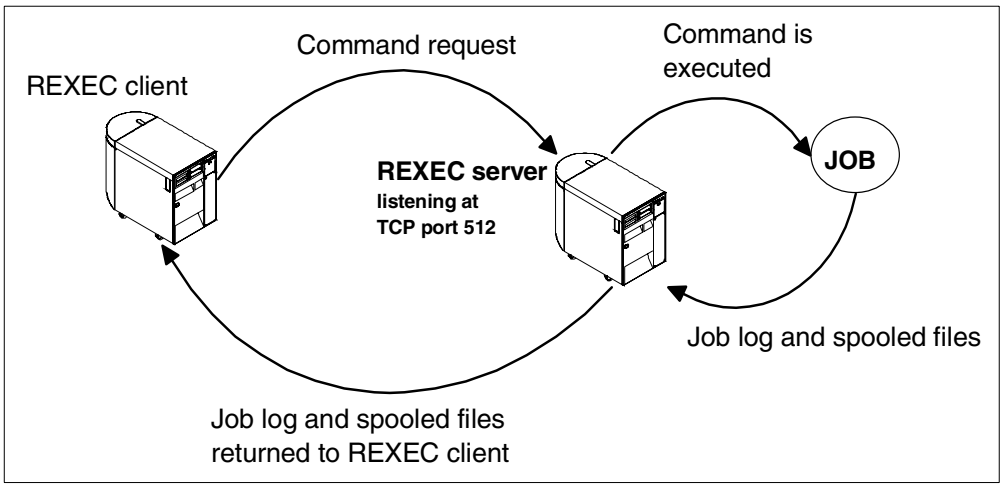


Figure 746. The REXEC protocol flow

The REXEC protocol uses TCP well-known port 512 at the server side, and the REXEC daemon will listen at this port.

The REXEC protocol works across different platforms. Figure 747 on page 614 shows an example of several platforms, acting both as REXEC servers and REXEC clients.

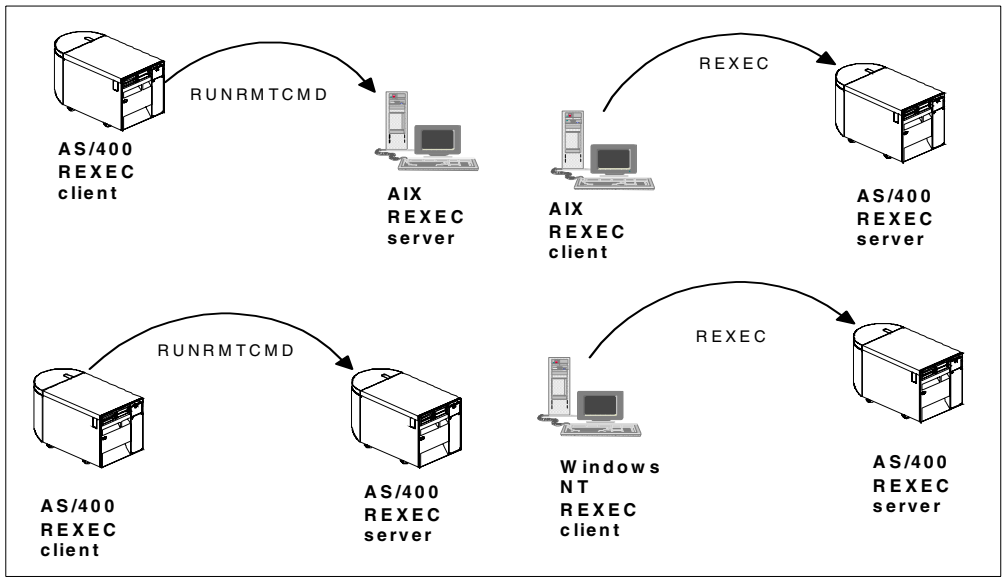


Figure 747. REXEC client and REXEC server communication

REXEC consists of two parts: a *server* and a *client*. The AS/400 system can act both as a server and as a client, connecting to different platforms in a heterogeneous network where TCP/IP is used as the network protocol.

The server part of REXEC, Remote Execution Daemon (REXECD*)*, executes the incoming commands. A valid user name and password must be specified to execute the requested command. Figure 748 shows these daemon jobs in the QSYSWRK subsystem. The daemon jobs are named QTRXCnnnnn.

```
                          Work with Active Jobs                    AS21
                                                        11/12/98  08:34:29
CPU %:     .0     Elapsed time:   00:00:00     Active jobs:   104

Type options, press Enter.
  2=Change   3=Hold   4=End   5=Work with   6=Release   7=Display message
  8=Work with spooled files   13=Disconnect ...

Opt  Subsystem/Job  User        Type  CPU %  Function        Status
       QTPOP00577   QTCP        BCH      .0                  DEQW
       QTPOP00624   QTCP        BCH      .0                  DEQW
       QTPOP00921   QTCP        BCH      .0                  TIMW
       QTRXC00477   QTCP        BCH      .0                  DEQW
       QTRXC00581   QTCP        BCH      .0                  SELW
       QTSMTPBRCL   QTCP        BCH      .0  PGM-QTMSBRDG     DEQW
       QTSMTPBRSR   QTCP        BCH      .0  PGM-QTMSBRSR     DEQW
       QTSMTPCLNT   QTCP        BCH      .0  PGM-QTMSCLCP     DEQW
       QTSMTPSRVR   QTCP        BCH      .0  PGM-QTMSSRCP     SELW
                                                                More...
Parameters or command
===>
F3=Exit   F5=Refresh       F7=Find      F10=Restart statistics
F11=Display elapsed data   F12=Cancel   F23=More options   F24=More keys
```

Figure 748. The REXEC server daemon: Jobs on the AS/400 system

The client part sends the command along with user identification and password. Figure 749 shows an example of the command.

```
                      Run Remote Command (RUNRMTCMD)

Type choices, press Enter.

Command  . . . . . . . . . . . . > 'DSPLIB LIB(COOL2RED)'

                                                                        ...
Remote location:
  Name or address  . . . . . . . > AS22


  Type . . . . . . . . . . . . . > *IP          *SNA, *IP
Remote user ID . . . . . . . . . > COOLUSER     Character value, *NONE...
Remote password  . . . . . . . . > COOLPASS     Character value, *NONE

                                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display       F24=More keys
```

Figure 749. The REXEC client on AS/400 system: RUNRMTCMD command

17.2 Managing REXEC

The REXEC functions at the client side requires no special configuration except from a working TCP/IP interface. The REXEC daemon at the server side has to be started to process the incoming commands.

17.2.1 Starting the REXEC daemon

The following section describes how the system administrator can start the REXEC daemon either by using the Operations Navigator interface or by using the green-screen interface.

17.2.1.1 Using the Operations Navigator

The REXEC server can be started using the Operations Navigator (Figure 750 on page 616):

1. To start Operations Navigator, click **Start->Programs->IBM AS400 Client Access->AS/400 Operations Navigator**. The AS/400 Operations Navigator window appears.
2. Double-click on the **AS/400 Network** (A).
3. Double-click the system icon (B) for the AS/400 system that you are configuring. The system components appear.
4. Double-click **Network** (C). The network components appear.
5. Double-click **Servers** (D). The available protocols appear.
6. Double-click **TCP/IP** (E). The available services appear in the right window.

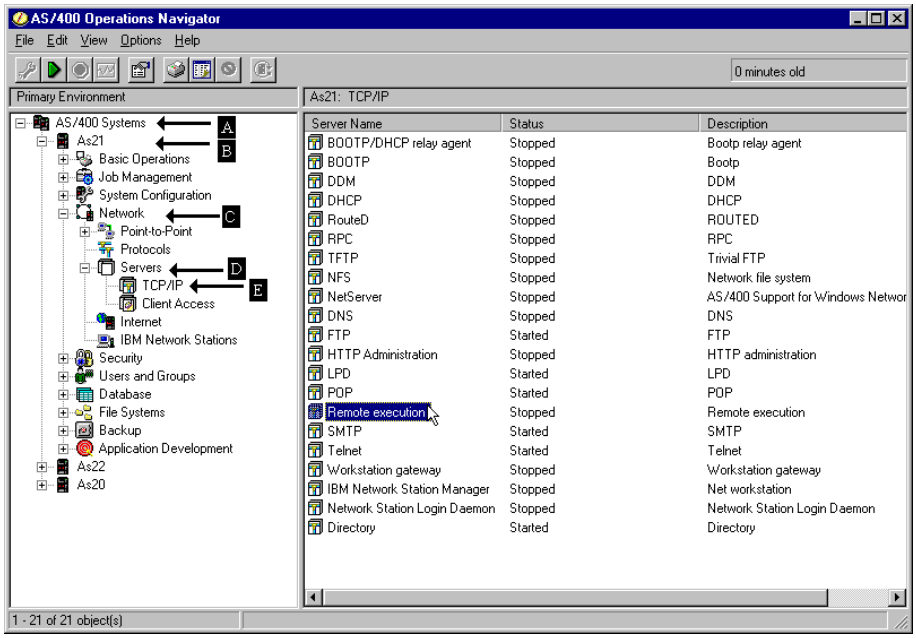


Figure 750. Starting the REXEC server using Operations Navigator (Part 1)

7. Right-click on **Remote execution**, and select **Start** from the pop-up menu as shown in Figure 751.

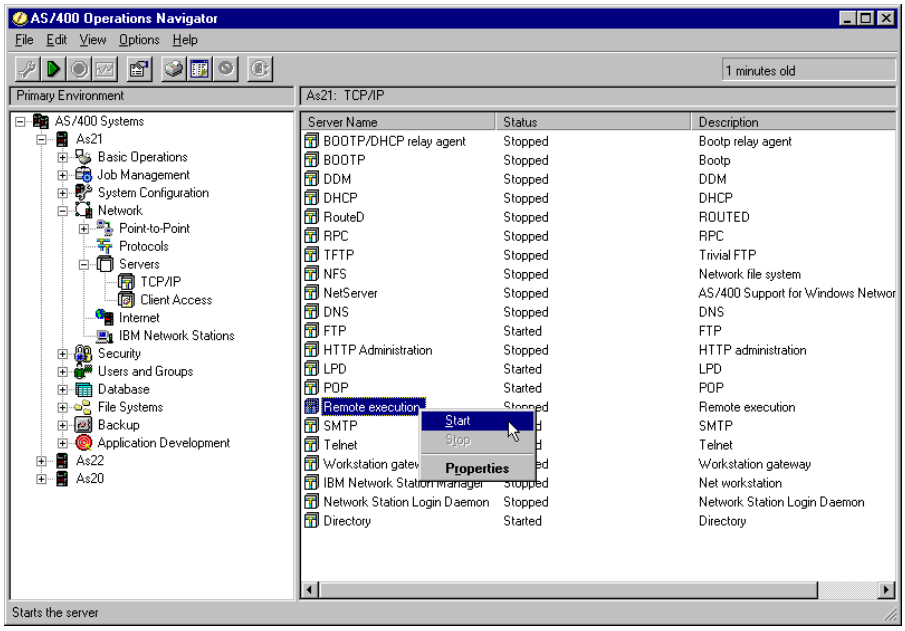


Figure 751. Starting the REXEC server using Operations Navigator (Part 2)

17.2.1.2 Using the green-screen interface

From the command entry, enter the Start TCP/IP Server (STRTCPSVR) command. Specify the REXEC server:

```
STRTCPSVR SERVER(*REXEC)
```

17.2.2 Ending the REXEC daemon

This section shows you how to end the REXEC daemon using both the Operations Navigator interface and the AS/400 command line interface.

17.2.2.1 Using Operations Navigator

The REXEC server can be stopped using the Operations Navigator (Figure 752). Follow these steps:

1. To start Operations Navigator, click **Start->Programs->IBM AS400 Client Access->AS/400 Operations Navigator**. The AS/400 Operations Navigator window appears.
2. Double-click on the AS/400 network (A).
3. Double-click the system icon (B) for the AS/400 system that you are configuring. The system components appear.
4. Double-click **Network** (C). The network components appear.
5. Double-click **Servers** (D). The available protocols appear.
6. Double-click **TCP/IP** (E). The available services appear in the right window.

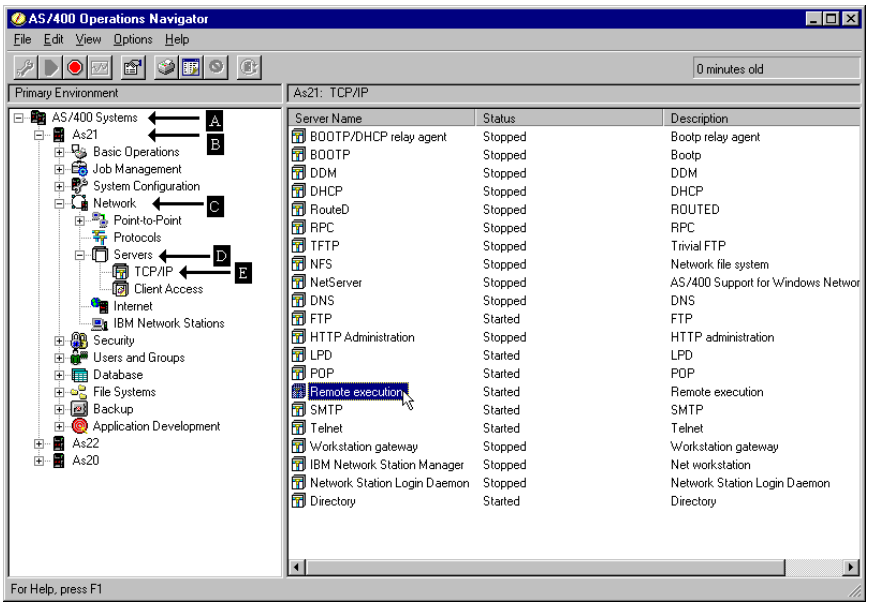


Figure 752. Stopping the REXEC server using Operations Navigator (Part 1)

7. Right-click with the mouse on the **Remote execution**, and select **Stop** from the pop-up menu as shown in Figure 753 on page 618.

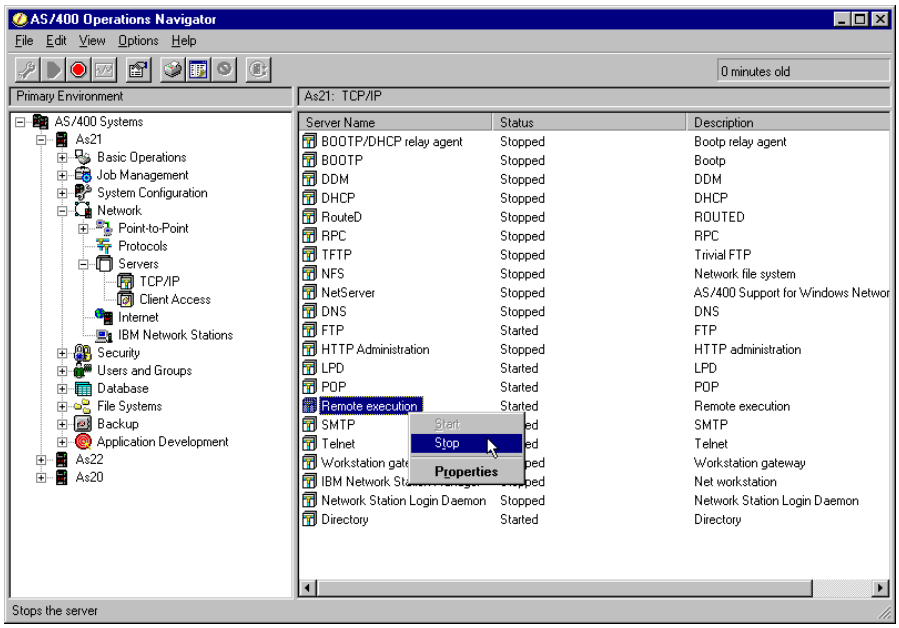


Figure 753. Stopping the REXEC server using Operations Navigator (Part 2)

17.2.2.2 Using the green-screen interface

From the command entry, enter the End TCP/IP Server (ENDTCPSVR) command, specifying the REXEC server:

```
ENDTCPSVR SERVER(*REXEC)
```

17.2.3 Checking that the REXEC daemon is running

The status of the REXEC daemon can easily be determined by using the Operations Navigator or the green-screen interface. Figure 750 on page 616 shows the Operations Navigator interface. Use the F5 key (Refresh) to update the display with the current status of the daemon. As shown in Figure 748 on page 614, the Work with Active Jobs (WRKACTJOB) command can also be used to determine the status of the daemon.

However, there is an additional way of viewing the status of the REXEC daemon. This applies not only to the REXEC daemon, but also to other TCP/IP services, such as FTP, WWW, Telnet, SMTP, POP3, etc. Viewing the status can be done using the Work with TCP/IP Connection Status (NETSTAT) command. From the command line, enter:

```
NETSTAT OPTION(*CNN)
```

The display in Figure 754 is shown.

```
                    Work with TCP/IP Connection Status
                                                                  System:   AS21
Local internet address  . . . . . . . . . . . :   *ALL

Type options, press Enter.
  4=End   5=Display details

     Remote           Remote     Local
Opt  Address          Port       Port       Idle Time  State
     *                *          ftp-con >  002:22:12  Listen
     *                *          telnet     024:44:28  Listen
     *                *          telnet     000:21:14  Listen
     *                *          smtp       024:45:16  Listen
     *                *          pop3       024:45:13  Listen
     *                *          snmp       023:40:49  *UDP
     *                *          as-svrmap  000:19:14  Listen
     *                *          exec       017:02:38  Listen
     *                *          lpd        024:45:44  Listen
     *                *          1030       023:40:49  *UDP
     *                *          as-cent >  000:20:10  Listen
                                                                       More...
F5=Refresh   F11=Display byte counts   F13=Sort by column
F14=Display port numbers   F22=Display entire field   F24=More keys
```

Figure 754. Using NETSTAT to check the REXEC daemon

Locate the line containing `exec` in the Local Port column. This shows that the REXEC daemon is active and waiting for incoming commands.

17.3 REXEC settings

The following section describes the REXEC settings available and how these settings can be configured. Both the Operations Navigator and green-screen interfaces are shown.

17.3.1 Settings can be changed

The REXEC server allows for the configuration of the following parameters:

- Start when TCP/IP is starting

  Should the REXEC server start automatically when the TCP/IP starts?

- Initial number of servers to start

  If the REXEC server system receives many REXEC requests, this setting allows for additional simultaneous executions of REXEC commands. If the system receives many requests, this number should be adjusted upwards.

- Inactivity time-out

  When will the REXEC server disconnect an idle connection?

- ASCII coded character set identifier

  Will any conversion of characters be needed if the server returns any data to the client?

17.3.2 Changing REXEC settings using Operations Navigator interface

The REXEC settings can be changed using Operations Navigator as shown in Figure 755. Complete the following steps:

1. To start Operations Navigator, click **Start->Programs->IBM AS400 Client Access->AS/400 Operations Navigator**. The AS/400 Operations Navigator window appears (Figure 755).
2. Double-click the AS/400 network (A).
3. Double-click the system icon (B) for the AS/400 system that you are configuring. The system components appear.
4. Double-click **Network** (C). The network components appear.
5. Double-click **Servers** (D). The available protocols appear.
6. Double-click **TCP/IP** (E). The available services appear in the right window.
7. Select **Remote execution** in the right window. Right-click, and select **Properties**.

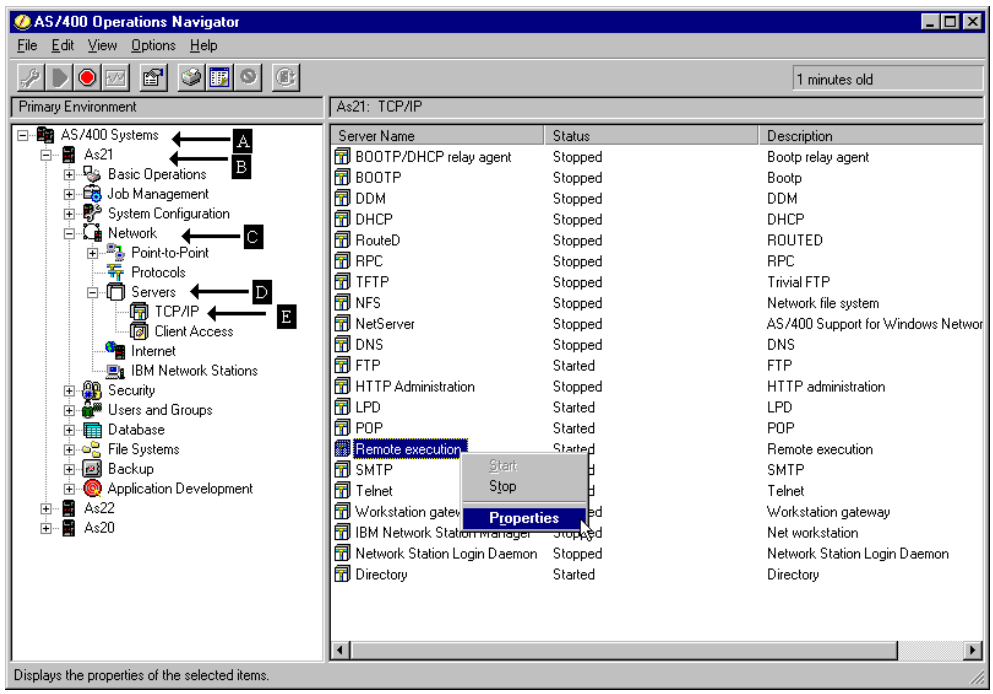


Figure 755. Changing the REXEC settings using the Operations Navigator

This allows you to change the settings as shown in Figure 756.

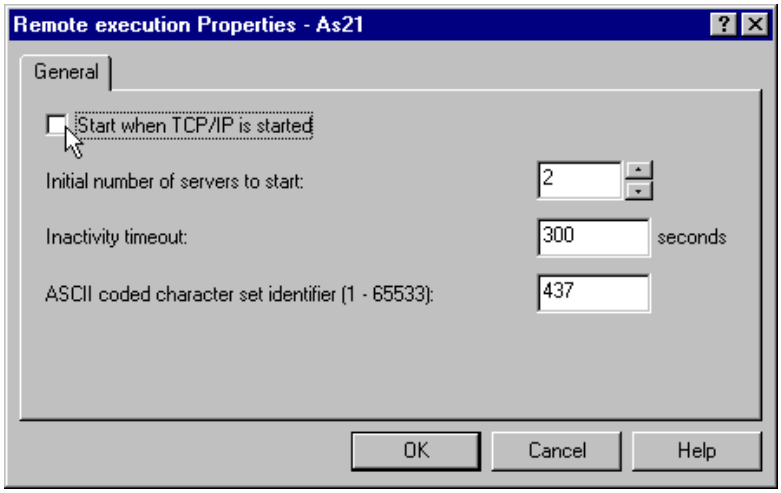


Figure 756. Accessing the REXEC settings using Operations Navigator

8. After changing the settings, the REXEC server should be restarted. This can be done by stopping the server as described in 17.2.2, “Ending the REXEC daemon” on page 617, and starting the server as described in 17.2.1, “Starting the REXEC daemon” on page 615.

17.3.3 Changing REXEC settings using the green-screen interface

To change the REXEC server settings, use the Configure TCP/IP REXEC (`CFGTCPRXC`) command. Enter menu option `1` on the menu that appears. The display shown in Figure 757 appears.

```
                          Configure TCP/IP REXEC
                                                          System:   AS21
Select one of the following:

     1. Change REXEC attributes









Selection or command
===>
```

Figure 757. Changing the REXEC settings using the green-screen interface (Part 1)

```
                    Change REXEC Attributes (CHGRXCA)

Type choices, press Enter.

Autostart servers  . . . . . .   *NO          *YES, *NO, *SAME
Number of initial servers  . . .  2           1-20, *SAME, *DFT
Inactivity timeout . . . . . .   300          1-2147483647, *SAME, *DFT
Coded character set identifier    00437       1-65533, *SAME, *DFT

                                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 758. Changing the REXEC settings using the green-screen interface (Part 2)

After changing the settings, the REXEC server should be restarted. This can be done by stopping the server as described in 17.2.2, “Ending the REXEC daemon” on page 617, and starting the server as described in 17.2.1, “Starting the REXEC daemon” on page 615.

17.4 Scenarios

This section gives a brief introduction and overview to the REXEC scenarios in this chapter. This includes scenarios with the AS/400 system configured as a REXEC server and the AS/400 system configured as a REXEC client.

17.4.1 REXEC server

The AS/400 system as a REXEC server is described in the following two scenarios, connecting different platforms.

17.4.1.1 Scenario 1: AS/400 server and the Windows NT client

This scenario shows the use of the AS/400 system AS21 as a REXEC server. The Windows NT system uses the REXEC client to send a command request to the AS/400 system. The client issues the Display Library (DSPLIB) command to view the contents of the COOL2RED library on the AS21 AS/400 system. The scenario is shown in Figure 759.

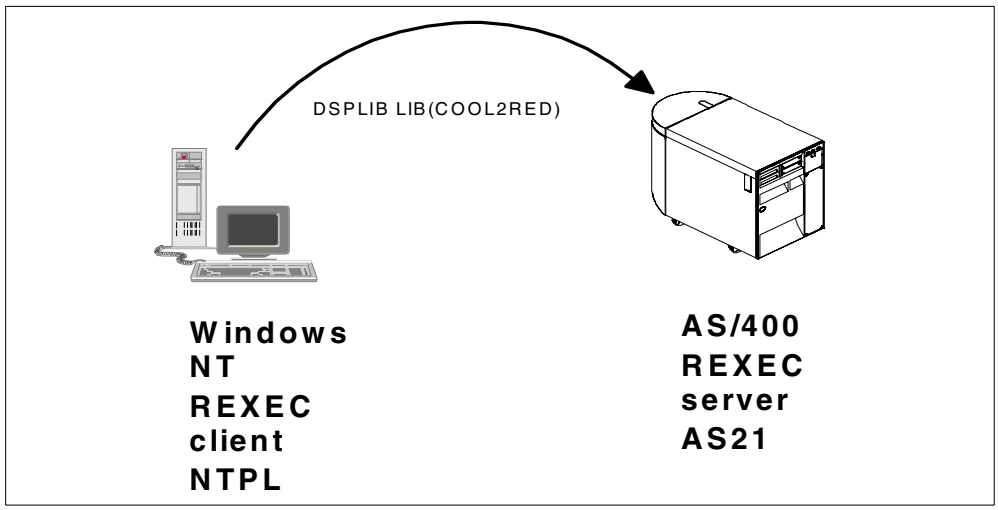


Figure 759. Scenario 1: AS/400 server and Windows NT client

17.4.1.2 Scenario 2: AS/400 server and the AS/400 client

This scenario shows the use of the AS/400 system as both a REXEC client and a REXEC server. The AS/400 system AS21 acts as a REXEC server and the AS/400 system AS22 acts as a REXEC client. The client issues the Display Library (DSPLIB) command to view the contents of the COOL2RED library at the AS21 AS/400 system. The scenario is shown in Figure 760.

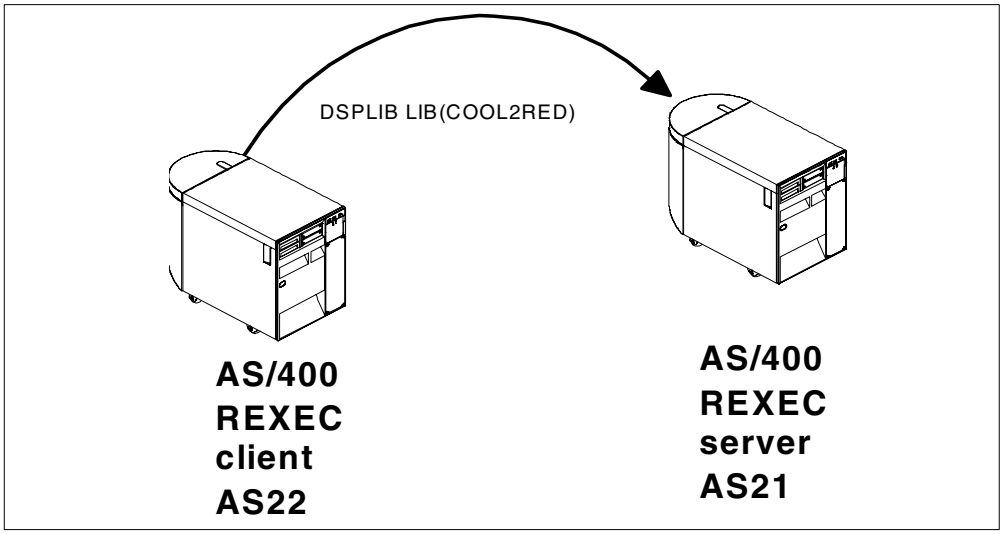


Figure 760. Scenario 2: AS/400 server and AS/400 client

17.4.2 REXEC client

This section shows the use of the AS/400 system as a REXEC client.

17.4.2.1 Scenario 3: Windows 9x server and AS/400 client

This scenario, Figure 761 on page 624, shows the use of the REXEC server function of Client Access for Windows 95/NT, also named *Remote Command.* The PC runs the incoming commands from the AS21 REXEC client. The AS/400

system issues the DIR command to view the root directory on the Windows 9x system.

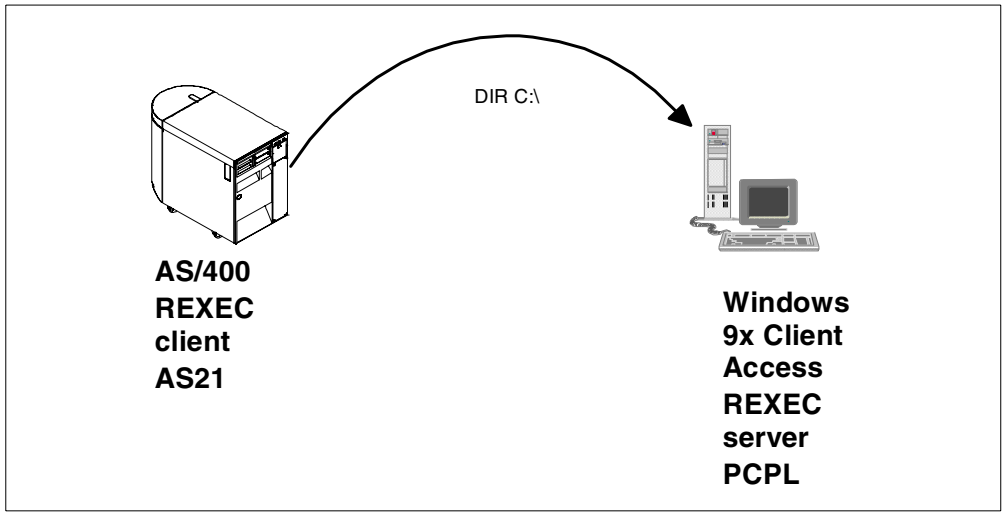


Figure 761. Scenario 3: Windows 9x server using Client Access and AS/400 client

17.5 Configuring the scenarios

This section contains the detailed description and configuration of the REXEC scenarios. The AS/400 system is shown both as a REXEC server and as a REXEC client.

17.5.1 Scenario 1: AS/400 server and the Windows NT client

This section describes scenario 1, where the AS/400 system and the Windows NT systems both use the REXEC protocol to communicate.

17.5.1.1 Task overview

In this scenario, the following actions occur:

1. Ensure that the basic setup of TCP/IP has been completed at both the AS/400 system AS21 and the Windows NT system NTPL. The basic setup includes IP interfaces and host names, either by using the host table or using a DNS server. Refer to Chapter 2, “TCP/IP basic installation and configuration” on page 7, for information on configuring the basic TCP/IP setting on the AS/400 system.
2. Ensure that the REXEC server daemon job is running at the target system AS/400 system AS21. Refer to 17.2.3, “Checking that the REXEC daemon is running” on page 618.
3. Make sure that the target system AS21 has a valid user profile to use when the client Windows NT system NTPL sends a remote command.
4. Execute the command from the requesting Windows NT system NTPL.
5. Check the result of the remote execution.

17.5.1.2 Configuring the scenario

Create a user profile at the AS21 system using the Create User Profile (CRTUSRPRF) command or the Operations Navigator interface. The green-screen interface is shown in Figure 762.

```
                        Create User Profile (CRTUSRPRF)

Type choices, press Enter.

User profile . . . . . > REXEC          Name
User password  . . . . . .  REXECCOOL     Name, *USRPRF, *NONE
Set password to expired  . . . .  *NO           *NO, *YES
Status . . . . . .  *ENABLED      *ENABLED, *DISABLED
User class . . . . . .  *USER         *USER, *SYSOPR, *PGMR...
Assistance level . . . . . .  *SYSVAL       *SYSVAL, *BASIC, *INTERMED...
Current library  . . . . . .  *CRTDFT       Name, *CRTDFT
Initial program to call  . . . .  *NONE         Name, *NONE
  Library  . . . . . .                    Name, *LIBL, *CURLIB
Initial menu . . . . . .  MAIN           Name, *SIGNOFF
  Library  . . . . . .    *LIBL         Name, *LIBL, *CURLIB
Limit capabilities . . . . . .  *NO            *NO, *PARTIAL, *YES
Text 'description' . . . . . .  User for executing remote commands on AS21



                                                                      Bottom
F3=Exit    F4=Prompt    F5=Refresh    F10=Additional parameters    F12=Cancel
F13=How to use this display         F24=More keys
```

Figure 762. Creating a user profile for use when issuing the REXEC command

17.5.1.3 Testing the scenario

Use the following steps to test the scenario:

1. Start a command entry on the Windows NT system as shown in Figure 763.

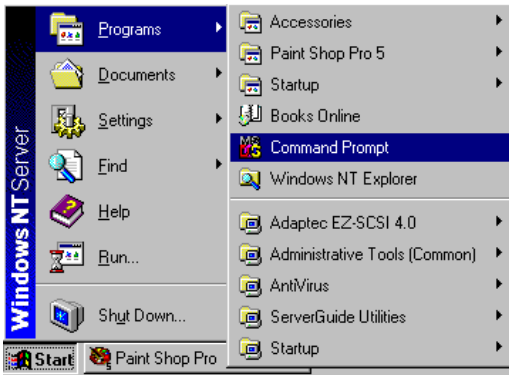


Figure 763. Starting the command entry on the Windows NT system NTPL

2. Type the following command to view the command syntax, as seen in Figure 764 on page 626:

```
REXEC
```

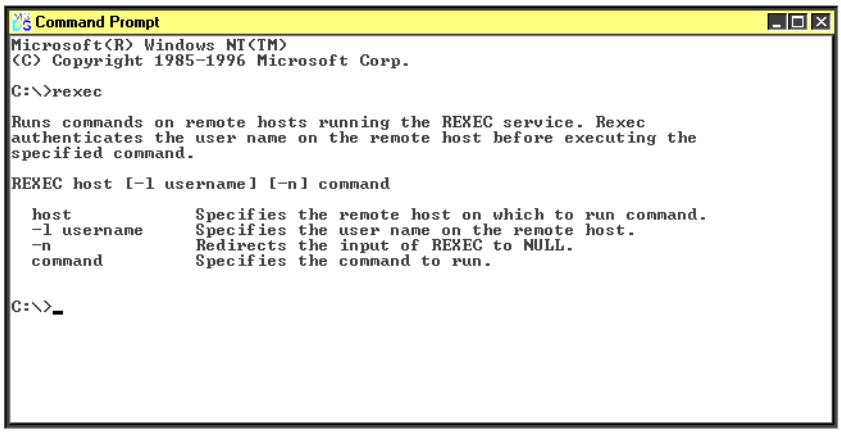


Figure 764. Using the REXEC command to view the command syntax before testing

3. Enter the REXEC command with the correct parameters as shown in Figure 765.

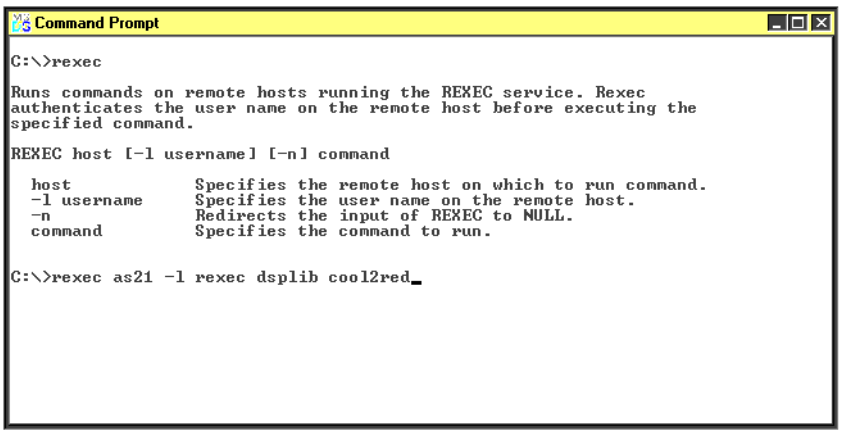


Figure 765. Entering the REXEC command to be run on the AS21 AS/400 system

4. When prompted by the PC system (Figure 766), enter the password.

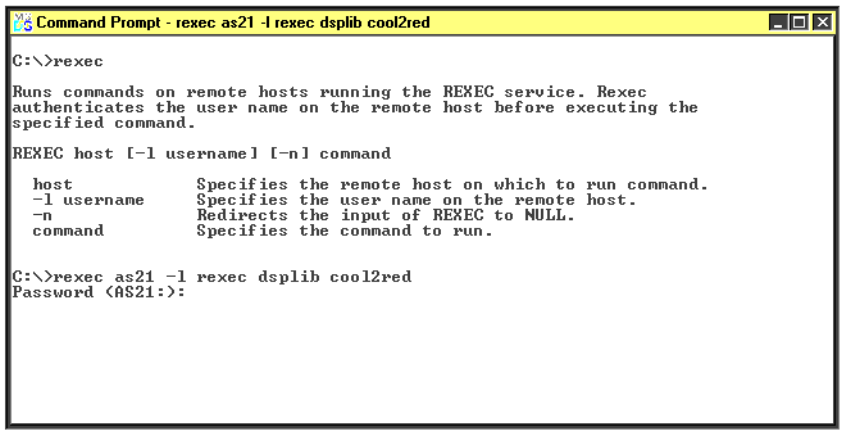


Figure 766. Entering the password for the REXEC user: Password not displayed

5. The result from the execution is returned to the command entry display as shown in Figure 767.

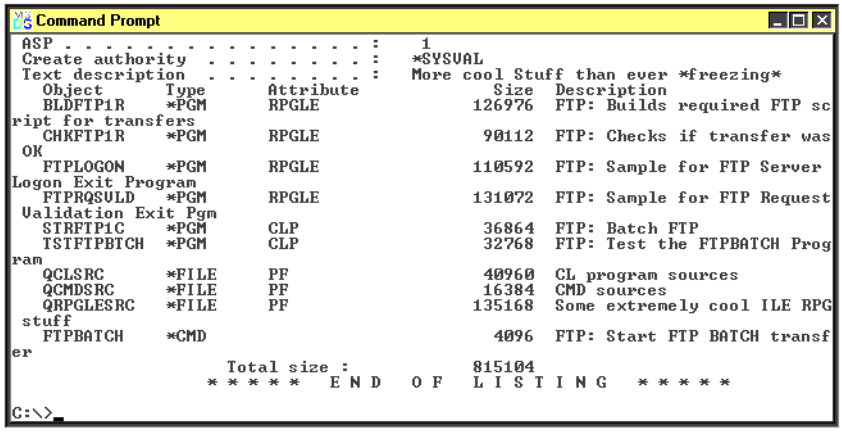


Figure 767. Result of execution displayed on the client after the remote call

17.5.2 Scenario 2: AS/400 server and the AS/400 client

This section describes scenario 2, where two AS/400 systems uses the REXEC protocol to communicate.

17.5.2.1 Task overview

This scenario includes these actions:

1. Ensure that the basic setup of TCP/IP has been completed on both AS21 and AS22 systems. The basic setup includes IP interfaces and host names, either by using the host table on the AS/400 system or using a DNS server. Refer to Chapter 2, “TCP/IP basic installation and configuration” on page 7, for information on configuring the basic TCP/IP setting on the AS/400 system.
2. Ensure that the REXEC server daemon job is running at the target system AS21. Refer to 17.2.3, “Checking that the REXEC daemon is running” on page 618.
3. Make sure that the target system, AS21, has a valid user profile for use when the client system, AS22, sends a remote command.
4. Run the command from the source system, AS22.
5. Check the result of the remote execution.

17.5.2.2 Configuring the scenario

Create a user profile at the AS21 system using the Create User Profile (CRTUSRPRF) command on the Operations Navigator interface. The Operations Navigator interface is shown in Figure 768 and Figure 769 on page 628.

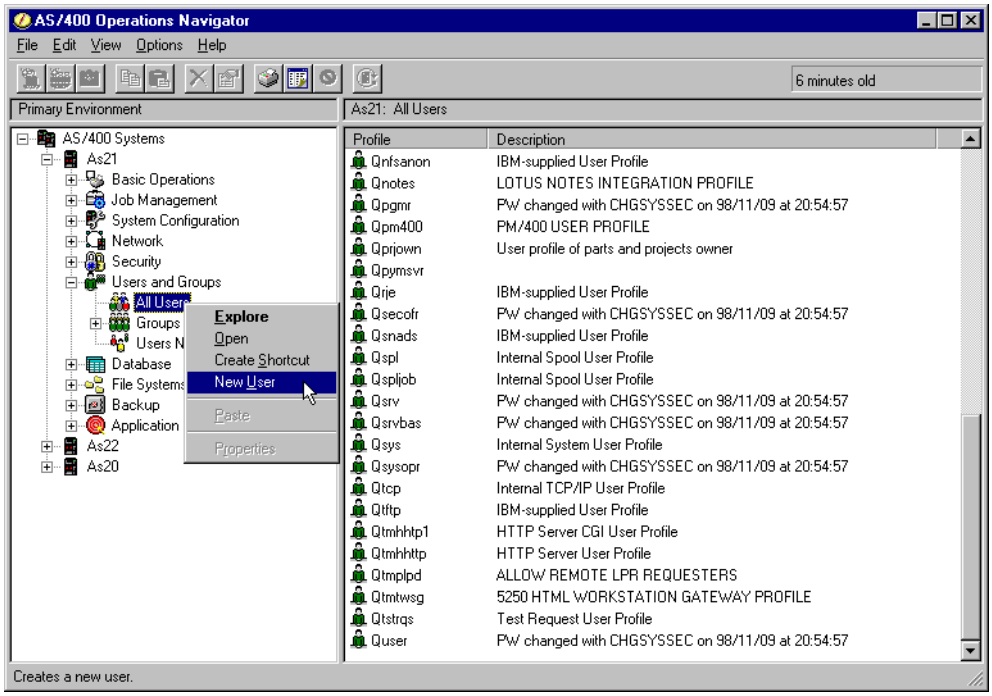


Figure 768. Creating a user profile using Operations Navigator (Part 1)

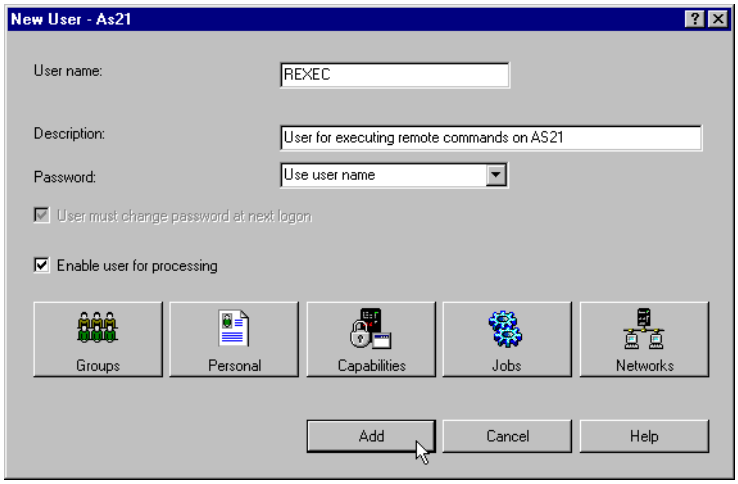


Figure 769. Creating a user profile using Operations Navigator (Part 2)

17.5.2.3 Testing the scenario

To test the scenario, follow these steps:

1. Execute the Run Remote Command (RUNCMTCMD) at the source system AS22 (Figure 770).

```
                    Run Remote Command (RUNRMTCMD)

Type choices, press Enter.

Command  . . . . . . . . . . . . > 'DSPLIB LIB(COOL2RED)'




                                                                      ...
Remote location:
  Name or address  . . . . . . . > AS21


  Type . . . . . . . . . . . . . > *IP          *SNA, *IP
Remote user ID . . . . . . . . . > REXEC        Character value, *NONE...
Remote password  . . . . . . . .   REXEC        Character value, *NONE

                                                                    Bottom
```

Figure 770. Sending the remote command from the AS22 system to the AS21 system

2. The result of the command is returned from the AS21 REXEC daemon to the AS22 REXEC client as a spooled file. Any messages from the job log at the remote system and any spooled files generated at the remote system are returned to the client. Use the Work with Spool Files (WRKSPLF) command to view the output associated with the REXEC user profile as shown in Figure 771 and Figure 772 on page 630.

```
                        Work with All Spooled Files

Type options, press Enter.
  1=Send   2=Change   3=Hold   4=Delete   5=Display   6=Release   7=Messages
  8=Attributes        9=Work with printing status


                           Device or                      Total     Cur
Opt  File        User      Queue       User Data   Sts    Pages    Page  Copy
     QSYSPRT     ITSCID29  QPRINT                  RDY        1             1
     QSYSPRT     ITSCID29  QPRINT                  RDY        1             1
     QSYSPRT     ITSCID29  QPRINT                  RDY        4             1
     QSYSPRT     ITSCID29  QPRINT                  RDY        4             1
     QPCSMPRT    ITSCID29  QPRINT                  RDY       45             1
     QSYSPRT     ITSCID29  QPRINT                  RDY        1             1
 5   QSYSPRT     ITSCID29  QPRINT                  RDY        1             1


                                                                      Bottom
Parameters for options 1, 2, 3 or command
===>
```

Figure 771. Verifying the result using the WRKSPLF command

```
                              Display Spooled File
File  . . . . . :   QSYSPRT                          Page/Line   1/6
Control . . . . .                                    Columns     1 - 78
Find  . . . . . .
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
Printer device PRT01 not found. Output queue changed to QPRINT in library QGPL
 5769SS1 V4R3M0  980729                 Display Library
               11/11/98 11:29:50        Page    1
 Library . . . . . . . . . . . . . :   COOL2RED
 Type  . . . . . . . . . . . . . . :   PROD
 Number of objects . . . . . . . . :          10
 ASP . . . . . . . . . . . . . . . :    1
 Create authority  . . . . . . . . :   *SYSVAL
 Text description  . . . . . . . . :   More cool Stuff than ever *freezing*
   Object      Type      Attribute            Size  Description
   BLDFTP1R    *PGM      RPGLE              126976  FTP: Builds required FTP
ript for transfers
   CHKFTP1R    *PGM      RPGLE               90112  FTP: Checks if transfer w
 OK
   FTPLOGON    *PGM      RPGLE              110592  FTP: Sample for FTP Serve
Logon Exit Program
```

Figure 772. Viewing the result of the remote command

17.5.3 Scenario 3: Windows 9x server and the AS/400 client

This section describes scenario 3, where the AS/400 system and the Windows 9x systems both uses the REXEC protocol to communicate. The REXEC protocol on the Windows 9x system is installed as part of Client Access for Windows 95/NT.

17.5.3.1 Task overview

In this task, the following process occurs:

1. Ensure that the basic setup of TCP/IP has been completed at both the AS/400 system (AS21) and the Windows 95 PC (PCPL). The basic setup includes IP interfaces and host names, either by using the host table or using a DNS server. Refer to Chapter 2, “TCP/IP basic installation and configuration” on page 7, for information on configuring the basic TCP/IP setting on the AS/400 system.
2. Ensure that the REXEC server daemon job is running at the PC. The REXEC server is part of Client Access for Windows 95/NT.
3. Ensure that the REXEC daemon at PCPL has been configured with a valid user ID and a password.
4. Execute the command from the source system AS21.
5. Check the result of the remote execution.

17.5.3.2 Configuring the scenario

Configure the REXEC daemon at PCPL by using the following procedure:

1. Select **Start->Programs->IBM AS/400 Client Access->Client Access Properties** (Figure 773).

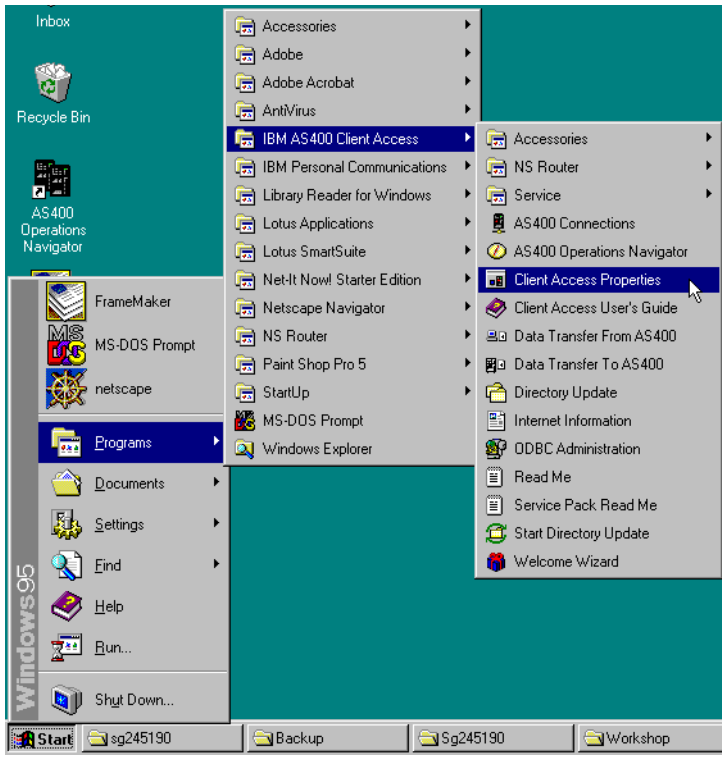


Figure 773. Selecting the Client Access properties

2. Select the **Remote Command** tab (Figure 774).

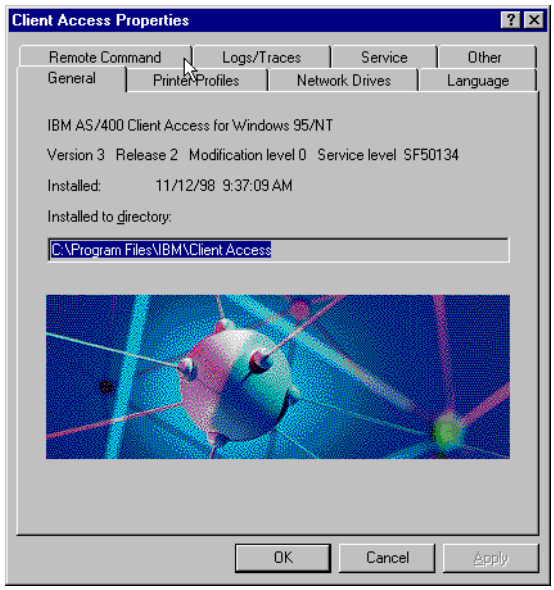


Figure 774. Selecting Remote Command

3. Click **Add** to add a new user (Figure 775 on page 632).

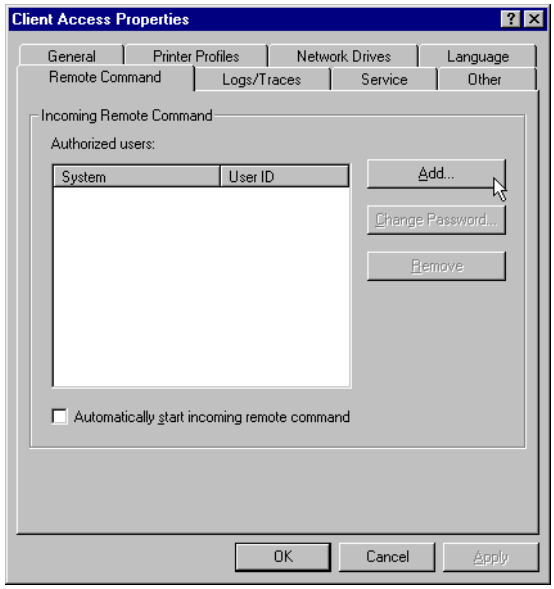


Figure 775. Adding a user ID and password using the Add button

4. Enter the required information (Figure 776), and click the **OK** button.

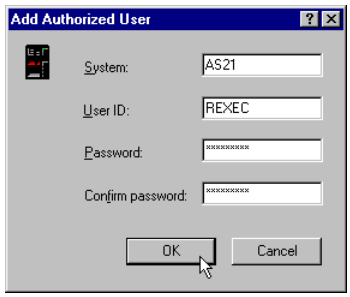


Figure 776. Entering the system name, user ID, and password

5. Place a check mark in the option to automatically start the REXEC daemon when the PC boots (Figure 777).

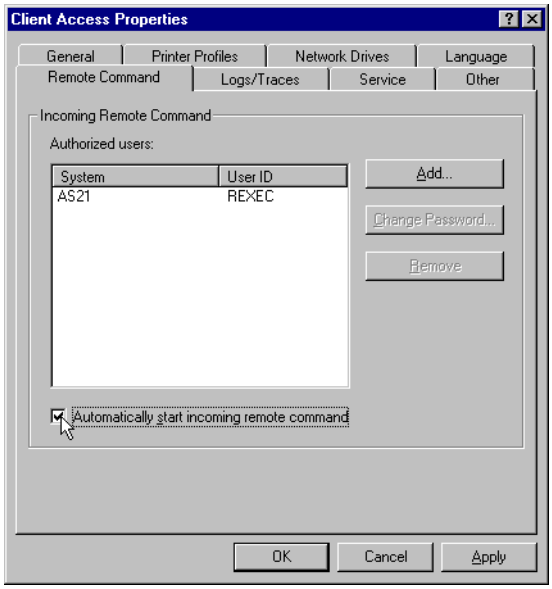


Figure 777. Enabling the REXEL daemon to start when the PC boots

6. Click the **OK** button to confirm the setting.

   The REXECD is now ready.

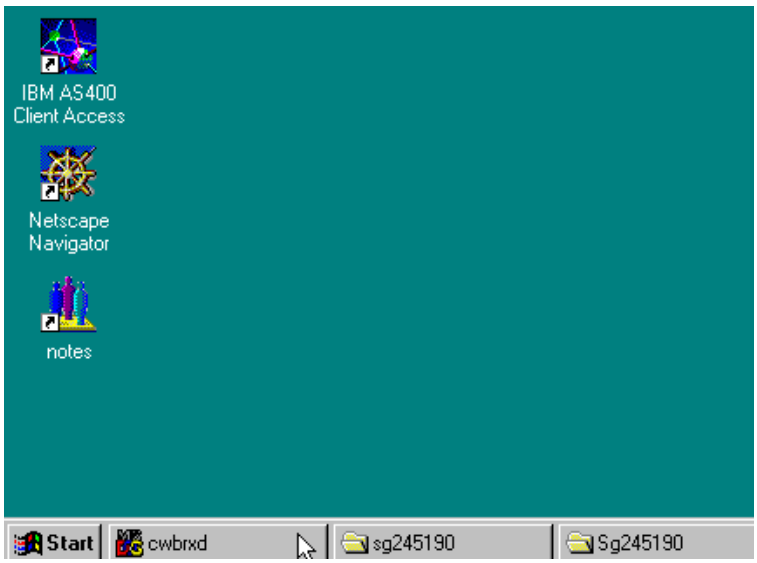


Figure 778. The cwbrxd window minimized and ready after the PC is rebooted

7. Maximize the window running cwbrxd (Figure 778) to check the status of the daemon. Figure 779 on page 634 shows that the REXEC daemon is ready to accept incoming commands.

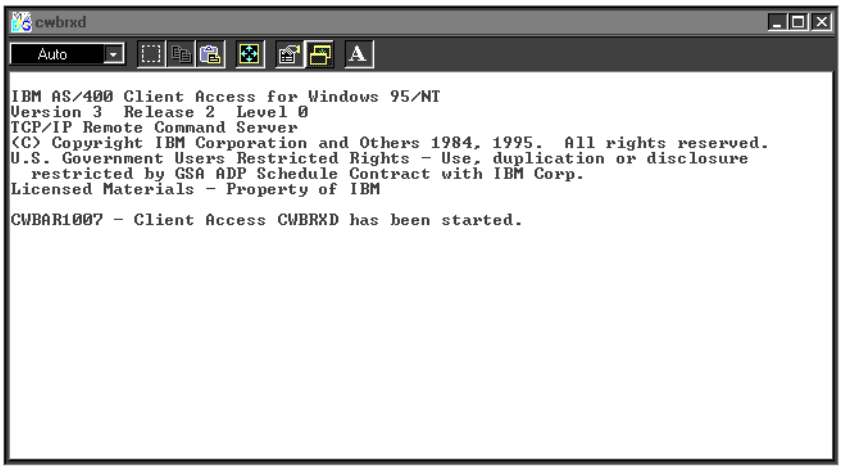


Figure 779. Maximizing the CWBRXD icon

17.5.3.3 Testing the scenario

Now test the scenario by using these steps:

1. Enter the Run Remote Command (RUNRMTCMD) on the AS21 AS/400 system as shown in Figure 780.

```
                    Run Remote Command (RUNRMTCMD)

Type choices, press Enter.

Command  . . . . . . . . . . . . > 'DIR C:\'




                                                                      ...
Remote location:
  Name or address  . . . . . . . > PCPL


  Type . . . . . . . . . . . . . > *IP          *SNA, *IP
Remote user ID . . . . . . . . . > REXEC        Character value, *NONE...
Remote password  . . . . . . . .   REXECCOOL    Character value, *NONE

                                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display       F24=More keys
```

Figure 780. Entering the RUMRMTCMD on the AS21 AS/400 system

2. The execution of the remote command can be verified by maximizing the *cwbrxd* icon (Figure 781) and by checking the spooled files (Figure 782) of the user signed on to the AS21 AS/400 system.

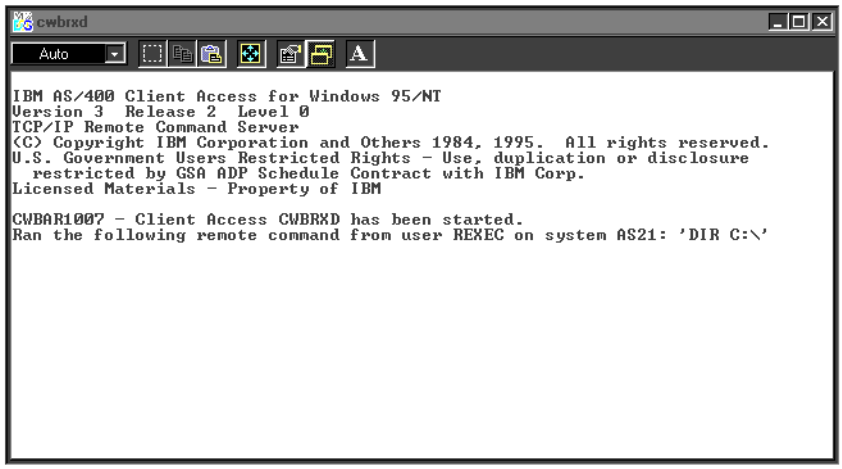


Figure 781. Verifying the execution of the remote command at the PCPL PC

```
                              Display Spooled File
File  . . . . . :   QSYSPRT                          Page/Line   1/6
Control . . . . .                                    Columns     1 - 78
Find  . . . . . .
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
 Volume in drive C is ITSO
 Volume Serial Number is 3642-1E86
 Directory of C:\
COMMAND  COM        92,870  07-11-95  9:50a COMMAND.COM
CONFIG   SYS             0  11-05-98  5:07p CONFIG.SYS
WIN95          <DIR>        04-01-98  8:31a WIN95
AUTOEXEC BAT            93  11-05-98  5:07p AUTOEXEC.BAT
NETLOG   TXT           489  04-01-98  8:24a NETLOG.TXT
WINDOWS        <DIR>        04-01-98  8:21a WINDOWS
PROGRA~1       <DIR>        04-01-98  8:25a Program Files
AUTOEXEC 001            47  04-01-98  9:47a AUTOEXEC.001
READIBMW       <DIR>        04-01-98  9:46a READIBMW
IBMAV95        <DIR>        05-21-98 10:09a IBMAV95
TEMP           <DIR>        04-01-98  9:19a TEMP
BACKUP~1 BAT            64  05-21-98 10:06a BACKUP OF AUTOEXEC.BAT
NOTES          <DIR>        04-01-98  9:32a notes
```

Figure 782. Verifying the remote command by checking the spooled file on AS21

17.6 Security considerations

There are a few points that should be considered before using the REXEC functions. This relates to all platforms using the REXEC protocol, not just the AS/400 system. As shown in Figure 749 on page 615, the REXEC function requires the use of a user ID and a password. During the protocol conversations between the REXEC client and the REXEC server daemon, the user ID and password are exchanged between the two hosts. These values are not encrypted in any way. That is, the user ID and the password are sent across the network as clear-text. Look at the portion of the communication trace shown in Figure 783 on page 636. Look for the `REXECCOOL` text to identify the user ID (REXEC) and the password (REXECCOOL).

```
     80   R      84     3836.8                0004AC47A3C7  8004AC210475   LLC     UI                        OFF
AA    AA
        Routing Info . :  0270
                    Frame Type :  IP           TOS: NORMAL           Length:    79  Protocol: TCP          Datagram
ID: D883
                                     Src Addr: 10.1.1.2         Dest Addr: 10.1.1.1         Fragment Flags: MAY  ,LAST
                      SNAP Header:  0000000800
                      IP Header  :  4500004FD883000040068C210A0101020A010101
                      IP Options :  NONE
                      TCP  . . . :  Src Port:  1028,Unassigned    Dest Port:   512,EXEC
                                    SEQ Number:  1718700105 ('66714849'X)  ACK Number: 1710958100 ('65FB2614'X)
                                    Code Bits: ACK PSH                  Window:  8192  TCP Option: NONE
                      TCP Header :  04040200667148496 5FB261450182000F0860000
        Data . . . . . :  3000524558454300 5245584543434F4F  4C004453504C4942 204C494228434F4F
*0.REXEC.REXECCOOL.DSPLIB LIB(COO*
                     4C325245442900                                              *L2RED).
*
```

Figure 783. Communications trace of REXEC session

This is a serious security consideration to note, since the user IDs and the passwords can be grabbed using sniffers and other TCP/IP trace or dump utilities. Therefore, the REXEC function should *not* be used across a public TCP/IP network, such as the Internet. Special considerations should also be paid to the use of the REXEC function within the internal networks of the organization.

If the use of the REXEC function is required across any public TCP/IP networks, consider using VPN. For more information on VPN, please refer to *TCP/IP Tutorial and Technical Overview,* GG24-3376, and *AS/400 Internet Security: Implementing AS/400 Virtual Private Networks* , SG24-5404.

The REXEC function can also be secured by the use of IP Filters, allowing only selected hosts to communicate with a particular REXEC server. Refer to Chapter 10, “Network Address Translation and IP Packet Filtering” on page 359, for more information about IP filtering.

The ordinary OS/400 security system is used when using the REXEC function, so any command or function the users has access to on the AS/400 system using an ordinary 5250 session is also available when using the REXEC function. For example, if the user has authority to the Power Down System (PWRDWNSYS) command, effectively the user can power down the system from any remote host using the REXEC function.

The system value, QMAXSIGN (Maximum sign-on attempts allowed), works the same way as when a user is using a 5250 session. After the maximum allowed tries has been used, the user profile is disabled as shown in Figure 784.

```
                        Additional Message Information

Message ID . . . . . :   CPF1393       Severity . . . . . :   70
Message type . . . . . :   Information
Date sent  . . . . . :   11/13/98      Time sent  . . . . . :   08:51:03

Message . . . . :   Subsystem QSYSWRK disabled user profile ITSCID29 on device
  *N.
Cause . . . . . :   User profile ITSCID29 has been disabled because the
  maximum number of sign-on attempts specified for the QMAXSIGN system value
  has been reached.
Recovery  . . . :   To enable the user profile, have the security officer
  change the STATUS parameter to *ENABLED on the Change User Profile
  (CHGUSRPRF) command.
                                                                        Bottom
Press Enter to continue.

F3=Exit   F6=Print   F9=Display message details   F12=Cancel
F21=Select assistance level
```

Figure 784. The user is disabled if the maximum signon attempt is reached

The system values Limit device sessions and Limit security officer device access (QLMTSECOFR) do not limit the use of the REXEC command, since a display device is not used. The REXEC jobs are treated as batch jobs.

17.7 Common errors

The typical error in relation to REXEC is missing or misspelled user profiles or passwords. Furthermore, misspelled remote commands or supplying the wrong IP address or host names can lead to problems.

If using a REXEC client on, for example, a Windows 9x or Windows NT system, the command entry shows if any errors are encountered. Any messages from the job being executed at the AS/400 REXEC server are returned to the client as shown in Figure 767 on page 627.

If using the REXEC client on the AS/400 system, any messages related to the execution at the remote host is are found is the spooled file returned from the remote system. This is shown in Figure 782 on page 635.

Chapter 18. DDM and DRDA over TCP/IP

This chapter contains information about Distributed Data Management (DDM) and Distributed Relational Database Architecture (DRDA) over TCP/IP. Support for DRDA and DDM over TCP/IP was made available with OS/400 version V4R2M0.

18.1 An overview of DDM and DRDA

DDM and DRDA are protocols used to access data distributed across multiple machines. DRDA is an open architecture and the architecture reference is published by the open group. DDM is part of the DRDA architecture.

DDM is part of the Operating System/400 licensed program. DDM support on the AS/400 system allows application programs or users to access data files that reside on remote systems. It also allows remote systems to access data files on the local AS/400 system. Any system that supports the DDM architecture as a source system can access data (if authorized to do so) on any other system to which it is attached. The attached system must support DDM as a target system (the system that receives a request from another system to use one or more files located on the system). However, the source and target systems must support compatible subsets and levels of the DDM architecture. For more information on DDM levels and compatibility, see *Distributed Data Management V4R1*, SC41-5307.

Folder management services (FMS) support allows personal computer users to access folders and documents that reside on an AS/400 target system. Remote systems that support Level 3.0 or Level 2.0 of the DDM architecture for the stream access method can access folders and documents on the local AS/400 system.

DDM extends the file accessing capabilities of the AS/400 database management support. In this redbook, database management refers to the system function that controls local file processing. It controls access to data in files stored on the local AS/400 system, and it controls the transfer of that data to requesting programs on the same system.

DDM controls remote file processing. It enables application programs running on one AS/400 system to access data files stored on another system supporting DDM. Similarly, other systems that have DDM can access files in the database of the local AS/400 system. DDM makes it easier to distribute file processing between two or more systems. DRDA support for distributed relational database processing is used by IBM relational database products. DRDA support defines protocols for communication between an application program and a remote relational database.

DRDA support provides distributed relational database management in both IBM and non-IBM environments. In IBM environments, relational data is managed with the following programs:

- DB2 for OS/390
- DB2 for VSE and VM
- DB2 Connect Personal Edition
- DB2 Connect Enterprise Edition

- DB2 Universal Database Workgroup Edition
- DB2 Universal Database Enterprise Edition
- DB2 Universal Database Extended Enterprise Edition
- DB2 for AS/400 support in the OS/400 licensed program on the AS/400 system

DRDA support provides the structure for access to database information for relational database managers operating in like and unlike environments. For example, access to relational data between two or more AS/400 systems is distribution in a like environment, and access to relational data between an AS/400 system and systems using the DB2 database manager is distribution in an unlike environment.

SQL is the standard IBM database language. It provides the necessary consistency to enable distributed data processing across like and unlike operating environments. Within DRDA support, SQL allows users to define, retrieve, and manipulate data across environments that support a DRDA implementation.

The DRDA implementation on the AS/400 system uses DDM architecture commands to communicate with other systems. However, distributed relational database and DDM support handle some functions differently.

Using distributed relational database processing, the application connects to a remote system using a relational database directory on the local system. The relational database directory provides the necessary links between a relational database name and the communications path to that database. An application running under a distributed relational database only has to identify the database name and run the SQL statements needed for processing.

Using DDM support, the remote file is identified and the communications path is provided by means of a DDM file on the local system.

18.2 Using DDM over TCP/IP

DDM can be used for:

- Allocating, opening, or closing one or more files.
- Reading, writing, changing, or deleting records in a file.
- Copying the contents of a file.
- Performing operations on physical or logical file members (such as adding, clearing, or removing members), but only if the target is an AS/400 system or System/38.
- Accessing remote files for non-data purposes, such as:
  - Displaying information about one or more files, using such commands as Display File Description (DSPFD) and Display File Field Description (DSPFFD). These commands can display the file attributes of the DDM file on the source system or the file or field attributes of the remote file on the target system.
  - Controlling the locking of files on the target system, using the Allocate Object (ALCOBJ) and Deallocate Object (DLCOBJ) commands.

– Deleting, renaming, creating, and changing files using the Delete File (DLTF), Rename Object (RNMOBJ), Create Physical File (CRTPF), Create Source Physical File (CRTSRCPF), Create Logical File (CRTLF), Change Physical File (CHGPF), Change Logical File (CHGLF), and Change Source Physical File (CHGSRCPF) commands.

– Sending a CL command to the target system (an AS/400 system and a System/38 only) so it can be run there, instead of on the source system (where it may not be useful to run it), using the Submit Remote Command (SBMRMTCMD) command. The SBMRMTCMD command is the method you use to move, save, or restore files on a target system. For example, a Move Object (MOVOBJ) command may be sent to move a database file on the target system. For typical uses of the SBMRMTCMD command, refer to its description in Chapter 5, “CL Command Descriptions and DDS Considerations for DDM”, or refer to the four volumes of the *OS/400 CL Reference* set, SC41-5723, SC41-5724, SC41-5725 and SC41-5726, for a more complete description. Plus, for an abridged version, see *OS/400 CL Reference*, SC41-5722.

18.2.1 DDM server

The AS/400 system must be running the DDM server to act as either a DDM or DRDA application server. This section describes how to configure and start the DDM server.

18.2.1.1 Configuring the DDM server

The DDM server must be configured and started on the AS/400 host system. Once the DDM server is started, AS/400 and non-AS/400 DDM compliant systems can access files on the AS/400 system through the DDM interface.

Adding the host table entry

Issue the `GO CFGTCP` command, and enter option `10` (Work with TCP/IP host table entries). The screen shown in Figure 785 on page 642 is displayed.

```
                  Work with TCP/IP Host Table Entries
                                                              System:   AS22
Type options, press Enter.
  1=Add   2=Change   4=Remove   5=Display   7=Rename

     Internet         Host
Opt  Address          Name
 1   10.5.69.234
     10.5.69.230      AS20
     10.5.69.232      AS21
     10.5.69.233      AS22
     10.1.2.2         NTPL
     127.0.0.1        LOOPBACK
                      LOCALHOST

                                                                      Bottom
F3=Exit   F5=Refresh   F6=Print list   F12=Cancel   F17=Position to
```

Figure 785. Using the Work with TCP/IP Host Table Entries to add a new entry

Enter option `1` (Add), and type the address you are adding. Press Enter. The display shown in Figure 786 appears.

```
                  Add TCP/IP Host Table Entry (ADDTCPHTE)

Type choices, press Enter.

Internet address . . . . . > '10.5.69.234'
Host names:
  Name . . . . . .  _____________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________
             + for more values _
Text 'description' . . . . . .  _____________________________________________
_____________________________

                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 786. Adding a New TCP/IP Host Table Entry

Type the host name and description, and press Enter. You return to the Work with TCP/IP Host Table Entries screen (Figure 785). Your new host should be added to the list.

Changing the DDM TCP/IP attributes

Perform the following steps:

1. Start Operations Navigator by clicking **Start->Programs->IBM AS400 Client Access->AS/400 Operations Navigator**. The AS/400 Operations Navigator window appears (Figure 787).

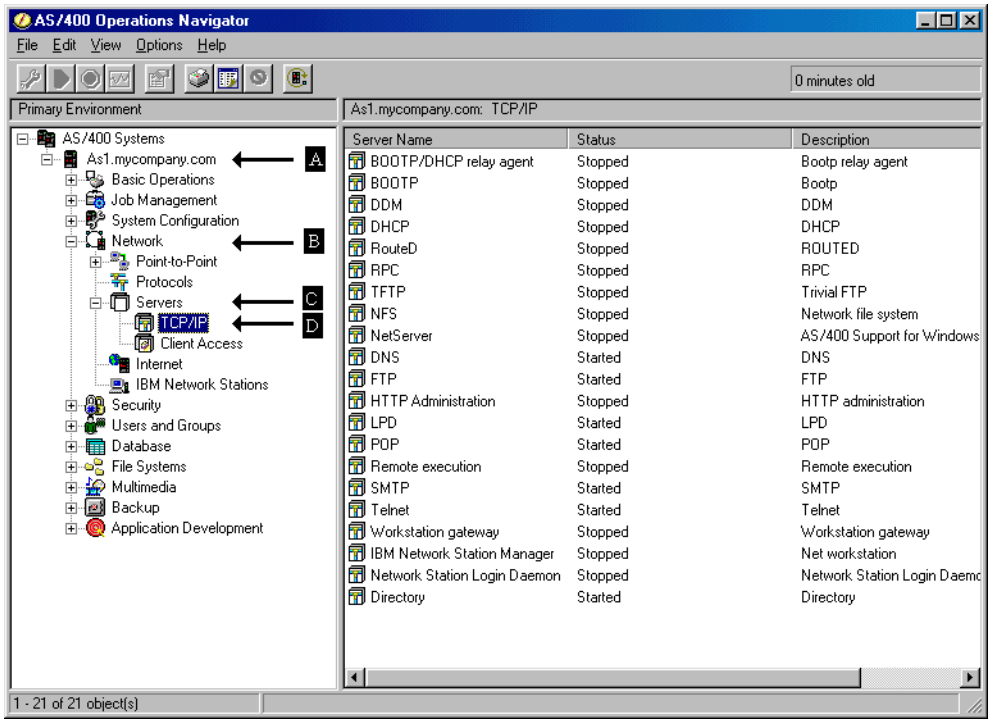


Figure 787. Operations Navigator: TCP/IP servers

2. Double-click the system icon (A) for the AS/400 system that you are configuring. The system components appear.
3. Double-click the **Network** icon (B). The network components appear.
4. Double-click the **Servers** icon (C). The available server types appear.
5. Double-click the **TCP/IP** icon (D). All of the TCP/IP servers are listed in the right window.
6. Right-click the DDM server in the right-hand window. The context menu for the DDM server appears (Figure 788 on page 644).

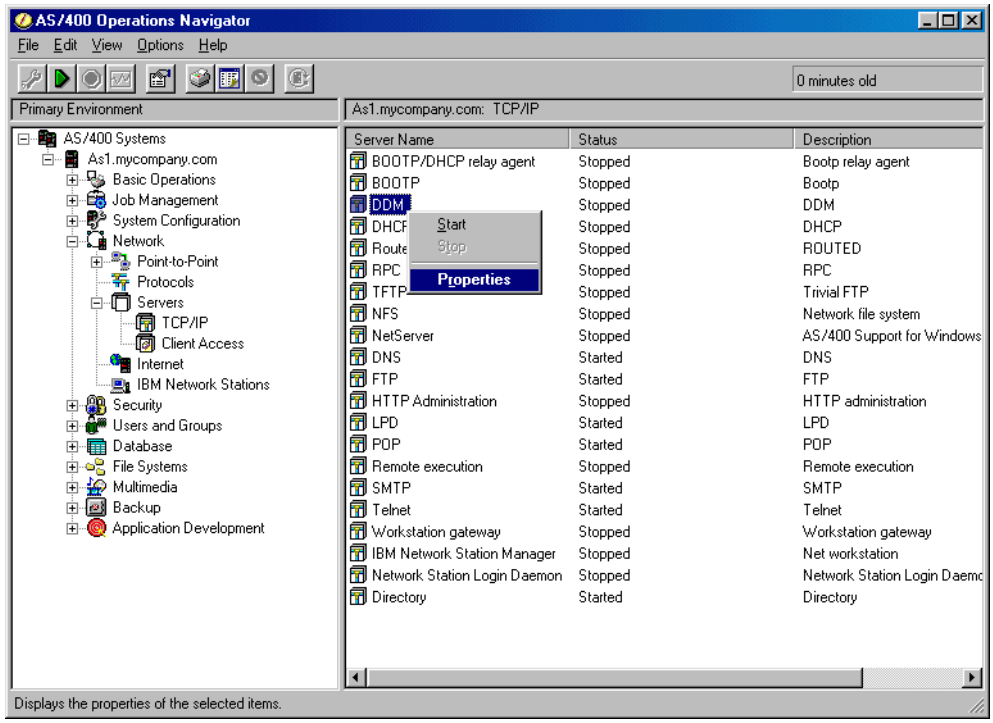


Figure 788. Selecting properties on the DDM server context menu

7. Select **Properties** from the context menu, and the DDM server properties window appears (Figure 789).

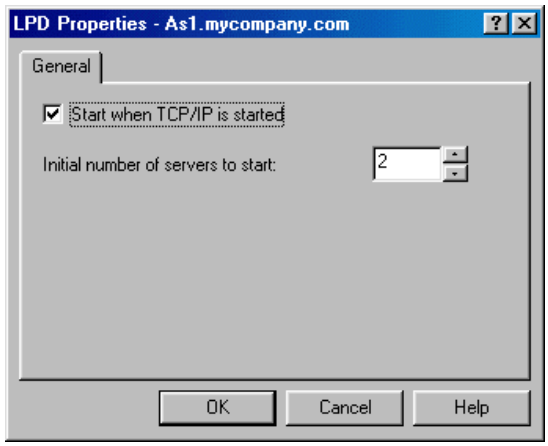


Figure 789. DDM server Properties window

You can now select the number of listener jobs to be started and whether the DDM server should be started whenever TCP/IP is started on the AS/400 system.

Starting the DDM server

If you set the Autostart server parameter to *YES when setting the DDM TCP/IP attributes (see step 2 on page 649), the server will be started when TCP/IP is started.

You can start the DDM server manually through Operations Navigator by completing these steps:

1. Start Operations Navigator, and open the TCP/IP servers display (see steps 1 through 5 in “Changing the DDM TCP/IP attributes” on page 643).
2. Right-click on the DDM server in the right-hand window. The context menu for the DDM server appears (Figure 790).

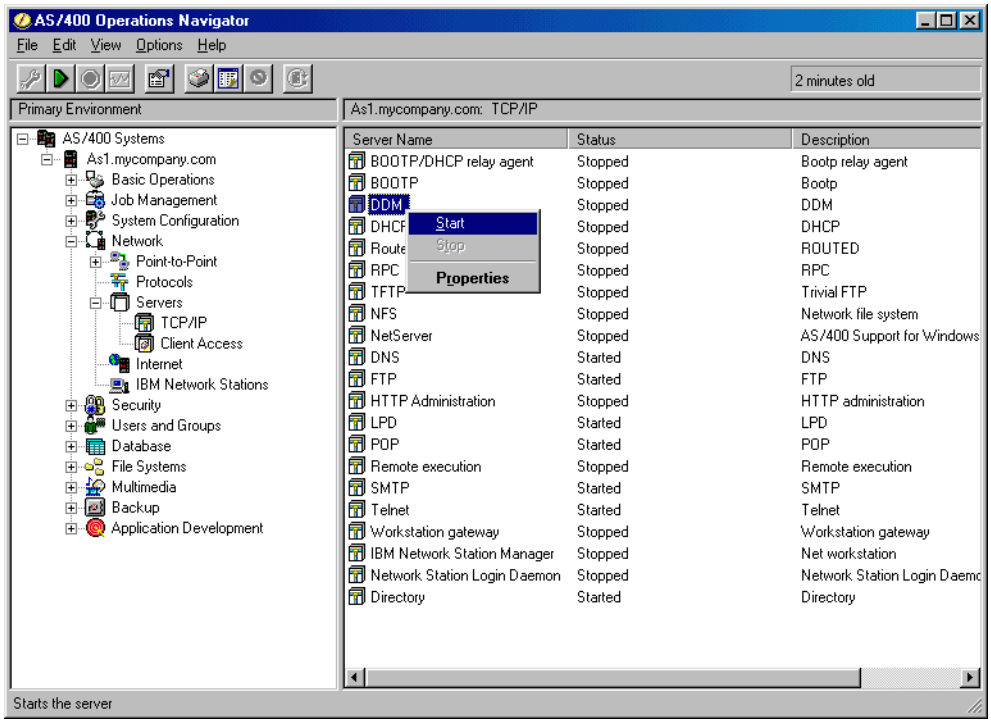


Figure 790. Selecting Start on the DDM server context menu

3. Select **Start** from the context menu.

You can also start the DRDA/DDM server manually by using the command:

```
STRTCPSVR SERVER(*DDM)
```

When the server has been started, you can find a new job, QRWTLSTN, running in the QSYSWRK subsystem. This is the listener job waiting to service connection requests on port 446.

Stopping the DDM server

You can stop the DDM server manually through Operations Navigator by completing the following steps:

1. Start Operations Navigator and open the TCP/IP servers display (see steps 1 through 5 in “Changing the DDM TCP/IP attributes” on page 643).
2. Right-click on the DDM server in the right-hand window. The context menu for the DDM server appears (Figure 791 on page 646).

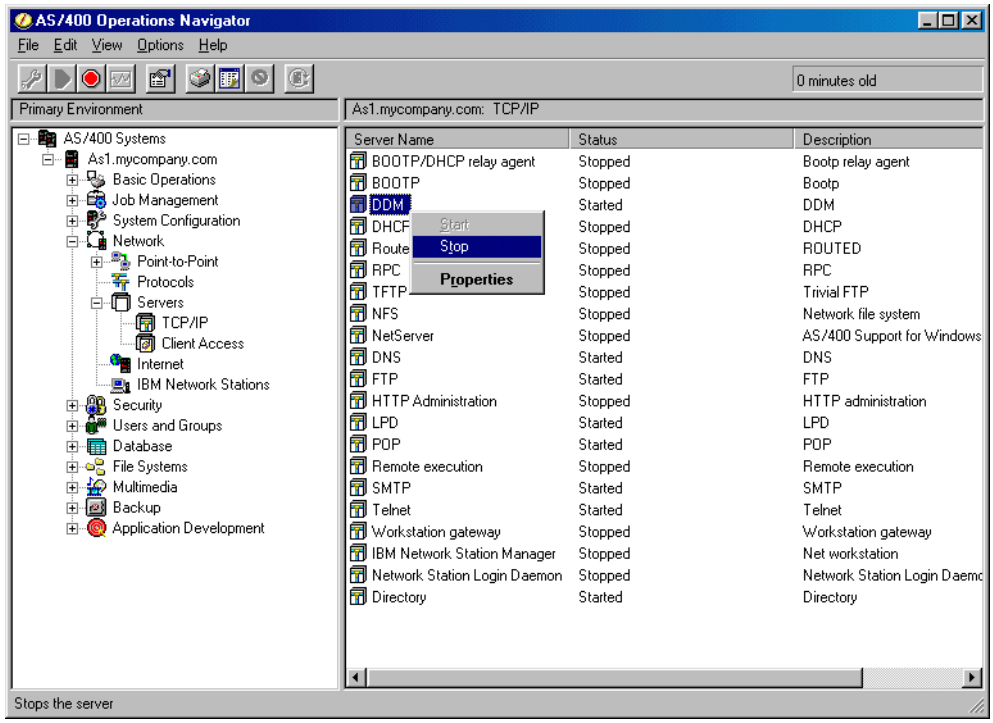


Figure 791. Selecting Stop on the DDM server context menu

3. Select **Stop** from the context menu.

You can also end the DDM server by issuing the command:

```
ENDTCPSVR SERVER(*DDM)
```

18.2.2 DDM client

OS/400 V4R3 does not support DDM source (client) access over TCP/IP. That is, you cannot create a DDM file on an AS/400 system that uses TCP/IP. This also means that you cannot run a SBMRMTCMD on an AS/400 system over a TCP/IP connection. However, the new TCP/IP support allows PC clients using DDM to access DB2 for AS/400 as a DDM server over TCP/IP, and the RUNRMTCMD can possibly be used as a substitute for SBMRMTCMD over TCP/IP.

AS/400 systems running OS/400 V4R3 or earlier must use DRDA to act as an application client over TCP/IP. See 18.3.2, “DRDA requester” on page 649.

18.2.2.1 Using DDM over TCP/IP with OS/400 V4R4

OS/400 V4R4 contains DDM client support. This means that a DDM file can be created with the Create DDM File (CRTDDMF) command that uses TCP/IP to talk to the DDM server. This DDM file can then be used that DDM files are currently used on SNA networks.

Figure 792 and Figure 793 show the CRTDDMF parameters under OS/400 V4R4. There is a new parameter for the connection type which can be set to *SNA or *IP. If it is set to *IP, the Remote location parameter is set to the host name of the DDM server.

```
                          Create DDM File (CRTDDMF)

Type choices, press Enter.

DDM file . . . . . > DDMTEST1      Name
  Library  . . . . . >   QGPL       Name, *CURLIB
Remote file:
  File . . . . . > QDDSSRC       Name, *NONSTD
    Library  . . . . . >   QGPL       Name, *LIBL, *CURLIB
  Nonstandard file 'name'  . . .



Remote location:
  Name or address  . . . . . > as22



  Type . . . . . > *IP          *SNA, *IP

                                                                      More...
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display       F24=More keys
```

Figure 792. Create DDM File (CRTDDMF) parameters in OS/400 V4R4 (Part 1)

```
                          Create DDM File (CRTDDMF)

Type choices, press Enter.

Text 'description' . . . . . .   'Test DDM over TCP/IP file'









                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display       F24=More keys
```

Figure 793. Create DDM File (CRTDDMF) parameters in OS/400 V4R4 (Part 2)

Once the DDM file is created, it can be used by commands and programs just like SNA-based DDM files. For example, the DDM file can be copied using the CPYF command as if it were a local file. Figure 794 on page 648 shows the CPYF parameters to copy the DDM file.

```
                              Copy File (CPYF)

Type choices, press Enter.

From file  . . . . . > DDMTEST1      Name
  Library  . . . . . >   QGPL        Name, *LIBL, *CURLIB
To file  . . . . . > DDMTEST2        Name, *PRINT
  Library  . . . . . >   QGPL        Name, *LIBL, *CURLIB
From member  . . . . . > *ALL        Name, generic*, *FIRST, *ALL
To member or label . . . . .  *FIRST Name, *FIRST, *FROMMBR
Replace or add records . . . . . > *REPLACE  *NONE, *ADD, *REPLACE...
Create file  . . . . . > *YES        *NO, *YES
Print format . . . . . .  *CHAR      *CHAR, *HEX

                                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display       F24=More keys
```

Figure 794. Copy File (COPYF) parameters to copy the DDM file

Once the file is copied, you can use the WRKOBJ command (Figure 795) to see the original DDM file created and the physical file created when it was copied.

```
                              Work with Objects

Type options, press Enter.
  2=Edit authority       3=Copy   4=Delete   5=Display authority   7=Rename
  8=Display description   13=Change description

Opt  Object      Type      Library     Attribute   Text
     QDKT        *OUTQ     QGPL                    Default Diskette Output Que
     QPFROUTQ    *OUTQ     QGPL
     QPRINT      *OUTQ     QGPL                    Default Printer Output Queu
     QPRINTS     *OUTQ     QGPL                    Printer Output Queue Intend
     QPRINT2     *OUTQ     QGPL                    Printer Output Queue Intend
     DDMTEST1    *FILE     QGPL        DDMF        Test DDM over TCP/IP file
     DDMTEST2    *FILE     QGPL        PF          Default source data base fi
     QAAPFILE    *FILE     QGPL        LF          Symbol set symbol definitio
     QAAPFILE$   *FILE     QGPL        PF          Symbol set small symbol def
     QAAPFILE#   *FILE     QGPL        PF          Symbol set medium symbol de
     QAAPFILE@   *FILE     QGPL        PF          Symbol set large symbol def
                                                                      More...
Parameters for options 5, 7 and 13 or command
===>
F3=Exit   F4=Prompt   F5=Refresh   F9=Retrieve   F11=Display names and types
F12=Cancel   F16=Repeat position to   F17=Position to
```

Figure 795. Work with Objects (WRKOBJ) showing the DDM and copied files

18.3 Using DRDA over TCP/IP

The support for DRDA over TCP/IP on the AS/400 system has been made available with OS/400 version V4R2M0. The current implementation of DRDA over TCP/IP supports DRDA level 1. This will satisfy the UNIX/Windows NT client and DataPropagator needs, and provides the foundation on which a future Distributed Unit of Work product will be built.

The DRDA application server is based on multiple connection-oriented server jobs running in the QSYSWRK subsystem. A DRDA background program (listener) listens for TCP connect request on well-known DRDA port 446. The DRDA server jobs are defined by prestart job entries. Once the application requester connects to the listener at the AS, the listener issues a request to wake up a prestarted server job. The listener then passes the socket descriptor to the server job and any further communication occurs directly between the client application and server job.

If you use the SNA implementation of DRDA, you need to configure the controller that governs the communication between the local and the remote system. You then need to refer to this controller in the device description parameter of the ADDRDBDIRE command. If you use the TCP/IP implementation of DRDA, then you can just refer to the IP address of the remote server on the Remote Location Name parameter of the ADDRDBDIRE command, and specify the port (if it is other than the default DRDA port of 446). AS/400 servers will always use the default port. Therefore, all of the complexity of configuring the communications between the two servers is eliminated.

18.3.1 DRDA server

The same server is used for both DDM and DRDA TCP/IP access to DB2 for AS/400. The DRDA server consists of two or more jobs, one of which is called the DRDA listener, because it listens for connection requests and dispatches work to the other jobs. The other jobs, as initially configured, are prestart jobs which service requests from the DRDA or DDM client after the initial connection is made. The set of all associated jobs, the listener and the server jobs, are collectively referred to as the DRDA server.

18.3.1.1 Configuring the DRDA server

Configuring the DRDA server is very similar to configuring the DDM, because the DDM server is used to act as a DRDA application server.

The same steps are involved in setting up the DRDA server:

1. Add the host table entry.
2. Change the DDM TCP/IP Attributes.
3. Start the DDM Server.

See 18.2.1.1, “Configuring the DDM server” on page 641, for more information on configuring and starting the DDM/DRDA server.

18.3.2 DRDA requester

The AS/400 can act as a DRDA application requester. When configured as an application requester, programs and commands running on the AS/400 requester

can access relational databases on any DRDA compliant server (such as another AS/400 system or a S/390 running DB/2).

18.3.2.1 Configuring the DRDA requester

The steps required to configure the DRDA requester are explained in the following sections.

Adding the host table entry

You add the host table entry on the DRDA requester the same way that you added the entry on the DDM Server (see 18.2.1.1, “Configuring the DDM server” on page 641).

The host table entry allows the AS/400 requester to identify the DRDA server and associate the appropriate IP address with the server.

Adding the AS/400 system to the relational database directory

Once the DRDA requester knows the address of the DRDA server, an entry is added to the Relational Database Directory using the Add RDB Directory (ADDRDBDIRE) command. Figure 796 shows the parameters for this command.

```
                    Add RDB Directory Entry (ADDRDBDIRE)

Type choices, press Enter.

Relational database  . . . . . > AS22
Remote location:
  Name or address  . . . . . > '10.5.69.233'



  Type . . . . . > *IP           *SNA, *IP
Text . . . . . > 'RDB Entry for AS22'

Port number or service program     *DRDA









                                                                      Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 796. Add RDB Directory Entry (ADDRDBDIRE) parameters

Defining a user profile for DRDA over TCP/IP

You can use the Add Server Authenticity (ADDSVRAUTE) command to add authentication information for a given user under which the connect is being done. The user ID and password are associated with the user profile and remote application server. This information flows to the application server (AS) each time the application requester (AR) issues a connect request.

1. Before using the ADDSVRAUTE command, make sure that you have *SECADM special authority and *OBJMGT and *USE authorities to the user profile to which the server authentication entry is being added.

2. Check whether the retain server security data (QRETSVRSEC) system value is set to 1. If the value is 0 (do not retain data), the password is not saved in the entry.
3. Add the authentication entry as shown in Figure 797 and Figure 798.

```
                    Add Server Auth Entry (ADDSVRAUTE)

Type choices, press Enter.

User profile . . . . . .   itscid57     Name, *CURRENT
Server . . . . . .         as22


User ID  . . . . . .       *USRPRF

                                                                    ...


                                                                  More...
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 797. Adding the authentication entry (Part 1)

```
                    Add Server Auth Entry (ADDSVRAUTE)

Type choices, press Enter.

User password  . . . . . .   *NONE

                                                                    ...


                                                                  Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 798. Adding the authentication entry (Part 2)

Note

Make sure the server name is in upper case. A generic authorization entry can be entered by using the ADDSVRAUTE command by specifying *ANY for the server name. *ANY causes connects to any RDB for which there is no specific authorization entry to use the user ID and password contained in the *ANY entry. If the default value of *NONE is left for the password parameter, only the user ID flows unless it is overridden by a USING parameter on the connect statement.

18.4 Writing a DDM or DRDA requester or server

DDM and DRDA are open standards and clients and servers for these standards exist on many platforms. If you want to write your own client or server, you should consult these manuals (see Appendix C, “Related publications” on page 695, for more information):

- *DRDA Volume 1: Distributed Relational Database Architecture (DRDA)*
- *DRDA Volume 2: Formatted Data Object Content Architecture*
- *DRDA Volume 3: Distributed Data Management (DDM) Architecture*

18.5 Examples of using DRDA over TCP/IP

The discussion given in the previous section contains details about the method of configuring DRDA over TCP/IP. In this section, we discuss the details of how to actually access the data in the remote server. We examine two different scenarios:

- Using interactive SQL
- Using C programming

18.5.1 Interactive SQL example

Probably the easiest way to take advantage of a DRDA connection to a remote database is to use interactive SQL. The following simple SQL session documents all major points to be remembered while running DRDA over TCP/IP.

Start interactive SQL with the Start SQL (STRSQL) command. Make sure that the commitment control level you are running at is at least *CHG. At the SQL prompt, press the F13 key, enter option `1`. Change the Commitment Control Attribute to `*CHG`. Return to the SQL session by pressing F3.

If you encounter an error when using the interface SQL session to test DRDA, consult 19.3, “DDM/DRDA problem determination” on page 664. You can display extra information about errors by positioning the cursor on the error message and pressing F1.

Now you are ready to view the statements shown in Figure 799.

```
                          Enter SQL Statements

Type SQL statement, press Enter.
   > release all                                                  (1)
     RELEASE of all relational databases completed.
   > commit                                                       (2)
     Commit completed.
   > connect to as22
     Current connection is to relational database AS22.         (3)
     Session attributes changed by the database manager.
   > select srcdta from qgpl/qddssrc where srcseq=1.00           (4)
     SELECT statement run complete.
===>

                                                                      Bottom
F3=Exit   F4=Prompt   F6=Insert line   F9=Retrieve   F10=Copy line
F12=Cancel            F13=Services     F24=More keys
```

Figure 799. Interactive SQL using DRDA to access a remote database

When you start the interactive SQL session, you are connected to the local database. Issue the command `release all` (1) to release the connection to the local database.

Then, issue the `commit` (2) command to move the connection from a released state to an unconnected state.

Once you are connected to the remote database, you can position the cursor on the connection message (3) and press F1. This shows additional information about message SQL7971 that includes details of the connection (Figure 801 on page 654 and Figure 802 on page 655). Connection type one is required to allow committable updates.

Once the connection is made, you can issue commands to display and update the remote database. A select state is used in this example (4) to display the first record in a remote database file.

The results of the select statement are shown in Figure 800 on page 654.

```
                                Display Data

Data width . . . . . :      80
 Position to line  . . . . .
Shift to column  . . . . . .

....+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8
 SRCDTA
   * START OF SPECIFICATIONS ****************************
******** End of data ********
```

Figure 800. Select statement results display

```
                        Additional Message Information

 Message ID . . . . . :   SQL7971


 Message . . . . :   Current connection is to relational database AS22.

 Cause . . . . . :   The product identification is QSQ04030, the server class
   name is QAS, and the user ID is ITSCID57. The connection method used is
   *DUW. The connection type is 1.  A list of the connection types follows:
     -- Type 1 indicates that committable updates can be performed and either
   the connection uses an unprotected conversation, is a connection to an
   application requester driver program using *RUW connection method, or is a
   local connection using *RUW connection method.
     -- Type 2 indicates that the conversation is unprotected and no
   committable updates can be performed.
     -- Type 3 indicates that the conversation is protected and it is unknown
   if committable updates can be performed.
                                                                       More...
 Press Enter to continue.

 F1=Help   F3=Exit   F6=Print      F9=Display message details
 F10=Display messages in job log   F12=Cancel   F21=Select assistance level
```

Figure 801. Connection type for DRDA over TCP/IP (Part 1)

```
                        Additional Message Information

Message ID . . . . . :   SQL7971


   -- Type 4 indicates that the conversation is unprotected and it is unknown
  if committable updates can be performed.
   -- Type 5 indicates that it is unknown if committable updates can be
  performed and the connection is either a local connection using *DUW
  connection method or a connection to an application requester driver program
  using *DUW connection method.
   If the relational database is *N, then the *LOCAL entry has not been added
  to the relational database directory.









                                                                        Bottom
Press Enter to continue.

F1=Help   F3=Exit   F6=Print      F9=Display message details
F10=Display messages in job log   F12=Cancel   F21=Select assistance level
```

Figure 802. Connection type for DRDA over TCP/IP (Part 2)

18.5.2 ILE C example

Coding your application program, which accesses a remote AS/400 system with DRDA over TCP/IP support, is similar to the procedure described for the Interactive SQL session. The following example code highlights the most important considerations:

- Compile your program with an isolation level *CHG or above. If you have a connection to the remote AS/400 system at compile time, you may create an appropriate SQL package on the target system. The following CRTSQLCI command creates the program object on the local system and the SQL package at the remote server:

  ```
  CRTSQLCI OBJ(QUSRSYS/DRDATST) SRCFILE(QGPL/QCSRC) SRCMBR(DRDATST) RDB(AS22)
  OBJTYPE(*PGM) OUTPUT(*PRINT) RDBCNNMTH(*RUW)
  ```

- The code in Figure 803 on page 656 and Figure 804 on page 657 shows the sample program written in ILE C code that uses DRDA over TCP/IP.

```
/*********************************************************************
 *                                                                   *
 * DRDATST.C                                                         *
 *                                                                   *
 * This sample uses DRDA to access a database on another AS/400.     *
 * The DRDA definition uses TCP/IP and was added with the command:   *
 *                                                                   *
 *     ADDRDBDIRE RDB(AS22) RMTLOCNAME('10.5.69.233' *IP)            *
 *     TEXT('RBD Entry for AS22')                                    *
 *                                                                   *
 * An authentication entry was then added with the command:          *
 *                                                                   *
 *     ADDSVRAUTE USRPRF(ITSCID57) SERVER(AS22)                      *
 *                                                                   *
 *********************************************************************/


#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <decimal.h>

EXEC SQL BEGIN DECLARE SECTION;
char Src_data[133];
EXEC SQL END DECLARE SECTION;

EXEC SQL INCLUDE SQLCA;

void show_error (char *prompt, int sqlcode, char *sqlerrmc);

void main()
{
    EXEC SQL
    release all;

    if (sqlca.sqlcode != 0)
        show_error ("Error occurred in the release of databases\n",
                    sqlca.sqlcode,
                    sqlca.sqlerrmc);

    printf( "Released all Databases...\n" );

    EXEC SQL
    commit;

    if (sqlca.sqlcode != 0)
        show_error ("Error occurred in commit release of database\n",
                    sqlca.sqlcode,
                    sqlca.sqlerrmc);

    EXEC SQL
    connect to AS22;

    if (sqlca.sqlcode != 0)
        show_error ("Error occurred connecting to database\n",
                    sqlca.sqlcode,
                    sqlca.sqlerrmc);

    printf( "Successfully connected to AS22..\n" ) ;

    EXEC SQL
    commit;
```

Figure 803. DRDA test program (Part 1)

```
    if (sqlca.sqlcode != 0)
        show_error ("Error occurred in commit of connection\n",
                    sqlca.sqlcode,
                    sqlca.sqlerrmc);

    printf( "Committed the Connection...\n" );

    EXEC SQL
    select srcdta
    into :Src_data
    from qgpl/qddssrc
    where srcseq = 1.00;

    if (sqlca.sqlcode != 0)
        show_error ("Error occurred in selecting record\n",
                    sqlca.sqlcode,
                    sqlca.sqlerrmc);

    printf ("Srcdta = \"%s\"\n", Src_data);

    exit(0);
}

void show_error (char *prompt, int sqlcode, char *sqlerrmc)
{
    printf(prompt);
    printf("The SQLCODE is %d\n" , sqlcode );
    printf("The Error Message :\n%s\n", sqlerrmc );
    exit( -1 );
}
```

Figure 804. DRDA test program (Part 2)

- After you compile and create the program, run the program with the command:

  ```
  CALL QUSRSYS/DRDATST
  ```

  Sample output is shown in Figure 805 on page 658.

```
  Released all Databases...
  Successfully connected to AS22..
  Commited the Connection...
  Srcdta = "      *  START OF SPECIFICATIONS   ******************************
           "
  Press ENTER to end terminal session.









===>

F3=Exit F4=End of File F6=Print F9=Retrieve F17=Top
F18=Bottom  F19=Left   F20=Right F21=User Window
```

Figure 805. Sample ILE C program results display

Chapter 19. Problem determination

This chapter contains information about problem determination for TCP/IP.

19.1 Telnet printer emulation problem determination

If you are facing problems with a PC5250 printer emulation print job, you have to follow these steps:

1. Pause the print queue on the PC.
2. Start a PC5250 trace.
3. Disconnect and reconnect the PC5250 printer emulation session.
4. Send the problem job.
5. Stop the trace when the spooling activity ceases.
6. From the folder Windows/Spool/printerID, you can save the files ending with .SHD and .SPL to diskette.
7. You can look at the .SPL file with a text editor
8. If there are any .TMP files in the file, you may save those too.
9. Release the print queue, let the job print, and save the paper output.
10. You can send the trace, the diskette, any dump information if your PC crashed, as well as the AS/400 spooled file, to IBM Support.

19.2 TCP/IP printing problem determination

The following section discusses some of the errors that may occur when trying to print over a TCP/IP network. There are too many possible errors to list them all.

If you get an error that is not listed here, you can obtain more information on the error by placing the cursor on the error line and pressing F1. The additional information displayed contains a more detailed description of the probable causes and suggests actions to fix the problem.

19.2.1 LPR/LPD printing problems

Error TCP3719 (Figure 806 on page 660) is produced when you try to print a file to an LPD server and the request is rejected. This is often caused by an error with the name of the printer queue. See 13.1.2.1, “Using the LPR command to send the file to the LPD server” on page 482, for information on the correct settings for LPR or SNDTCPSPLF command parameters.

To correct this error, find the correct queue name for the LPD server you are using and specify it in the LPR command.

```
                    Additional Message Information

Message ID . . . . . :   TCP3719
Date sent  . . . . . :   12/01/98      Time sent  . . . . . :   13:47:44

Message . . . . :   Send request failed for spooled file QSYSPRT.

Cause . . . . . :   The send request for spooled file QSYSPRT, number 1 for
  job 002485/ITSCID57/QPADEV0003 failed. The print job request is not valid
  for the Line Printer Daemon (LPD) server on the remote system.
Recovery  . . . :   Verify that the printer queue is valid on the remote
  system.  More detailed information about the error may be logged for the LPD
  server on the remote system.




                                                                       Bottom
Press Enter to continue.

F1=Help   F3=Exit   F6=Print      F9=Display message details
F10=Display messages in job log   F12=Cancel   F21=Select assistance level
```

Figure 806. Error description TCP3719

Another error that can occur when trying to use LPR to send a file to an LPD server is TCP3436. Figure 807 shows the additional information for this error.

This error can be caused by:

- The IP address specified for the remote system is incorrect.
- The remote system is not currently available.

```
                    Additional Message Information

Message ID . . . . . :   TCP3436      Severity . . . . . :   10
Message type . . . . . :   Diagnostic
Date sent  . . . . . :   12/16/98      Time sent  . . . . . :   15:55:54

Message . . . . :   No response from remote host system within open time-out.
Cause . . . . . :   An open request was sent to a remote host system, but that
  host system did not respond to the request before the open time-out expired.
  This may be due to the fact that the TCP/IP services are not currently
  available on the remote host system.
Recovery  . . . :   Try the request again or contact the system administrator
  to start the TCP/IP services on the remote host system.




                                                                       Bottom
Press Enter to continue.

F3=Exit   F6=Print   F9=Display message details
F10=Display messages in job log   F12=Cancel   F21=Select assistance level
```

Figure 807. Error description TCP3436

Another error that can occur when trying to use LPR to send a file is TCP3427 (Figure 808). This error occurs when the remote host is not running an LPD server or is not accepting connections to the LPD server.

Ensure that you have specified the correct IP address and that the remote LPD server is ready to accept new connections.

```
                      Additional Message Information

Message ID . . . . . :   TCP3427      Severity . . . . . :   10
Message type . . . . . :   Diagnostic
Date sent  . . . . . :   12/16/98     Time sent  . . . . . :   16:04:00

Message . . . . :   Remote host system rejected the open attempt.
Cause . . . . . :   This may have occurred because the remote host does not
  have ports available for use or does not support TELNET.
Recovery  . . . :   Try the request again or contact the system administrator.









                                                                      Bottom
Press Enter to continue.

F3=Exit   F6=Print   F9=Display message details
F10=Display messages in job log   F12=Cancel   F21=Select assistance level
```

Figure 808. Error description TCP3427

19.2.2 TCP/IP printer driver problems

The most common error message when trying start a printer using the direct TCP/IP printer driver is CPD337F. Figure 809 and Figure 810 on page 662 show the Additional Message Information displays for this message.

```
                      Additional Message Information

Message ID . . . . . :   CPD337F
Date sent  . . . . . :   12/16/98      Time sent  . . . . . :   14:49:34

Message . . . . :   Remote device rejected an attempt by the writer to open a
  connection.

Cause . . . . . :   This error may have occurred for the following reasons.
    1) An incorrect port number or internet address was specified
       in the printer device description.
    2) The remote device, if it is a printer, may be busy printing
       from another host
    3) The remote device, if it is a printer, may not be ready to
       print and may require operator intervention.
Recovery  . . . :   Do one of the following:
    1) Verify that the correct port number and internet address was
       specified in the printer device description. If a remote location
                                                                        More...
Press Enter to continue.

F1=Help   F3=Exit   F6=Print   F9=Display message details   F12=Cancel
F21=Select assistance level
```

Figure 809. Error CPD337F additional message information (Part 1)

```
                      Additional Message Information

Message ID . . . . . :   CPD337F
Date sent  . . . . . :   12/16/98      Time sent  . . . . . :   14:49:34

       name was used, verify that the internet address for this location
       name is correct. Start the writer if it has ended and try printin
       again.
    2) No recovery action is required. Printing will continue when
       the writer is able to use the printer.
    3) Ready the printer. Printing will continue when the printer
       becomes ready to print.









                                                                      Bottom
Press Enter to continue.

F1=Help   F3=Exit   F6=Print   F9=Display message details   F12=Cancel
F21=Select assistance level
```

Figure 810. Error CPD337F additional message information (Part 2)

This message has multiple possible causes. The most common causes are:

- The IP address of the printer is specified incorrectly or is unreachable.
- The port number specified for the printer is incorrect.
- The printer is not configured to accept direct TCP/IP connections for PCL/PJL print jobs on the port specified.

To correct this error, you should:

1. End the printer device with the command:

   ```
   ENDWTR WTR(TCPPRINTER) OPTION(*IMMED)
   ```

2. Enter the command:

   ```
   WRKCFGSTS CFGTYPE(*DEV) CFGD(*PRT)
   ```

   This brings up the display shown in Figure 811. From this display, you enter option `2` for the printer you are working with to vary the printer off. Then, enter option `8` to work with the printer's description. This produces the display shown in Figure 812 on page 664. From this display, select option `2` change the device description.

3. Check that the device description is using the correct IP address and port number for the printer you are using. If these values are correct, use your printer management software to check that the printer itself is correctly configured to use the IP address and port number for PCL/PJL printing.

   Once the printer device description has been changed and saved, vary the printer back on from the WRKCFGSTS screen.

4. Restart the printer with the command:

   ```
   STRPRTWTR DEV(TCPPRINTER)
   ```

```
                      Work with Configuration Status                AS21
                                                          12/16/98  15:16:12
Position to  . . . . .               Starting characters

Type options, press Enter.
  1=Vary on   2=Vary off   5=Work with job   8=Work with description
  9=Display mode status    13=Work with APPN status...

Opt  Description       Status               -------------Job--------------
     TCPPRINTER       ACTIVE/WRITER        TCPPRINTER  QSPLJOB     003825
                                                                      Bottom
Parameters or command
===>
F3=Exit   F4=Prompt   F12=Cancel   F23=More options   F24=More keys
```

Figure 811. Work with Configuration Status display for the printer device

```
                        Work with Device Descriptions
                                                                System:   AS21
Position to  . . . . .              Starting characters

Type options, press Enter.
  2=Change   3=Copy    4=Delete   5=Display   6=Print   7=Rename
  8=Work with status   9=Retrieve source

Opt  Device      Type        Text
     TCPPRINTER  3812

                                                                        Bottom
Parameters or command
===>
F3=Exit   F4=Prompt   F5=Refresh   F6=Create   F9=Retrieve   F12=Cancel
F14=Work with status
```

Figure 812. Work with Device Description display for the printer device

19.3 DDM/DRDA problem determination

There are many errors that can occur when using DDM or DRDA to access files on a remote machine. There are many common errors, some of which are listed in this section.

If you get an error that is not listed here, you can obtain more information on the error by placing the cursor on the error line and pressing F1. The additional information displayed contains a more detailed description of the probable causes and suggested actions to fix the problem.

19.3.1 Connection errors with DDM and DRDA over TCP/IP

Error CPD3E37 occurs if you try to connect to a DRDA server that references a host name that does not exist in the TCP/IP Host Table and cannot be resolved by a DNS server (Figure 813 and Figure 814).

```
                        Additional Message Information

Message ID . . . . . :   CPD3E37      Severity . . . . . :   40
Message type . . . . . :   Diagnostic
Date sent  . . . . . :   12/15/98     Time sent  . . . . . :   16:16:37

Message . . . . :   Attempt to map host name to IP address failed with reason
  code 5.
Cause . . . . . :   An error with reason code 5 occurred when attempting to
  map the host name as22 to an IP address.  Reason code meanings are listed
  below.
     5 -- Host name was not found
    10 -- No address corresponding to host name
    15 -- Unrecoverable error getting host name
    20 -- No reply from name server - try again later
Recovery  . . . :   Using the reason code, determine the cause of the error;
  if necessary, use WRKRDBDIRE to correct the host name in the RDB directory.
Technical description . . . . . :   A failure occurred using the sockets
                                                                      More...
Press Enter to continue.

F3=Exit   F6=Print   F9=Display message details
F10=Display messages in job log   F12=Cancel   F21=Select assistance level
```

Figure 813. Host alias not found from DDM file definition (Part 1)

```
                        Additional Message Information

Message ID . . . . . :   CPD3E37      Severity . . . . . :   40
Message type . . . . . :   Diagnostic

  gethostbyname call.  The reason code is the h_errno code returned by
  gethostbyname.









                                                                      Bottom
Press Enter to continue.

F3=Exit   F6=Print   F9=Display message details
F10=Display messages in job log   F12=Cancel   F21=Select assistance level
```

Figure 814. Host alias not found from DDM file definition (Part 2)

To fix this problem, be sure to verify these points:

- The RBD entry listed by the WRKRDBDIRE command contains the correct host name.
- The host name is defined in the TCP/IP Host Table (entering option `10` on the CFGTCP menu) or is resolved by a DNS server.
- The DNS (if used) is available. If the DNS server is not on the AS/400 server, ensure that communications between the DNS server and the AS/400 server are functioning properly.

19.3.2 DRDA errors

The error shown in Figure 815 occurs if the DDM server is not started on the DRDA application server. If you position the cursor on the error message and press F1, you get extended error message information for error code SQL30080 as shown in Figure 816 and Figure 817.

The additional information display shows other possible causes of the error, such as a communications problem between the requester and the server.

```
                              Enter SQL Statements

Type SQL statement, press Enter.
   > release all
     RELEASE of all relational databases completed.
   > commit
     Application process not in a connected state.
   > connect to as22
     Communication error occurred during distributed database processing.
===> connect to as22









                                                                        Bottom
F3=Exit   F4=Prompt   F6=Insert line   F9=Retrieve   F10=Copy line
F12=Cancel            F13=Services     F24=More keys
```

Figure 815. Error displayed in an interactive SQL session using DRDA

```
                        Additional Message Information

Message ID . . . . . :   SQ30080       Severity . . . . . :   30
Message type . . . . . :   Diagnostic

Message . . . . :   Communication error occurred during distributed database
  processing.
Cause . . . . . :   A communication error occurred. A possible list of reasons
  may include:
    -- The remote system is not available.
    -- The communications network is not available.
    -- The userid used to start the connection may not exist on the remote
  system.
    -- The password may not be valid for the userid. The characters and case
  of the password specified must match exactly the password on the remote
  system.  An AS/400 Application Server requires that passwords be specified
  in uppercase.
    The major return code is 00 and the minor return code is 00. If the
                                                                        More...
Press Enter to continue.

F3=Exit   F6=Print   F9=Display message details
F10=Display messages in job log   F12=Cancel   F21=Select assistance level
```

Figure 816. Additional information for error SQ30080 (Part 1)

```
                        Additional Message Information

Message ID . . . . . :   SQ30080       Severity . . . . . :   30
Message type . . . . . :   Diagnostic

  connect attempt used TCP/IP, both return codes will be 00.  If the major and
  minor return codes are not both 00, the meaning of the codes can be found in
  the APPC Programmer's Guide.
Recovery  . . . :   See previous messages for more information.  Check the
  status of the remote system and the communications network for possible
  problems. If the application server is an AS/400, check QSYSOPR message
  queue for error messages.









                                                                        Bottom
Press Enter to continue.

F3=Exit   F6=Print   F9=Display message details
F10=Display messages in job log   F12=Cancel   F21=Select assistance level
```

Figure 817. Additional information for error SQ30080 (Part 2)

Figure 818 on page 668 shows the error produced in an interactive SQL session when the user ID is not defined on the remote DRDA server. Figure 819 on page 668 shows additional information on the error received.

```
                          Enter SQL Statements

Type SQL statement, press Enter.
   > release all
     RELEASE of all relational databases completed.
   > commit
     Application process not in a connected state.
   > connect to as22
     Authorization failure on distributed database connection attempt.
===>

                                                                      Bottom
F3=Exit   F4=Prompt   F6=Insert line   F9=Retrieve   F10=Copy line
F12=Cancel            F13=Services     F24=More keys
```

Figure 818. Error displayed in an interactive SQL session using DRDA

```
                      Additional Message Information

Message ID . . . . . :   CPF9190       Severity . . . . . :   40
Message type . . . . . :   Diagnostic
Date sent  . . . . . :   12/14/98      Time sent  . . . . . :   11:58:41

Message . . . . :   Authorization failure on DDM TCP/IP connection attempt.
Cause . . . . . :   A connection attempt failed with reason code 5. The reason
  codes and their meanings are as follows:
    0 -- Unknown cause.
    1 -- Password expired.
    2 -- Password not valid.
    3 -- Password missing.
    4 -- Protocol violation.
    5 -- User ID not found.
    6 -- User ID not valid. For a DB2/400 server this could mean a damaged
  user profile or PASSWORD(*NONE).
    7 -- User ID revoked or disabled.
                                                                      More...
Press Enter to continue.

F3=Exit   F6=Print   F9=Display message details
F10=Display messages in job log   F12=Cancel   F21=Select assistance level
```

Figure 819. Additional information for error CPF9190

19.4 DHCP server logging and problem determination

You can enable logging through a configuration option within Operations Navigator. Select the **Logging** tab on the DHCP Server Properties display. You can even specify the type of logging that you want to perform, depending on the types of things that you want to log.

Figure 820 shows how to enable logging on the AS/400 DHCP server.

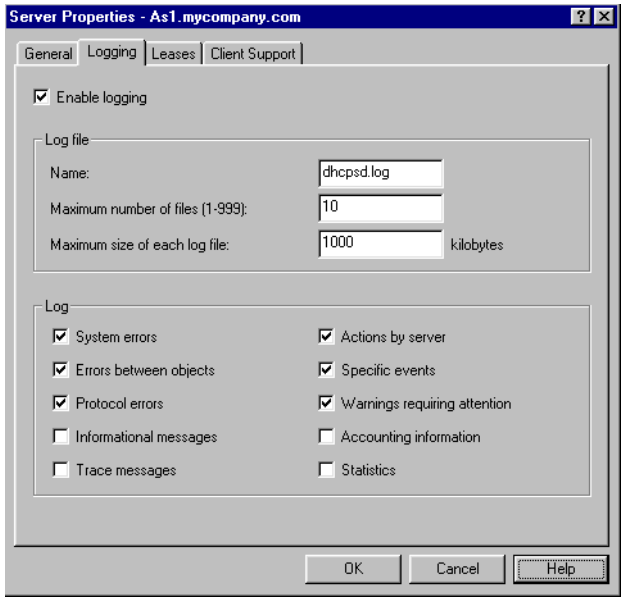


Figure 820. DHCP server logging configuration

The following files in the IFS directory /QIBM/UserData/OS400/DHCP are helpful logs that can be used for problem determination reasons.

- **dhcpsd.log**

  This file is used as the default logging/tracing file in case of a regular DHCP server.

- **dhcprd.log:**

  This file is used as the default logging/tracing file in case of a BOOTP/DHCP Relay Agent.

Note

You can configure this logging trace file to roll into multiple files based on the maximum size.

Figure 821 on page 670 provides an overview of the DHCP server jobs, files, and logs.

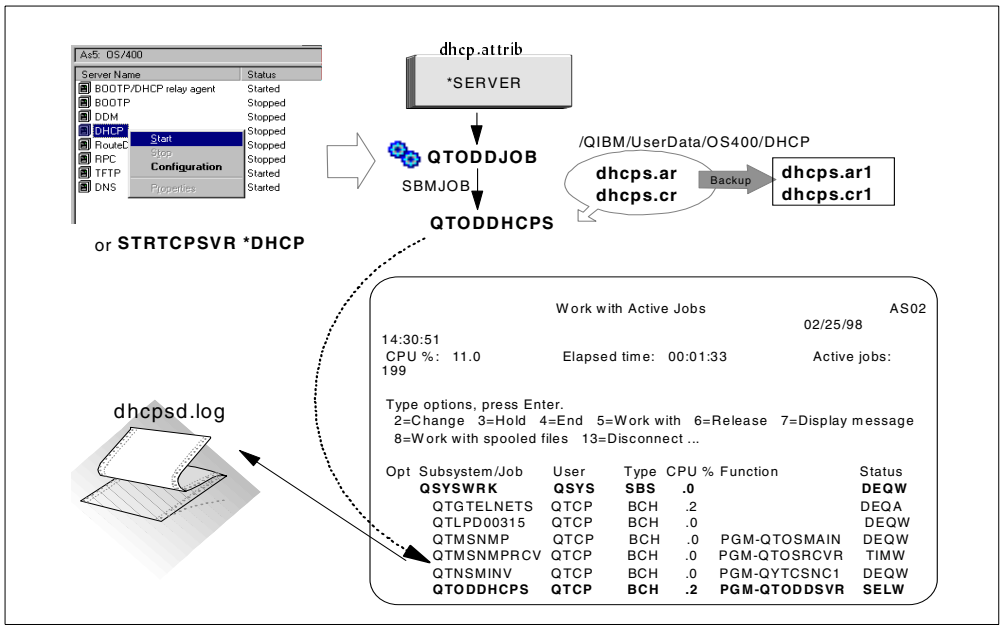


Figure 821. AS/400 DHCP server jobs, files, and logs

Figure 822 provides an overview of the BOOTP/DHCP Relay Agent jobs, files, and logs.

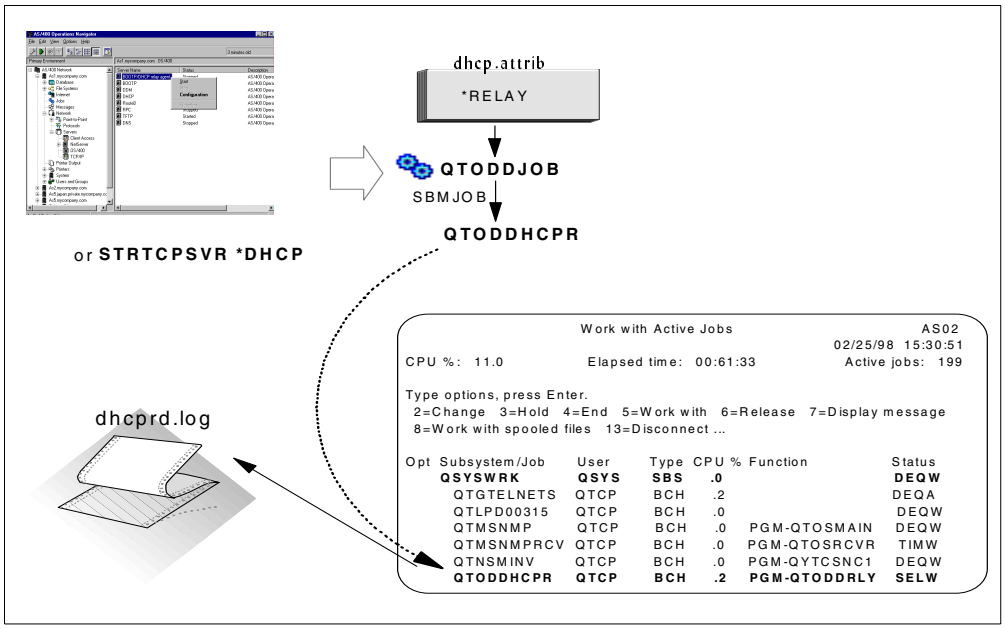


Figure 822. BOOTP/DHCP Relay Agent jobs, files, and logs

19.5 DNS server problem determination

We describe how to prevent DNS server problems and how to identify problem symptoms. Several tools are available to a DNS administrator to troubleshoot those DNS problems.

19.5.1 Tips for preventing problems

This section offers ways to prevent problems from occurring. We found these tips while we were writing this book. The list is not all inclusive.

IFS file ownership

When the Operations Navigator GUI creates DNS files in the IFS directory /QIBM/UserData/OS400/DNS, the file is created with the owner value set to the AS/400 user profile that the user used to connect. If this user profile is deleted, all objects owned by this user profile are also deleted.

Tip

To prevent accidentally deleting the DNS files (if the user's user profile is deleted), change the ownership of the files to a system-supplied user profile such as QTCP.

Restarting the DNS server

As a DNS server queries other name servers for information for which it is not authoritative, it stores the answers in its cache. That way if another client asks for the same query, the name server can supply the response from its cache instead of querying the authoritative name server again.

Completely stopping and starting the DNS server clears the cache. Therefore, to keep a DNS server's cache rich with information, avoid stopping and starting the name server needlessly. If a configuration change is made, the smart icon "update server" should be used to update the configuration without stopping and starting the name server. The "update server" smart icon does not clear the name server's cache.

Manually editing the DB files in DNS IFS directory

Every domain file configured with Operations Navigator's DNS configuration results in the creation of a file with a .DB extension in the IFS directory /QIBM/UserData/OS400/DNS. We recommend that you *do not* manually edit these files. If changes need to be made, use the Operations Navigator DNS configuration displays. The reasons for this recommendation are:

- Every DB file contains an Start of Authority (SOA) record. This record contains the serial number that secondary name servers check to make sure their secondary domain files are at the same level as the primary domain files. Therefore, when changes are made to the primary domain files, the Operations Navigator DNS configuration automatically increments this serial number. If the DB files are edited manually, it is up to the DNS administrator to remember to manually increment the serial number in the SOA record.
- When the Operations Navigator DNS configuration is used to add (or delete) hosts to the forward mapping primary domain files, Operations Navigator automatically adds (or deletes) the host to the corresponding reverse mapping (in-addr.arpa) file if the enable Create and delete reverse mappings by default is checked. This saves the DNS administrator configuration time and prevents inadvertently omitting the hosts in the in-addr.arpa file. This convenience is lost when domain files are manually edited. A DNS administrator must remember to add or delete a host from the in-addr.arpa file, as well as the forward mapping file.

- When a change is made in the DB files, the DNS server needs to be stopped and started or restarted to pick up the change. The AS/400 STRTCPSVR SERVER(*DNS) RESTART(*DNS) command stops and starts the DNS server, which causes the cache to be cleared. We *do not* recommend doing this. The recommended method to pick up configuration changes is to use the Operations Navigator DNS configuration “update server” smart icon. The configuration is then refreshed without clearing the cache. Since Operations Navigator should to be used to pick up the configuration changes, the DNS administrator should make the changes with this GUI in the first place.

The LOCALHOST host

Every primary forward mapping domain file created for a DNS server should include the host *localhost* with an IP address of 127.0.0.1. Consequently, every DNS server should have a reverse mapping domain 0.0.127.in-addr.arpa created also.

19.5.2 Problem determination tools

Several tools are available for troubleshooting DNS problems. Some of these tools should only be used if instructed to do so by AS/400 Software Support people.

- Operations Navigator DNS Configuration displays:

  The Operations Navigator DNS Configuration displays should be used to check for completeness and for mistyped domain names, host names, and IP addresses. From Operations Navigator, the DNS administrator can make sure the server has been updated after a DNS configuration change has been made, ensure newly created domains are enabled, and verify the server is started.

- AS/400 job logs:

  After configuring a name server for the first time or after any major configuration changes, always review the QTOBDNS job log for errors after the name server has been started. Many DNS configuration errors cause errors to be posted in this job log. Therefore, this job log is a critical tool when debugging a DNS problem.

- Nslookup interactive tool:

  Nslookup is a way to pose queries to the name server and view its responses interactively. This tool can be useful if you suspect a query is not being resolved the way you think it should be. It is also useful as an informal testing mechanism after a name server is first configured.

- Querylog file in AS/400 IFS:

  This is a log of the queries the name server has received. It can be useful to verify that the query from a client actually made it to the name server (that is, the TCP connectivity exists and the client is indeed sending the query as you expect).

  These tools are discussed in more detail in *AS/400 TCP/IP Autoconfiguration: DNS and DHCP Support,* SG24-5147.

You can find many tools on the Web to test your DNS server and other parts of your TCP/IP configuration. One tool that we used during our testing is *CyberKit*. It

provides several tools including nslookup. For this and many more tools, visit the Web site at: `http://www.tucows.com/`

19.5.3 AS/400 job logs

AS/400 job logs provide a wealth of information about jobs that are running, as well as jobs that have terminated. This sections describes some of the information that can be found in the DNS job logs.

19.5.3.1 The active QTOBDNS job log

With so much time spent on Operations Navigator configuring the AS/400 DNS server, it is easy to overlook one of the most informative logs for DNS problem determination: the AS/400 job log of the DNS job QTOBDNS.

If the DNS server is started, the QTOBDNS job is active and running under the QSYSWRK subsystem. You can locate it with the following AS/400 command:

```
WRKACTJOB SBS(QSYSWRK)
```

Once the QSYSWRK active jobs are displayed, page down until you find the job named QTOBDNS. Enter option `5` in front of this job to work with the job. On the next display, enter option `10` to display the job log. When the job log is shown, press F10 to display the detailed messages. The bottom of the job log is displayed. You may need to page up to find error messages logged at the time you had a problem. If you find an error message in the job log that needs investigating, you can see more details by placing the cursor on the message itself and pressing F1 for help.

Note

When you finish configuring the AS/400 DNS server and start the DNS server for the first time, we highly recommend that you review the QTOBDNS job log for errors that are logged when the DNS server starts. For example, spelling errors in the names of primary domain files and spelling errors in the domain names of hosts cause error messages to be posted to the QTOBDNS job log.

19.5.3.2 The inactive QTOBDNS job log

Sometimes it is necessary to review the job log of QTOBDNS after the DNS has been stopped and the job QTOBDNS has ended. Or, even more importantly, the DNS server is having such a severe problem that the QTOBDNS job starts and ends before you have a chance to review the active QTOBDNS job log.

To find the job log of a job that is no longer active, you need to find the job's spooled output. It helps to know the user under which the job runs. The DNS job runs using the QTCP user profile. Therefore, use the following command:

```
WRKSPLF QTCP
```

If the job has recently ended, the job's spooled file may be near the bottom of the resulting list. Press F18 to go to the bottom of the Work With Spooled Files List. The name of the job is listed under the User Data column.

19.5.4 NSLOOKUP

The AS/400 name server lookup (nslookup) program queries domain name servers in interactive mode. This allows you to query name servers to get information about various hosts and domains, or to display a list of hosts within that domain.

You may also consider using a tool on a PC to do NSLOOKUP functions. One tool that we used during our testing is *CyberKit*.

To access the AS/400 NSLOOKUP function in V4R2, you must make a program call. Use the command:

```
call pgm(qdns/qtoblkup)
```

To access the NSLOOKUP function in V4R3 and beyond, use the NSLOOKUP command. On an AS/400 command line, type `NSLOOKUP` and press Enter. You may also type the AS/400 style command `STRDNSQRY` and press Enter. Either command starts the NSLOOKUP function. To do a simple look-up, type the name and press Enter. The information returned by the DNS displays on the screen.

After entering the command, the AS/400 display should look similar to the example shown in Figure 823.

```
Default Server:  as1.mycompany.com
  Address:  10.5.69.222

  >
===>

F3=Exit F4=End of File F6=Print F9=Retrieve F17=Top
F18=Bottom  F19=Left   F20=Right F21=User Window
```

Figure 823. Initial NSLOOKUP display

Refer to the site at: `http://www.as400.ibm.com/infocenter/` and search for `NSLOOKUP` for more information on the NSLOOKUP program. You can also find information in the book *AS/400 TCP/IP Autoconfiguration: DNS and DHCP Support,* SG24-5147.

Appendix A. Sample code

This appendix includes sample code that is referenced in this book. This code is available in a save file that can be downloaded from the redbooks site at:
http://www.redbooks.ibm.com

A.1 Getting the material to your system

The materials included in the additional materials section of the site are provided "as is." That means that no support is available for them.

Go to the ITSO Web site at: http://www.redbooks.ibm.com

Select the **Additional Materials** tab. Scroll down the page until you find the **SG245190** directory. Review the readme.txt file for information about how to get the additional materials to your system.

A.2 IP address mapping sample source code

This sample program in 13.5.5, "Using an initial program to map printer IP addresses" on page 514, can change a printer's IP address to match the clients IP address:

```
/* Module Description ************************************************/
/*                                                                   */
/*********************************************************************/
/*                                                                   */
/*                                                                   */
/*********************************************************************/
/*                                                                   */
/* Source File Name: qrmtwtr.c                                       */
/*                                                                   */
/* Module Name: Telnet Remote *OUTQ IP Initialization program.       */
/*                                                                   */
/* Source File Description:                                          */
/*                                                                   */
/*   This module contains functions to let Telnet clients sign on    */
/*   to an AS/400 and optionally run this program via the user       */
/*   profile INLPGM parameter, setting the queue and IP address      */
/*   of the default print *OUTQ based upon their workstation IP      */
/*   address. A mapping table is used to associate printer           */
/*   information with client IP addresses.                           */
/*                                                                   */
/*   If this program is not being called manually, it's              */
/*   expected a user profile calls this program via the INLPGM       */
/*   parameter at sign-on time. The profile should be defined with   */
/*   a private or dedicated output queue (to avoid conflicts with    */
/*   a shared or system output queue):                               */
/*                                                                   */
/*      QSYS/CHGUSRPRF USRPRF(USRPRFNM) INLPGM(*LIBL/QRMTWTR)        */
/*      OUTQ(QUSRSYS/USRPRFNM)                                       */
/*                                                                   */
/*   The custom output queue should be defined as follows:           */
/*                                                                   */
/*      QSYS/CRTOUTQ OUTQ(QUSRSYS/USRPRFNM) RMTSYS(*INTNETADR)       */
/*      RMTPRTQ('lpt1') AUTOSTRWTR(1) CNNTYPE(*IP) DESTTYPE(*OTHER)  */
/*      TRANSFORM(*YES) MFRTYPMDL(*IBM42071)                         */
/*      INTNETADR('1.2.3.4')                                         */
/*                                                                   */
/*   If the default OUTQ name on the user profile definition is not  */
/*   the same as the user profile name, this program will simply     */
/*   exit.                                                           */
/*                                                                   */
/*   If the output queue name matches the user profile name, the     */
/*   writer is ended, the IP address is changed, and the writer      */
/*   is restarted as part of the change command. The following       */
```

```
/*   sequence then occurs:                                              */
/*                                                                      */
/*      QSYS/ENDWTR WTR(USRPRFNM) OPTION(*IMMED)                        */
/*                                                                      */
/*      QSYS/CHGOUTQ OUTQ(QUSRSYS/USRPRFNM) RMTSYS(*INTNETADR)          */
/*      RMTPRTQ('printer') AUTOSTRWTR(1) CNNTYPE(*IP)                   */
/*      INTNETADR('x.x.x.x') TRANSFORM(xform) MFRTYPMDL(mfgtype)        */
/*      DESTTYPE(desttype)                                              */
/*                                                                      */
/*   where 'x.x.x.x', 'printer', 'mfgtype', 'desttype' and 'xform'      */
/*   values are from mapping table.   If no entry is found in mapping   */
/*   table, this program simply exits without changing the OUTQ.        */
/*                                                                      */
/* End Module Description ***************************************************/

#pragma comment (copyright, \
 "(C) Copyright IBM Corp. 1997.\
 All rights reserved.\
 US Government Users Restricted Rights -\
 Use, duplication or disclosure restricted\
 by GSA ADP Schedule Contract with IBM Corp.\
 Licensed Materials - Property of IBM")

#define _QRMTWTR_C

/*****************************************************************************/
/* All file scoped includes go here                                          */
/*****************************************************************************/
#include <recio.h>              /* _Ropen, _Rwrite, _Rreadf,          */
#include <stdio.h>              /* sprintf, printf, fopen, fread,     */
#include <time.h>               /* time(), ctime()                    */
#include <stdlib.h>             /* system, atoi, free, malloc, exit,  */
#include <string.h>             /* strxxx, memxxx                     */
#include <ctype.h>              /* isspace, toupper                   */
#include <except.h>             /* _CNL_Hndlr_Parms_T structure       */
#include <signal.h>             /* _GetExcData(), signal()            */
#include <stdarg.h>             /* va_start(), va_arg(), va_end()     */

#ifdef __ILEC400__
  #include <qusec.h>            /* Include for API error code struct */
  #include <unistd.h>           /* File system APIs (POSIX)          */
  #include <qdcrdevd.h>         /* QDCRDEVD - Retrieve Device Desc   */
  #include <qcapcmd.h>          /* QCAPCMD - Process CL commands     */
  #include <qmhrtvm.h>          /* QMHRTVM - retrieve program message*/
  #include <qsyrusri.h>         /* Qsy_USRI0300_T - user info        */
  #include <qusrjobi.h>         /* Qwc_JOBI0100_T - job info         */
  #include <qmhrtvm.h>          /* QMHRTVM - retrieve program msg    */
  #include <qmhchgem.h>         /* QMHCHGEM - change exception msg   */
  #include <qmhsndpm.h>         /* QMHSNDPM - send program message   */
#else
  #include <qusec.cleinc>       /* Include for API error code struct */
  #include <unistd.h>           /* File system APIs (POSIX)          */
  #include <qdcrdevd.cleinc>    /* QDCRDEVD - Retrieve Device Desc   */
  #include <qcapcmd.cleinc>     /* QCAPCMD - Process CL commands     */
  #include <qmhrtvm.cleinc>     /* QMHRTVM - retrieve program message*/
  #include <qsyrusri.cleinc>    /* Qsy_USRI0300_T - user info        */
  #include <qusrjobi.cleinc>    /* Qwc_JOBI0100_T - job info         */
  #include <qmhrtvm.cleinc>     /* QMHRTVM - retrieve program msg    */
  #include <qmhchgem.cleinc>    /* QMHCHGEM - change exception msg   */
  #include <qmhsndpm.cleinc>    /* QMHSNDPM - send program message   */
#endif

/*****************************************************************************/
/* All file scoped constants go here                                         */
/*****************************************************************************/
const int fTrue = 1;
const int fFalse = 0;

/*****************************************************************************/
/* All file scoped type declarations go here                                 */
/*****************************************************************************/
typedef struct _ERRSTRUCT {
  int  Bytes_Provided;
  int  Bytes_Available;
  char Exception_Id[7];
  char Reserved;
  char Exception_Data[256];
} ERRSTRUCT;
```

```
typedef ERRSTRUCT *PERRSTRUCT;

static ERRSTRUCT esErrCode;

/*********************************************************************/
/* All file scoped macro invocations go here                         */
/*********************************************************************/
#ifndef MAX
 #define MAX(a,b) (((a) > (b)) ? (a) : (b))
#endif

#ifndef MIN
 #define MIN(a,b) (((a) < (b)) ? (a) : (b))
#endif

/*********************************************************************/
/* If the DEBUG preprocessor definition is not defined, the          */
/* buffer() and record() functions will not write anything to        */
/* standard output.                                                  */
/*********************************************************************/
#define DEBUG

/*********************************************************************/
/* Some message API definitions                                      */
/*********************************************************************/
#define MSG_DIAG       "*DIAG     "
#define MSG_ESCAPE     "*ESCAPE   "
#define MSG_INFO       "*INFO     "
#define MSG_INQ        "*INQ      "
#define MSG_COMP       "*COMP     "
#define MSG_NOTIFY     "*NOTIFY   "
#define MSG_RQS        "*RQS      "
#define MSG_STATUS     "*STATUS   "

#define MSQ_Q_CUR_PROG "*         "
#define MSQ_Q_SYSOPR   "*SYSOPR             "

#define QTCPMSG    "QTCPMSG   *LIBL     "            /* Stack MSGF    */
#define QTCPMSGF   "QTCPMSGF  *LIBL     "            /* Apps MSGF     */
#define QCPFMSG    "QCPFMSG   *LIBL     "            /* Most CPFxxxx  */
#define QCEEMSG    "QCEEMSG   *LIBL     "            /* CEE9901       */
#define QC2MSGF    "QC2MSGF   *LIBL     "            /* ILE Signals   */

/*********************************************************************/
/* All internal function prototypes go here                          */
/*********************************************************************/
static void buffer(char *Buffer, int Length);
static void record(char *Format, ...);
void handler(int iSignal);
static void cl_command(char *Command);
static int Pad(char *pszString, int iLength);
static int Trim(char *pszString, int iLength);

/*********************************************************************/
/* All file scoped variable declarations go here                     */
/*********************************************************************/
static int fRemoveEscapeMsg = 0;
static int fException       = 0;
static char acMsg[256];
static char acMsgKey[4];
static int i = 0;

/* Function Specification ********************************************/
/*                                                                   */
/* Function Name: Main                                               */
/*                                                                   */
/* Descriptive Name: Default Print Routing                           */
/*                                                                   */
/*   This program will use the IP address of the Telnet client       */
/*   (obtained via the QDCRDEVD API) to set up the default print     */
/*   device for the terminal session. The print device will be       */
/*   selected using a mapping table, with the client IP address      */
/*   being used as an index. The table will indicate the name of    */
/*   the remote output queue and the IP address, and we will use     */
/*   CL commands to set these values in the users default *OUTQ      */
/*   definition.                                                     */
/*                                                                   */
```

```
/*   The purpose here is to illustrate how an application can route   */
/*   printing over TCP/IP networks using the client IP address to     */
/*   determine what printer is "local" to that client, or to a        */
/*   "secure" printer for certain clients (such as payroll).          */
/*                                                                    */
/*   Printing can occur back to the client (dial-up) workstation, to  */
/*   a network printer (such as HP), or to any remote output queue.   */
/*   The only requirement here is that the *OUTQ definition must be   */
/*   set up ahead of time and must be dedicated to this particular    */
/*   user. If the *OUTQ is shared, results can be unpredictable       */
/*   as each client could modify the IP address.                      */
/*                                                                    */
/* Restrictions:                                                      */
/*                                                                    */
/*   User default output queues must not be shared. If the *OUTQ      */
/*   name does not match that of the user profile, the IP address     */
/*   will not be modified, to avoid modifying a shared *OUTQ.         */
/*                                                                    */
/* Notes: The "argv[]" parameters are "char *" by definition.         */
/*        Reference integers as "*(int(argv[1]))", for example.       */
/*        Consider the method for passing them back to the caller.    */
/*                                                                    */
/*                                                                    */
/* End Function Specification *****************************************/
void main(int argc, char *argv[])
{
  /********************************************************************/
  /* Local Variables                                                  */
  /********************************************************************/
  int  iParms;                          /* # required parameters     */

  Qsy_USRI0300_T usri0300;              /* For QSYRUSRI API call     */

  char QName[10 + 1];                   /* Default user *OUTQ name   */
  char QLibrary[10 + 1];                /* Default user *OUTQ lib    */
  char pszClientIP[15 + 1];             /* Client IP address         */

  char Command[256];                    /* Working buffer            */

  struct {
    char acJobName[10];                 /* Output from QUSRJOBI call */
    char acUserName[10];
    char acJobNumber[10];
  } JobName;

  Qwc_JOBI0100_t  jobi0100;             /* For QUSRJOBI API call     */

  /********************************************************************/
  /* Get the IP address of the client for logging                     */
  /* New definition for API call - additional members for IP address  */
  /********************************************************************/
  typedef _Packed struct Auxiliary_Devices {
    int  Auxiliary_Device_Addr;
    char Auxiliary_Device_Type[10];
    char Reserved[2];
  } Auxiliary_Devices_t;                /* From qdcrdevd.h           */

  typedef _Packed struct DEVD0600 {
    int  Bytes_Returned;
    int  Bytes_Available;
    char Date_Info_Retrieved[7];
    char Time_Info_Retrieved[6];
    char Device_Name[10];
    char Device_Category[10];
    char Online_At_IPL[10];
    char Text_Desc[50];
    char Reserved1[3];
    int  char_ID_Graphic_char_Set;
    int  char_ID_Code_Page;
    int  Max_Length_Request_Unit;
    int  Inactive_Timer;
    int  DBCS_Feature_RAM_Size;
    int  Activation_Timer;
    int  Switch_Setting;
    int  Device_Port;
    int  Max_Outstand_Frames;
    int  Idle_Timer;
    int  NRM_Poll_Timer;
```

```
  int  Frame_Retry;
  int  Offset_Auxiliary_Devices;
  int  Num_Auxiliary_Devices;
  int  Length_Auxiliary_Devices;
  char Device_Class[10];
  char Device_Type[10];
  char Device_Model[10];
  char Local_Location_Addr[10];
  char Attached_Non_Switch_Ctl_Name[10];
  char Keyboard_Language_Type[10];
  char Drop_Line_At_Signoff[10];
  char Allow_Blinking_Cursor[10];
  char Print_Device[10];
  char Remote_Location_Name[10];
  char Local_Location_Name[10];
  char Remote_Net_ID[10];
  char Ctl_Session_Device_Desc[10];
  char Assoc_Printer_Name[10];
  char Assoc_Printer_Remote_Net_ID[10];
  char Alternate_Printer_Name[10];
  char Alternate_Printer_Remote_Net_ID[10];
  char Output_Queue_Name[10];
  char Output_Queue_Library[10];
  char Printer[10];
  char Print_File_Name[10];
  char Print_File_Library[10];
  char Work_Station_Custom_Obj_Name[10];
  char Work_Station_Custom_Obj_Lib[10];
  char Application_Type[10];
  char DBCS_Feature_Matrix_Size[10];
  char DBCS_Feature_Language_ID[10];
  char DBCS_Feature_Last_Code_Point[10];
  char SNA_Pass_Through_Device[10];
  char SNA_Pass_Through_Group_Name[10];
  char Emulated_Device[10];
  char Emulated_Device_Model[10];
  char Emulating_ASCII_Device[10];
  char Physical_Attachment[10];
  char Line_Speed[10];
  char Word_Length[10];
  char Parity_Type[10];
  char Stop_Bits[10];
  char ASCII_Terminal_ID[20];
  char Assoc_APPC_Device[10];
  char Host_Signon_Logon_Command[256];
  char Pass_Through_ID[1];
  char Automatically_Configured[10];
  char Reserved2[3];
  int  Shared_Session_Num;
  char Dependent_Location_Name[10];
  char Network_Protocol[1];                                  /* NEW */
  char Network_Protocol_Address[18];                         /* NEW */
  char Internet_Address[15];                                 /* NEW */
  Auxiliary_Devices_t Aux_Devices[2];  /* commented out qdcrdevd.h  */
} DEVD0600_t;                          /* From qdcrdevd.h           */

DEVD0600_t devd0600;                   /* For QDCRDEVD API call     */

struct {                               /* IP address compare struct */
  char C1[4];                          /* Client IP address         */
  char C2[4];
  char C3[4];
  char C4[4];
  char M1[4];                          /* Mapping table IP address  */
  char M2[4];
  char M3[4];
  char M4[4];
} IP;

struct {                               /* Mapping table layout      */
  char Printer[10 + 1];                /* Printer device            */
  char PrinterIP[15 + 1];              /* Printer IP address        */
  char ClientIP[15 + 1];               /* Client IP address to match*/
  char User[10 + 1];                   /* User profile              */
  char Transform[5 + 1];               /* Transform SCS to ASCII?   */
  char MfgType[17 + 1];                /* Mfg type and model        */
  char DestType[10 + 1];               /* Destination type          */
} Map[500];                            /* Up to 500 entries         */
```

```
int iMap = -1;                          /* Index of matched entry    */

struct _rtvm0100 {
  int  Bytes_Return;
  int  Bytes_Available;
  int  Length_Message_Returned;
  int  Length_Message_Available;
  int  Length_Help_Returned;
  int  Length_Help_Available;
  char Message[240];                   /* commented out in qmhrtvm.h */
  char Message_Help[240];              /* commented out in qmhrtvm.h */
} rtvm0100;

char *pszMap="QUSRSYS/QRMTWTR(MAP)"; /* Map file name               */
_RIOFB_T *pFdBk;                     /* I/O feedback area           */
_RFILE *pRFile;                      /* Mapping file pointer        */
char pcRecord[240];                  /* Buffer for reads            */

/*****************************************************************/
/* Code                                                          */
/*****************************************************************/
record("\n");
record("qrmtwtr: >>>>> entry\n");
signal(SIGALL, &handler);      /* Trap all signals with our handler */
iParms = 1;                    /* Total of 0 parms on the interface */
record("qrmtwtr: argc: %d  iParms: %d\n", argc, iParms);
if (argc < iParms) {
  record("qrmtwtr: Invalid number of parameters!\n");
  record("qrmtwtr: <<<<< exit\n\n");
  exit(0);
} /* endif */

/*****************************************************************/
/* Initialize for QMHSNDPM API calls later                       */
/*****************************************************************/
memset(acMsg, 0x00, sizeof(acMsg));
memset(acMsgKey, 0x00, sizeof(acMsgKey));

/*****************************************************************/
/* Get user profile information                                  */
/*****************************************************************/
/* Record structure for USRI0300 format                          */
/* NOTE:  The following type definition only defines the fixed   */
/* portion of the format. Any varying length field will have to be */
/* defined by the user.                                          */
/*****************************************************************/
/* typedef struct Qsy_Qualified_Name {                           */
/*   char  Name[10];                                             */
/*   char  Library[10];                                          */
/* } Qsy_Qual_Name_T;                                            */
/*****************************************************************/
/* typedef struct Qsy_USRI0300 {                                 */
/*   int   Bytes_Returned;                                       */
/*   int   Bytes_Available;                                      */
/*   char  User_Profile[10];                                     */
/*   char  Previous_Signon[13];                                  */
/*   char  Reserved_1;                                           */
/*   int   Signon_Notval;                                        */
/*   char  User_Status[10];                                      */
/*   char  Pwdchg_Date[8];                                       */
/*   char  No_Password;                                          */
/*   char  Reserved_2;                                           */
/*   int   Pwdexp_Interval;                                      */
/*   char  Pwdexp_Date[8];                                       */
/*   int   Pwdexp_Days;                                          */
/*   char  Password_Expired;                                     */
/*   char  User_Class[10];                                       */
/*   Qsy_Special_Auth_T  Special_Auth;                           */
/*   char  Group_Profile[10];                                    */
/*   char  Owner[10];                                            */
/*   char  Group_Auth[10];                                       */
/*   char  Assistance_Level[10];                                 */
/*   char  Current_Library[10];                                  */
/*   Qsy_Qual_Name_T  Initial_Menu;                              */
/*   Qsy_Qual_Name_T  Initial_Program;                           */
/*   char  Limit_Capabilities[10];                               */
/*   char  Text_Description[50];                                 */
```

```
/*   char  Display_Signon[10];                                    */
/*   char  Limit_DeviceSsn[10];                                   */
/*   char  Keyboard_Buffering[10];                                */
/*   char  Reserved_3[2];                                         */
/*   int   Max_Storage;                                           */
/*   int   Storage_Used;                                          */
/*   char  Scheduling_Priority;                                   */
/*   Qsy_Qual_Name_T  Job_Description;                            */
/*   char  Accounting_Code[15];                                   */
/*   Qsy_Qual_Name_T  Message_Queue;                              */
/*   char  Msgq_Delivery[10];                                     */
/*   char  Reserved_4[2];                                         */
/*   int   Msgq_Severity;                                         */
/*   Qsy_Qual_Name_T  Output_Queue;                               */
/*   char  Print_Device[10];                                      */
/*   char  Special_Environment[10];                               */
/*   Qsy_Qual_Name_T  Attention_Program;                          */
/*   char  Language_Id[10];                                       */
/*   char  Country_Id[10];                                        */
/*   int   CCSID;                                                 */
/*   Qsy_User_Optn_T  User_Options;                               */
/*   Qsy_Qual_Name_T  Sort_Sequence;                              */
/*   char  Object_Audit[10];                                      */
/*   Qsy_Audit_Level_T  Audit_Level;                              */
/*   char  Group_Auth_Type[10];                                   */
/*   int   Supp_Group_Offset;                                     */
/*   int   Supp_Group_Number;                                     */
/*   uid_t UID;                                                   */
/*   gid_t GID;                                                   */
/*   int   HomeDir_Offset;                                        */
/*   int   HomeDir_Len;                                           */
/*   char  Supp_Group_Names[][10];                                */
/*   char  Qsy_Path_Info[];                                       */
/* } Qsy_USRI0300_T;                                              */
/****************************************************************/

/****************************************************************/
/* Now get the current user's output queue using QSYRUSRI API   */
/****************************************************************/
record("qrmtwtr: Retrieve current user profile *OUTQ information\n");
esErrCode.Bytes_Provided = 0;
QSYRUSRI(&usri0300, sizeof(Qsy_USRI0300_T), "USRI0300", "*CURRENT  ",
        &esErrCode);
memcpy(QName, usri0300.Output_Queue.Name, 10);
memcpy(QLibrary, usri0300.Output_Queue.Library, 10);
Trim(QName, 10);
Trim(QLibrary, 10);
record("qrmtwtr: User *OUTQ is: >%.10s/%.10s<\n", QLibrary, QName);

/****************************************************************/
/* Now get the current device using the QUSRJOBI API            */
/* typedef _Packed struct Qwc_JOBI0100 {                        */
/*   int  Bytes_Return;                                         */
/*   int  Bytes_Avail;                                          */
/*   char Job_Name[10];                                         */
/*   char User_Name[10];                                        */
/*   char Job_Number[6];                                        */
/*   char Int_Job_ID[16];                                       */
/*   char Job_Status[10];                                       */
/*   char Job_Type[1];                                          */
/*   char Job_Subtype[1];                                       */
/*   char Reserved[2];                                          */
/*   int  Run_Priority;                                         */
/*   int  Time_Slice;                                           */
/*   int  Default_Wait;                                         */
/*   char Purge[10];                                            */
/* } Qwc_JOBI0100_t;                                            */
/****************************************************************/
record("qrmtwtr: Retrieve virtual terminal device name\n");
memset(&JobName, ' ', sizeof(JobName));
JobName.acJobName[0] = '*';
esErrCode.Bytes_Provided = 0;
QUSRJOBI(&jobi0100, sizeof(jobi0100), "JOBI0100", &JobName,
         "               ", &esErrCode);
record("qrmtwtr: Virtual device: >%.10s<\n", jobi0100.Job_Name);

/****************************************************************/
/* Now get the IP address using the QDCRDEVD API                */
```

```
/*********************************************************************/
record("qrmtwtr: Retrieve Client IP address from: %.10s\n",
       jobi0100.Job_Name);
memset(&devd0600, 0x00, sizeof(devd0600));
esErrCode.Bytes_Provided = 0;
QDCRDEVD(&devd0600, sizeof(devd0600), "DEVD0600", jobi0100.Job_Name,
         &esErrCode);
memcpy(pszClientIP, devd0600.Internet_Address, 15);
Trim(pszClientIP, 15);
record("qrmtwtr: Client IP address is: >%.15s<\n", pszClientIP);

/*********************************************************************/
/* Scan client IP address into compare structure                     */
/*********************************************************************/
memset(&IP, 0x00, sizeof(IP));
sscanf(pszClientIP, "%[^.].%[^.].%[^.].%s", IP.C1,IP.C2,IP.C3,IP.C4);

/*********************************************************************/
/* Read mapping table into local variable - up to 500 entries.       */
/*********************************************************************/
if (NULL != (pRFile = _Ropen(pszMap, "rr")) ) {
  record("qrmtwtr: Map file >%s< opened\n", pszMap);
  record("qrmtwtr: Attempting match for: %s\n", pszClientIP);
  pFdBk = _Rreadf(pRFile, pcRecord, sizeof(pcRecord), __NO_LOCK);
  i = 0;
  while (pFdBk->num_bytes > 0) {
    /*****************************************************************/
    /* If '#' character in first column, this is a comment           */
    /*****************************************************************/
    if (pcRecord[0] == '#') {
      record("qrmtwtr: Line %d is a comment line\n", i+1);
      /***************************************************************/
      /* Step mapping table index variable                           */
      /***************************************************************/
      i++;
      /***************************************************************/
      /* Read next mapping table record                              */
      /***************************************************************/
      pFdBk = _Rreadn(pRFile, pcRecord, sizeof(pcRecord), __NO_LOCK);
      continue;
    } /* endif */
    record("qrmtwtr: Reading line %d\n", i+1);
    /*****************************************************************/
    /* Read and parse mapping table                                  */
    /*****************************************************************/
    sscanf(pcRecord, "| %s | %s | %s | %s | %s | %s | %s | ",
           Map[i].Printer,
           Map[i].PrinterIP,
           Map[i].ClientIP,
           Map[i].User,
           Map[i].Transform,
           Map[i].MfgType,
           Map[i].DestType);
    /*****************************************************************/
    /* Read IP address from table into compare structure             */
    /*****************************************************************/
    sscanf(Map[i].ClientIP, "%[^.].%[^.].%[^.].%s",
           IP.M1, IP.M2, IP.M3, IP.M4);
    buffer((char *)&IP, sizeof(IP));
    /*****************************************************************/
    /* Compare IP address. Octets must match or be "*" wildcarded.  */
    /*****************************************************************/
    if ((!strcmp(IP.C1, IP.M1) || !strcmp("*", IP.M1)) &&
        (!strcmp(IP.C2, IP.M2) || !strcmp("*", IP.M2)) &&
        (!strcmp(IP.C3, IP.M3) || !strcmp("*", IP.M3)) &&
        (!strcmp(IP.C4, IP.M4) || !strcmp("*", IP.M4))) {
      record("qrmtwtr: Match found!\n");
      /***************************************************************/
      /* Set index of matched entry                                  */
      /***************************************************************/
      iMap = i;
      /***************************************************************/
      /* If field has '#' character, then use blanks for this field */
      /***************************************************************/
      if ('#'==Map[i].Printer[0])  {memset(Map[i].Printer,  ' ',10);}
      if ('#'==Map[i].PrinterIP[0]){memset(Map[i].PrinterIP,' ',10);}
      if ('#'==Map[i].ClientIP[0]) {memset(Map[i].ClientIP, ' ',10);}
      if ('#'==Map[i].User[0])     {memset(Map[i].User,     ' ',10);}
```

```
      if ('#'==Map[i].Transform[0]){memset(Map[i].Transform,' ',5);}
      if ('#'==Map[i].MfgType[0])  {memset(Map[i].MfgType,  ' ',17);}
      if ('#'==Map[i].DestType[0]) {memset(Map[i].DestType, ' ',10);}
      /*************************************************************/
      /* Show me the values selected (debug)                       */
      /*************************************************************/
      record("qrmtwtr: Map[%d].Printer:   >%.10s<\n",
             i, Map[i].Printer);
      record("qrmtwtr: Map[%d].PrinterIP: >%.15s<\n",
             i, Map[i].PrinterIP);
      record("qrmtwtr: Map[%d].ClientIP:  >%.15s<\n",
             i, Map[i].ClientIP);
      record("qrmtwtr: Map[%d].User:      >%.10s<\n",
             i, Map[i].User);
      record("qrmtwtr: Map[%d].Transform: >%.5s<\n",
             i, Map[i].Transform);
      record("qrmtwtr: Map[%d].MfgType:   >%.17s<\n",
             i, Map[i].MfgType);
      record("qrmtwtr: Map[%d].DestType:  >%.10s<\n",
             i, Map[i].DestType);
      /*************************************************************/
      /* Since a match was found, break out of loop                */
      /*************************************************************/
      break;
    } /* endif */
    /***************************************************************/
    /* Step mapping table index variable                           */
    /***************************************************************/
    i++;
    /***************************************************************/
    /* Read next mapping table record                              */
    /***************************************************************/
    pFdBk = _Rreadn(pRFile, pcRecord, sizeof(pcRecord), __NO_LOCK);
  } /* endwhile */

  /*****************************************************************/
  /* Close mapping table file                                      */
  /*****************************************************************/
  if (NULL != pRFile) {
    record("qrmtwtr: File >%s< closed\n", pszMap);
    _Rclose(pRFile);
  } /* endif */

} /* endif */

/*******************************************************************/
/* Now build and run a CL command to modify the output queue for   */
/* this session, but only if the output queue name matches the user */
/* profile being used (to avoid changing system output queues).    */
/*******************************************************************/
record("qrmtwtr: User profile: >%.10s<  Default OUTQ name: >%.10s<\n",
       usri0300.User_Profile, usri0300.Output_Queue.Name);

if (memcmp(usri0300.User_Profile, usri0300.Output_Queue.Name, 10) ) {
  record("qrmtwtr: Names do not match.  No IP change.\n");
  /*****************************************************************/
  /* Build failure message for job log                             */
  /*****************************************************************/
  sprintf(acMsg,
          "Default output queue name does not match user profile. ",
          "Output queue %s/%s not modified.",
          QLibrary, QName);
} else if (-1 == iMap) {
  record("qrmtwtr: No entry in mapping table.  No IP change.\n");
  /*****************************************************************/
  /* Build failure message for job log                             */
  /*****************************************************************/
  sprintf(acMsg,
          "IP address '%s' not listed in %s.  ",
          "Output queue %s/%s not modified.",
          pszClientIP, pszMap, QLibrary, QName);
} else if (0x00 == pszClientIP[0]) {
  record("qrmtwtr: IP address not set in device (SNA terminal?)\n");
  /*****************************************************************/
  /* Build failure message for job log                             */
  /*****************************************************************/
  sprintf(acMsg,
          "Device %.10s does not have an IP address.  ",
```

```
          "Output queue %s/%s not modified.",
          jobi0100.Job_Name, QLibrary, QName);
} else {
  record("qrmtwtr: Names match.  Changing IP...\n");
  memset(Command, 0x00, sizeof(Command));

  /*****************************************************************/
  /* First, end any writer that is running. Ignore (remove) any    */
  /* errors related to ending the writer (such as CPF3313).        */
  /*****************************************************************/
  fRemoveEscapeMsg = fTrue;
  sprintf(Command, "QSYS/ENDWTR WTR(%s) OPTION(*IMMED)", QName);
  cl_command(Command);
  fRemoveEscapeMsg = fFalse;
  /*****************************************************************/
  /* Give the ENDWTR time to end so the CHGOUTQ does not fail      */
  /*****************************************************************/
  record("qrmtwtr: Wait 2 seconds ...\n");
  sleep(2);
  /*****************************************************************/
  /* Change the IP address. Set flag to detect any errors trapped  */
  /* as a result of the CHGOUTQ call.                              */
  /*****************************************************************/
  if (!memcmp(Map[iMap].PrinterIP, "*CLIENT", 7)) {
    /***************************************************************/
    /* *CLIENT means to substitute the client IP address...        */
    /***************************************************************/
    sprintf(Command,
            "QSYS/CHGOUTQ OUTQ(%s/%s) RMTPRTQ('%s') CNNTYPE(*IP) "
            "AUTOSTRWTR(1) RMTSYS(*INTNETADR) INTNETADR('%s') "
            "TRANSFORM(%s) MFRTYPMDL(%s) DESTTYPE(%s)",
            QLibrary, QName,
            Map[iMap].Printer,
            Map[iMap].ClientIP,
            Map[iMap].Transform,
            Map[iMap].MfgType,
            Map[iMap].DestType);
  } else {
    /***************************************************************/
    /* ...otherwise use the given print IP address.                */
    /***************************************************************/
    sprintf(Command,
            "QSYS/CHGOUTQ OUTQ(%s/%s) RMTPRTQ('%s') CNNTYPE(*IP) "
            "AUTOSTRWTR(1) RMTSYS(*INTNETADR) INTNETADR('%s') "
            "TRANSFORM(%s) MFRTYPMDL(%s) DESTTYPE(%s)",
            QLibrary, QName,
            Map[iMap].Printer,
            Map[iMap].PrinterIP,
            Map[iMap].Transform,
            Map[iMap].MfgType,
            Map[iMap].DestType);
  } /* endelse */
  fException = fFalse;
  cl_command(Command);
  /*****************************************************************/
  /* If no error on the CHGOUTQ, post success message.             */
  /*****************************************************************/
  if (!fException) {
    /***************************************************************/
    /* Build success message for job log                           */
    /***************************************************************/
    sprintf(acMsg, "Output queue %s/%s now using printer '%s' at %s.",
            QLibrary, QName, Map[iMap].Printer, Map[iMap].PrinterIP);
  } else {
    /***************************************************************/
    /* Build failure message for job log                           */
    /***************************************************************/
    sprintf(acMsg, "Error changing output queue %s/%s.",
            QLibrary, QName);
  } /* endelse */
} /* endelse */

/*****************************************************************/
/* Post a success or failure message so we know what happened    */
/*****************************************************************/
record("qrmtwtr: Posting message to job log\n");
QMHSNDPM("CPF9897", QCPFMSG, acMsg, strlen(acMsg), MSG_INFO,
         MSQ_Q_CUR_PROG, 0, acMsgKey, &esErrCode);
```

```
  record("qrmtwtr: <<<<< exit\n\n");
  exit(0);
}                                                        /* End main */

/*********************************************************************/
/*@function buffer()                                                 */
/*                                                                   */
/* Parameters:                                                       */
/*                                                                   */
/* char *buffer - points at buffer to dump                           */
/* int  length  - length of buffer to dump                           */
/*                                                                   */
/* Description:                                                      */
/*                                                                   */
/* Dumps out a buffer in both hex and readable form.                 */
/*                                                                   */
/*         1934D8E3 D4E3E2D7 C3F0F0F2 40404040  |..QTMTSPC002    |    */
/*         40404040 40404040 40404040 40404040  |                |    */
/*         00000000 00000000 00000000 00000000  |................|    */
/*********************************************************************/
void buffer(char *Buffer, int Length)
{
#ifdef DEBUG
  int i=0, j=0, k=0, numrows=0;
  int iLineLen;
  unsigned char c;
  char acLine[20];
  char buff[512];
  char *ptr = buff;

  if (Buffer == (char *)NULL) {
    return;
  } /* endif */

  numrows = (Length + 15) / 16;
  for (i = 0; i < numrows; i++) {
    /*****************************************************************/
    /* Offset into line by 4 blank characters                        */
    /*****************************************************************/
    *ptr = ' ';
    ptr++;
    *ptr = ' ';
    ptr++;
    *ptr = ' ';
    ptr++;
    *ptr = ' ';
    ptr++;
    /*****************************************************************/
    /* Print 16 bytes of the buffer as the hexadecimal dump section  */
    /*****************************************************************/
    for (j = 0; j < 16; j++) {
      if (j + (i * 16) >= Length) {
        *ptr = ' ';
        ptr++;
        *ptr = ' ';
        ptr++;
      } else {
        sprintf(ptr, "%02X", Buffer[j + (i * 16)]);
        ptr+=2;
      } /* endif */
      /*****************************************************************/
      /* Odd number of bytes?                                          */
      /*****************************************************************/
      if (j % 4 == 3) {
        *ptr = ' ';
        ptr++;
      } /* endif */
    } /* endfor */
    /*****************************************************************/
    /* Print the same 16 bytes as the "readable" section             */
    /*****************************************************************/
    *ptr = ' ';
    ptr++;
    *ptr = '|';                                  /* Left frame bar  */
    ptr++;
    for (j=0; j < 16; j++) {
      if (j + (i * 16) >= Length) {
```

```
        *ptr = ' ';
        ptr++;
      } else {
        k++;                                        /* Count real data */
        c = Buffer[j + (i * 16)];
        if (isprint(c)) {
          *ptr = c;
          ptr++;
        } else if (c == 0x40) {
          *ptr = ' ';
          ptr++;
        } else {
          *ptr = '.';
          ptr++;
        } /* endif */
      }/* endif */
    } /* endfor */
    *ptr = '|';                                     /* Right frame bar */
    ptr++;
    memset(acLine, 0x00, sizeof(acLine));
    sprintf(acLine, "  Byte %d\n", (i * 16) + k);
    iLineLen = strlen(acLine);
    memcpy(ptr, acLine, iLineLen);
    ptr += iLineLen;
    *ptr = 0x00;
    ptr = &buff[0];
    printf("%s", buff);
  } /* endfor */
#endif
  return;
}

/*********************************************************************/
/*@function record()                                                 */
/*                                                                   */
/* Parameters:                                                       */
/*                                                                   */
/* variable arguments                                                */
/*                                                                   */
/* Log test result entry to standard out (normally console). This    */
/* will only occur if DEBUG is active. If this occurs in a batch     */
/* job, a spooled file is usually created holding output.            */
/*********************************************************************/
void record(char *Format, ...)
{
#ifdef DEBUG
  va_list arg_ptr;
  int iLen;
  char record[512];

  va_start(arg_ptr, Format);
  iLen = vsprintf(record, Format, arg_ptr);
  va_end(arg_ptr);
  record[iLen] = 0x00;
  printf("%s", record);
#endif
  return;
}

/*********************************************************************/
/*@function handler()                                                */
/*                                                                   */
/* Parameters:                                                       */
/*                                                                   */
/* int iSignal - value of the signal that caused the handler to be   */
/*               invoked. SIGABRT, SIGTERM, etc.                     */
/*********************************************************************/
void handler(int iSignal)
{
  _INTRPT_Hndlr_Parms_T   Signal;
  _INTRPT_Hndlr_Parms_T  *pSignal = &Signal;

  char *pszMsgFile = NULL;

  struct {
    int  Bytes_Return;
    int  Bytes_Available;
    int  Length_Message_Returned;
```

```
  int  Length_Message_Available;
  int  Length_Help_Returned;
  int  Length_Help_Available;
  char Message[256];
  char Message_Help[256];
} rtvm0100;

char *Signals[] = {
"SIGABRT",     /* 1   Abnormal termination                          */
"SIGFPE",      /* 2   Erroneous arithmetic operation                */
"SIGILL",      /* 3   Invalid hardware instruction                  */
"SIGINT",      /* 4   Interactive attention signal                  */
"SIGSEGV",     /* 5   Invalid memory reference                      */
"SIGTERM",     /* 6   Termination signal                            */
"SIGUSR1",     /* 7   Application defined signal 1                  */
"SIGUSR2",     /* 8   Application defined signal 2                  */
"SIGIO",       /* 9   I/O possible, or completed                    */
"SIGALL",      /* 10  All signals                                   */
"SIGOTHER",    /* 11  ILE C/400 signal                              */
"SIGKILL",     /* 12  Termination signal (cannot be caught,ignored) */
"SIGPIPE",     /* 13  Write on a pipe with no readers               */
"SIGALRM",     /* 14  Timeout signal                                */
"SIGHUP",      /* 15  Hangup detected on controlling terminal       */
"SIGQUIT",     /* 16  Interactive termination signal                */
"SIGSTOP",     /* 17  Stop signal (cannot be caught or ignored)     */
"SIGTSTP",     /* 18  Interactive stop signal                       */
"SIGCONT",     /* 19  Continue if stopped                           */
"SIGCHLD",     /* 20  Child process terminated or stopped           */
"SIGTTIN",     /* 21  Background read from controlling terminal     */
"SIGTTOU",     /* 22  Background write to controlling terminal      */
"SIGURG",      /* 23  High bandwidth data is available at a socket  */
"SIGPOLL",     /* 24  Pollable event                                */
"SIG25",       /* 25  Not defined                                   */
"SIG26",       /* 26  Not defined                                   */
"SIG27",       /* 27  Not defined                                   */
"SIG28",       /* 28  Not defined                                   */
"SIG29",       /* 29  Not defined                                   */
"SIG30",       /* 30  Not defined                                   */
"SIG31",       /* 31  Not defined                                   */
"SIGBUS",      /* 32  Bus error (specification exception)           */
"SIGDANGER",   /* 33  system crash imminent                         */
"SIGPRE",      /* 34  programming exception                         */
"SIGSYS",      /* 35  Bad system call                               */
"SIGTRAP",     /* 36  Trace/Breakpoint trap                         */
"SIGPROF",     /* 37  Profiling timer expired                       */
"SIGVTALRM",   /* 38  Virtual timer expired                         */
"SIGXCPU",     /* 39  CPU time limit exceeded                       */
"SIGXFSZ"      /* 40  File size limit exceeded                      */
};

record("handler: Caught signal %s\n", Signals[iSignal-1]);

/*****************************************************************/
/* Set file scoped flag so caller knows exception occurred       */
/*****************************************************************/
fException = fTrue;

/*****************************************************************/
/* Try and pull out the message text                             */
/*****************************************************************/
_GetExcData(&Signal);
if (!memcmp(Signal.Msg_Id, "TCP", 3) ) {
  /***************************************************************/
  /* TCP Apps (and Stack with recursive call) message file       */
  /***************************************************************/
  pszMsgFile = QTCPMSGF;
} else if (!memcmp(Signal.Msg_Id, "C2M16", 5) ) {
  /***************************************************************/
  /* ILE-C message file (primarily for signals, if use raise/abort) */
  /***************************************************************/
  pszMsgFile = QC2MSGF;
} else if (!memcmp(Signal.Msg_Id, "CEE99", 5) ) {
  /***************************************************************/
  /* ILE-C message file                                          */
  /***************************************************************/
  pszMsgFile = QCEEMSG;
} else {
  /***************************************************************/
```

```
    /* Most CPFxxxx messages                                              */
    /*********************************************************************/
    pszMsgFile = QCPFMSG;
  } /* endif */

  esErrCode.Bytes_Provided  = sizeof(ERRSTRUCT);
  esErrCode.Bytes_Available = 0;
  memset(&esErrCode.Exception_Id, ' ', sizeof(&esErrCode.Exception_Id));
  QMHRTVM(&rtvm0100,                /* Message information            */
          sizeof(rtvm0100),         /* Length of message information  */
          "RTVM0100",               /* Format name                    */
          Signal.Msg_Id,            /* Message identifier             */
          pszMsgFile,               /* Qualified message file name    */
          Signal.Ex_Data,           /* Message data                   */
          sizeof(Signal.Ex_Data),   /* Length of message data         */
          "*YES      ",             /* Replace substitution values    */
          "*NO       ",             /* Return format control          */
          &esErrCode,               /* Error Code                     */
          "*MSGID    ",             /* Retrieve option                */
          0,                        /* Convert to CCSID               */
          0);                       /* Message data CCSID             */

  if (!rtvm0100.Length_Message_Returned) {
    record("handler: Escape message not found\n");
  } else {
    record("handler: Escape message:\n");
    buffer(rtvm0100.Message, rtvm0100.Length_Message_Returned);
  } /* endelse */

  /*********************************************************************/
  /* Delete msg from job log if caller so desires (flag indicator)     */
  /*********************************************************************/
  if (fRemoveEscapeMsg) {
    record("handler: Remove escape message from job log\n");
    esErrCode.Bytes_Provided  = sizeof(ERRSTRUCT);
    esErrCode.Bytes_Available = 0;
    memset(&esErrCode.Exception_Id,' ',sizeof(&esErrCode.Exception_Id));
    QMHCHGEM(&Signal.Target, 0, &Signal.Msg_Ref_Key, "*REMOVE   ",
             "", 0, &esErrCode);
  } else {
    record("handler: Escape message not removed from job log\n");
  } /* endif */

  record("handler: Reset signals -> handler\n");
  signal(SIGALL, &handler);
  return;
}

/* Function Specification ********************************************/
/*                                                                   */
/* Function Name: cl_command                                         */
/*                                                                   */
/* Descriptive Name: run any CL command as if on a command line.     */
/*                                                                   */
/*  char * Command            - null terminated string               */
/*                                                                   */
/* End Function Specification ****************************************/
void cl_command(char *Command)                       /* Entry point */
{
  /*********************************************************************/
  /* Local Variables                                                   */
  /*********************************************************************/
  Qca_PCMD_CPOP0100_t cpop0100;
  char                cpop0100_out[512];
  int                 cpop0100_len;

  /*********************************************************************/
  /* typedef _Packed struct Qca_PCMD_CPOP0100 {                        */
  /*       int  Command_Process_Type;                                  */
  /*       char DBCS_Data_Handling;                                    */
  /*       char Prompter_Action;                                       */
  /*       char Command_String_Syntax;                                 */
  /*       char Message_Key[4];                                        */
  /*       char Reserved[9];                                           */
  /*    }  Qca_PCMD_CPOP0100_t;                                        */
  /*********************************************************************/
  record("cl_command: Command:\n");
  buffer(Command, strlen(Command));
```

```
  memset(cpop0100_out, 0x00, sizeof(cpop0100_out));
  memset(&cpop0100, 0x00, sizeof(Qca_PCMD_CPOP0100_t));
  cpop0100.Command_Process_Type  = 0;
  cpop0100.DBCS_Data_Handling    = '0';
  cpop0100.Prompter_Action       = '0';
  cpop0100.Command_String_Syntax = '0';
  esErrCode.Bytes_Provided = 0;
  QCAPCMD(Command, strlen(Command), &cpop0100,
          sizeof(Qca_PCMD_CPOP0100_t), "CPOP0100", cpop0100_out,
          sizeof(cpop0100_out) - 1, &cpop0100_len, &esErrCode);
  return;
}

/* Function Specification *********************************************/
/*                                                                    */
/* Function Name: Pad                                                 */
/*                                                                    */
/* char *pszString - null terminated string, may or may not have      */
/*                   blanks                                           */
/*                                                                    */
/* int iLength   - maximum length to pad the string (includes NULL    */
/*                   terminator). A NULL terminator WILL be added!    */
/*                                                                    */
/*                   iLength should be where you want the NULL term.  */
/*                                                                    */
/*                   Thus, iLength=10 means array position 10, not 9. */
/*                                                                    */
/* Descriptive Name: pads a string with blanks                        */
/*                                                                    */
/* End Function Specification *****************************************/
int Pad(char *pszString, int iLength)
{
  int iLen = strlen(pszString);

  if (iLength <= iLen) {
    return(iLen);
  } /* endif */
  /*******************************************************************/
  /* Start at the end of the string and add blanks.                  */
  /*******************************************************************/
  while (iLen < iLength) {
    pszString[iLen] = ' ';
    iLen++;
  } /* endwhile */
  /*******************************************************************/
  /* Add the null terminator to make it a 'C' string.                */
  /*******************************************************************/
  pszString[iLen] = 0x00;
  iLen = strlen(pszString);

  return(iLen);
}

/* Function Specification *********************************************/
/*                                                                    */
/* Function Name: Trim                                                */
/*                                                                    */
/* char *pszString - null terminated string, may have trailing        */
/*                   blanks                                           */
/*                                                                    */
/* int iLength   - start position of the string to be trimmed. This   */
/*                   need not be the length of the string. You should */
/*                   specify the position of any NULL terminator if you */
/*                   are trimming a 'C' string.                       */
/*                                                                    */
/*                   iLength should be where you want the NULL term.  */
/*                                                                    */
/*                   Thus, iLength=10 means array position 10, not 9. */
/*                                                                    */
/* Descriptive Name: pads a string with blanks                        */
/*                                                                    */
/* End Function Specification *****************************************/
int Trim(char *pszString, int iLength )
{
  int iLen = iLength - 1;

  /*******************************************************************/
  /* While we are not at the 1st char AND (a char is white space OR  */
```

```
  /* 0x00) keep backing up from end of string                          */
  /*********************************************************************/
  pszString[iLength] = 0x00;
  while ( (iLen >= 0) &&
          ((isspace(pszString[iLen]) ) || (0x00 == pszString[iLen])) ) {
    iLen--;
  } /* endwhile */
  /*********************************************************************/
  /* if iLen unchanged, it means no trimming was done (no white space */
  /* found).                                                           */
  /*********************************************************************/
  if (iLen == (iLength-1)) {
    return(iLen);
  } /* endif */
  /*********************************************************************/
  /* if iLen >= 0, we found 1 or more chars, so mark end with a        */
  /* NULL terminator.                                                  */
  /*********************************************************************/
  else if (iLen >= 0) {
    pszString[iLen + 1] = 0x00;
  } /* endif */
  /*********************************************************************/
  /* else iLen < 0, which means we didn't find anything but white      */
  /* space                                                             */
  /*********************************************************************/
  else {
    pszString[0] = 0x00;
  } /* endelse */

  iLen = strlen(pszString);
  return(iLen);
}

#undef _QRMTWTR_C
/*********************************************************************/
/*                         END OF QRMTWTR.C                          */
/*********************************************************************/
```

Appendix B. Special notices

This publication is intended to help plan and implement TCP/IP on the AS/400 system. The information in this publication is not intended as the specification of any programming interfaces that are provided by Operating System/400, operating system options and features, or AS/400 TCP/IP Connectivity Utilities/400. See the PUBLICATIONS section of the IBM Programming Announcement for Operating System/400 for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY 10594 USA.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The information about non-IBM ("vendor") products in this manual has been supplied by the vendor and IBM assumes no responsibility for its accuracy or completeness. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

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

Any performance data contained in this document was determined in a controlled environment, and therefore, the results that may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data for their specific environment.

The following document contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples contain the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

Reference to PTF numbers that have not been released through the normal distribution process does not imply general availability. The purpose of including these reference numbers is to alert IBM customers to specific information relative to the implementation of the PTF when it becomes available to each customer according to the normal IBM PTF distribution process.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

ADSTAR	AFP
AIX	APPN
AS/400	AT
C/400	CT
DataPropagator	DB2
Distributed Relational Database Architecture	DRDA
eNetwork	IBM Global Network
IBM	ImagePlus
LPDA	Netfinity
Network Station	Nways
OfficeVision	Operating System/400
OS/2	OS/390
OS/400	RS/6000
SP	S/390
System/38	System/390
SystemView	VisualGen
VisualInfo	XT
400	

The following terms are trademarks of other companies:

Tivoli, Manage. Anything. Anywhere.,The Power To Manage., Anything. Anywhere.,TME, NetView, Cross-Site, Tivoli Ready, Tivoli Certified, Planet Tivoli, and Tivoli Enterprise are trademarks or registered trademarks of Tivoli Systems Inc., an IBM company, in the United States, other countries, or both. In Denmark, Tivoli is a trademark licensed from Kjøbenhavns Sommer - Tivoli A/S.

C-bus is a trademark of Corollary, Inc. in the United States and/or other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and/or other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States and/or other countries.

PC Direct is a trademark of Ziff Communications Company in the United States and/or other countries and is used by IBM Corporation under license.

ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States and/or other countries.

UNIX is a registered trademark in the United States and other countries licensed exclusively through The Open Group.

SET and the SET logo are trademarks owned by SET Secure Electronic Transaction LLC.

Other company, product, and service names may be trademarks or service marks of others.

Appendix C. Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

C.1 IBM Redbook publications

For information on ordering these ITSO publications see “How to get IBM Redbooks” on page 699.

- *IBM AS/400 Printing V,* SG24-2160
- *AS/400 Internet Security: IBM Firewall for AS/400,* SG24-2162
- *DB2/400 Advance database Functions,* SG24-4249
- *IBM 2210 Nways Multiprotocol Router Description and Configuration Scenarios - Volume 1,* SG24-4446
- *The Domino Defense: Security in Lotus Notes and the Internet*, SG24-4848
- *AS/400 Internet Security: Protecting Your AS/400 from HARM in the Internet*, SG24-4929
- *Understanding LDAP*, SG24-4986
- *LDAP Information Cookbook*, SG24-5110
- *AS/400 TCP/IP Autoconfiguration: DNS and DHCP Support,* SG24-5147
- *AS/400 IBM Network Station: Techniques for Deploying Network Station in a Wide Area Network (WAN),* SG24-5187
- *IBM Firewall for AS/400: VPN and NAT Support*, SG24-5376
- *AS/400 Internet Security: Implementing AS/400 Virtual Private Networks*, SG24-5404

C.2 IBM Redbooks collections

Redbooks are also available on the following CD-ROMs. Click the CD-ROMs button at http://www.redbooks.ibm.com/ for information about all the CD-ROMs offered, updates and formats.

CD-ROM Title	Collection Kit Number
System/390 Redbooks Collection	SK2T-2177
Networking and Systems Management Redbooks Collection	SK2T-6022
Transaction Processing and Data Management Redbooks Collection	SK2T-8038
Lotus Redbooks Collection	SK2T-8039
Tivoli Redbooks Collection	SK2T-8044
AS/400 Redbooks Collection	SK2T-2849
Netfinity Hardware and Software Redbooks Collection	SK2T-8046
RS/6000 Redbooks Collection (BkMgr Format)	SK2T-8040
RS/6000 Redbooks Collection (PDF Format)	SK2T-8043
Application Development Redbooks Collection	SK2T-8037
IBM Enterprise Storage and Systems Management Solutions	SK3T-3694

C.3 Other publications

These publications are also relevant as further information sources:

- *HTTP Server for AS/400 Webmaster's Guide*, GC41-5434
- *TCP/IP Tutorial and Technical Overview,* GG24-3376
- *IBM AS/400 TCP/IP Configuration and Operation,* GG24-3442
- *S/400 Client Access for Windows 95/NT - Setup V4R2*, SC41-3512
- *Distributed Data Management*, SC41-5307
- *TCP/IP Configuration and Reference,* SC41-5420
- *Communications Management Guide*, SC41-5406
- *Sockets Programming,* SC41-5422
- *OS/400 CL Reference*, SC41-5722
- *System API Reference,* SC41-5801
- Albitz, Paul and Liu, Cricket. *DNS and BIND*. O'Reilly and Associates, September 1998 (ISBN:15-659251-22).
- Stevens, Richard W. *TCP/IP Illustrated, Volume 1: The Protocols*. Addison-Wesley Publishing Company, January 1994 (ISBN: 02-016334-69).
- Stevens, Richard W. *TCP/IP Illustrated, Volume 2: The Implementation*. Addison-Wesley Publishing Company, January 1995 (ISBN: 02-016335-4X).
- Stevens, Richard W. *TCP/IP Illustrated, Volume 3: TCP for Transactions, HTTP, NNTP, and the UNIX Domain Protocols*. Addison-Wesley Publishing Company, January 1996 (ISBN: 02-016349-53).
- Stevens, Richard W. *Unix Network Programming, Volume 1: Networking APIs, Sockets and XTI*. Prentice Hall, October 1997 (ISBN: 01-349001-2X)

The following publications are only available online at:
`http://as400bks.rochester.ibm.com/pubs/html/as400/onlinelib.htm`

- *AS/400 Performance Capabilities Reference, V4R4,* SC41-0607
- *OS/400 CL Reference, Volume 1*, SC41-5723
- *OS/400 CL Reference, Volume 2*, SC41-5724
- *OS/400 CL Reference, Volume 3*, SC41-5725
- *OS/400 CL Reference, Volume 4*, SC41-5726

The following references are available on the Web at: `http://www.opengroup.org`
At the site, click the **current list of documents available on the Web** link.

- *DRDA Volume 1: Distributed Relational Database Architecture (DRDA)* (C911)
- *DRDA Volume 2: Formatted Data Object Content Architecture* (C912)
- *DRDA Volume 3: Distributed Data Management (DDM) Architecture* (C913)

The following Requests for Comments (RFCs) can be accessed on the Web at:
`http://www.rfc-editor.org`

- *Address Allocation for Private Internets* (RFC 1918)
- *Assigned Numbers* (RFC 1700)

- *Common DNS Operations and Configuration Errors* (RFC 1912)
- *Domain Names Concepts and Facilities* (RFC 1034)
- *Domain Names Implementations and Specifications* (RFC 1035)
- *Nonstandard for Transmission of IP Datagrams Over Serial Line*s (RFC 1055)
- *Security Consideration for IP Fragment Filtering* (RFC 1858)

C.4 Referenced Web sites

These Web sites are also relevant as further information sources:

- A wealth of AS/400 publications and related information can be found at the AS/400 Information Center Web site at:
  `http://publib.boulder.ibm.com/pubs/html/as400/ic2924/info/index.htm`
- Free guides, corporate information and newsworthy items regarding VeriSign are available at the VeriSign home page at: `http://www.verisign.com`
- The IBM WebSphere home page offers information on the WebSphere family of products, including some complimentary downloads. Visit their site at:
  `http://www.software.ibm.com/webservers`
- IBM offers several network security news and tips, including 10 Tips for Protecting Online Companies, at the IBM security Web page at:
  `http://www.ibm.com/security`
- For an updated list of available IBM software releases, visit the Web site at:
  `http://www.internet.ibm.com/commercepoint/registry`
- The RSA Security home page can be visited at: `http://www.rsa.com`
- An exhaustive list of available RFC documents can be accessed at:
  `http://www.rfc-editor.org`
- Visit the Microsoft home page at: `http://www.microsoft.com`
- Valuable Internet Engineering Task Force (IETF) documentation can be accessed at: `http://www.ietf.org/ids.by.wg/tn3270e.html`
- IBM TCP/IP reference information can be accessed at:
  `http://www.as400.ibm.com/tstudio/tech_ref/tcp/indexfr.htm`
- Information regarding the Telnet SSL Proxy Server can be viewed on the Web at: `http://www.as400.ibm.com/tstudio/tech_ref/tcp/sslproxy/index.htm`
- Access new and features regarding the IBM SecureWay Host On-Demand interface at: `http://www.software.ibm.com/enetwork/hostondemand`
- For a wide array of AS/400 information, visit the IBM AS/400 Information Center at: `http://www.as400.ibm.com/infocenter`
- Downloadable updates and support for SAP and information on AS/400 tools for R3 installations are available at:
  `http://service.software.ibm.com/dl/sap/saptools-d`
- Access the DNS Resource Directory, including newsgroups, at:
  `http://www.dns.net/dnsrd`
- CyberKit, a PC tool for implementing nslookup functions, can be purchased on the Web at: `http://www.tucows.com`

- Information regarding AS/400 Virtual Private Networking is available at: http://www.as400.ibm.com/vpn
- For information about ICSA certification, see the Web site at: http://www.icsa.net
- For an in-depth discussion about LDAP, directories and concepts, refer to the Web site at: http://www.umich.edu/~dirsvcs/ldap
- Netscape offers C and Java SDKs on the Web: http://developer.netscape.com/software/sdks/index.html
- Search for IBM's C and Java SDKs at: http://www.ibm.com/Help
- For an overview and links regarding Lightweight Directory Access Protocol (LDAP), visit the AS/400 Directory Services Web site at: http://www.as400.ibm.com/ldap
- For the latest information on obtaining Secure Sockets Layer (SSL) software and configuring AS/400 Directory Services to use SSL from your workstation, refer to the Client Access Web page at: http://www.as400.ibm.com/clientaccess
- A wealth of Open Group related publications can be accessed and ordered from the Open Group Web site at: http://www.opengroup.org

How to get IBM Redbooks

This section explains how both customers and IBM employees can find out about IBM Redbooks, redpieces, and CD-ROMs. A form for ordering books and CD-ROMs by fax or e-mail is also provided.

- **Redbooks Web Site** http://www.redbooks.ibm.com/

  Search for, view, download, or order hardcopy/CD-ROM Redbooks from the Redbooks Web site. Also read redpieces and download additional materials (code samples or diskette/CD-ROM images) from this Redbooks site.

  Redpieces are Redbooks in progress; not all Redbooks become redpieces and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

- **E-mail Orders**

  Send orders by e-mail including information from the IBM Redbooks fax order form to:

	e-mail address
In United States	usib6fpl@ibmmail.com
Outside North America	Contact information is in the “How to Order” section at this site: http://www.elink.ibmlink.ibm.com/pbl/pbl

- **Telephone Orders**

United States (toll free)	1-800-879-2755
Canada (toll free)	1-800-IBM-4YOU
Outside North America	Country coordinator phone number is in the “How to Order” section at this site: http://www.elink.ibmlink.ibm.com/pbl/pbl

- **Fax Orders**

United States (toll free)	1-800-445-9269
Canada	1-403-267-4455
Outside North America	Fax phone number is in the “How to Order” section at this site: http://www.elink.ibmlink.ibm.com/pbl/pbl

This information was current at the time of publication, but is continually subject to change. The latest information may be found at the Redbooks Web site.

IBM Intranet for Employees

IBM employees may register for information on workshops, residencies, and Redbooks by accessing the IBM Intranet Web site at http://w3.itso.ibm.com/ and clicking the ITSO Mailing List button. Look in the Materials repository for workshops, presentations, papers, and Web pages developed and written by the ITSO technical professionals; click the Additional Materials button. Employees may access MyNews at http://w3.ibm.com/ for redbook, residency, and workshop announcements.

IBM Redbooks fax order form

Please send me the following:

Title	Order Number	Quantity

First name	Last name	
Company		
Address		
City	Postal code	Country
Telephone number	Telefax number	VAT number

☐ Invoice to customer number

☐ Credit card number

Credit card expiration date	Card issued to	Signature

We accept American Express, Diners, Eurocard, Master Card, and Visa. Payment by credit card not available in all countries. Signature mandatory for credit card payment.

Index

Numerics

A

B

C

D

E

F

G

H

I

L

M

N

O

P

Q

R

W

X

Z

IBM Redbooks evaluation

V4 TCP/IP for AS/400: More Cool Things Than Ever
SG24-5190-00

Your feedback is very important to help us maintain the quality of IBM Redbooks. **Please complete this questionnaire and return it using one of the following methods:**

- Use the online evaluation form found at http://www.redbooks.ibm.com/
- Fax this form to: USA International Access Code + 1 914 432 8264
- Send your comments in an Internet note to redbook@us.ibm.com

Which of the following best describes you?
_ Customer _ Business Partner _ Solution Developer _ IBM employee
_ None of the above

Please rate your overall satisfaction with this book using the scale:
(1 = very good, 2 = good, 3 = average, 4 = poor, 5 = very poor)

Overall Satisfaction _____

Please answer the following questions:

Was this redbook published in time for your needs? Yes___ No___

If no, please explain:

_____

_____

_____

_____

What other Redbooks would you like to see published?

_____

_____

_____

Comments/Suggestions: (THANK YOU FOR YOUR FEEDBACK!)

_____

_____

_____

_____

_____

SG24-5190-00

Printed in the U.S.A.

V4 TCP/IP for AS/400: More Cool Things Than Ever

SG24-5190-00