FlowPoint™

FlowPoint™ DSL Router Family

Command Line Interface

Second Edition (February, 1999)

Copyright

Trademarks

180 Knowles Drive, Suite 100
Los Gatos, California 95030
Telephone: (408) 364-8300
Fax: (408) 364-8301
Email: support@flowpoint.com
Web Site: www.flowpoint.com

Flowpoint is a fully owned subsidiary of Cabletron Systems.

P/N 222-00549-01

Federal Communications Commission (FCC)

Part 15 CLASS B Statement

Section 15.105(b) of the Code of Federal Regulations

NOTE: This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant of Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

CAUTION: Any changes or modifications not expressly approved by the party responsible for this device could void the user's authority to operate this equipment.

Canadian D.O.C. Notice

This product conforms with Canadian Class B emissions regulations.

Ce produit se conforme aux réglements d'émission canadienne classe B.

Instructions for Trained Service Personnel Only

CAUTION: Danger of explosion if battery is incorrectly placed. Replace only with the same or equivalent type recommended by the manufacturer. Dispose of used batteries according to the manufacturer's instructions.

Approvals

Safety: EN60950, UL 1950, CUL to CSA 22.2 No. 950
Emissions: FCC Part 15 Class B, EN55022/CISPR22 Class B
Immunity: EN50082-1

FlowPoint Corporation Program License Agreement

IMPORTANT: Before utilizing this product, carefully read this License Agreement.

This document is an agreement between you, the end user, and FlowPoint Corporation ("FlowPoint") that sets forth your rights and obligations with respect to the FlowPoint software program ("the Program") contained in this package. The Program may be contained in firmware, chips, or other media. BY UTILIZING THE ENCLOSED PRODUCT, YOU ARE AGREEING TO BECOME BOUND BY THE TERMS OF THIS AGREEMENT, WHICH INCLUDES THE LICENSE AND THE LIMITATION OF WARRANTY AND DISCLAIMER OF LIABILITY. IF YOU DO NOT AGREE TO THE TERMS OF THIS AGREEMENT, PROMPTLY RETURN THE UNUSED PRODUCT TO THE PLACE OF PURCHASE FOR A FULL REFUND.

FlowPoint Software Program License

1. LICENSE. You have the right to use only the one (1) copy of the Program provided in this package subject to the terms and conditions of this License Agreement.

You may not copy, reproduce or transmit any part of the Program except as permitted by the Copyright Act of the United States or as authorized in writing by FlowPoint.

2. OTHER RESTRICTIONS. You may not reverse engineer, decompile, or disassemble the Program.

3. APPLICABLE LAW. This License Agreement shall be interpreted and governed under the laws and in the state and federal courts of New Hampshire. You accept the personal jurisdiction and venue of the New Hampshire courts.

Exclusion of Warranty and Disclaimer of Liability

1. EXCLUSION OF WARRANTY. Except as may be specifically provided by FlowPoint in writing, FlowPoint makes no warranty, expressed or implied, concerning the Program (including its documentation and media).

FLOWPOINT DISCLAIMS ALL WARRANTIES, OTHER THAN THOSE SUPPLIED TO YOU BY FLOWPOINT IN WRITING, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WITH RESPECT TO THE PROGRAM, THE ACCOMPANYING WRITTEN MATERIALS, AND ANY ACCOMPANYING HARDWARE.

2. NO LIABILITY FOR CONSEQUENTIAL DAMAGES. IN NO EVENT SHALL FLOWPOINT CORPORATION ("FLOWPOINT") OR ITS SUPPLIERS BE LIABLE FOR ANY DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS, PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, SPECIAL, INCIDENTAL, CONSEQUENTIAL, OR RELIANCE DAMAGES, OR OTHER LOSS) ARISING OUT OF THE USE OR INABILITY TO USE THIS FLOWPOINT PRODUCT, EVEN IF FLOWPOINT HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. BECAUSE SOME STATES DO NOT ALLOW THE EXCLUSION OR LIMITATION OF LIABILITY FOR CONSEQUENTIAL OR INCIDENTAL DAMAGES, OR IN THE DURATION OR LIMITATION OF IMPLIED WARRANTIES IN SOME INSTANCES, THE ABOVE LIMITATION AND EXCLUSIONS MAY NOT APPLY TO YOU.

United States Government Retricted Rights

The enclosed product (a) was developed solely at private expense; (b) contains "restricted computer software" submitted with restricted rights in accordance with section 52227-19 (a) through (d) of the Commercial Computer Software - Restricted Rights Clause and its successors, and (c) in all respects is proprietary data belonging to FlowPoint and/or its suppliers.

For Department of Defense units, the product is licensed with "Restricted Rights" as defines in the DoD Supplement to the Federal Acquisition Regulations, Section 52.227-7013 (c) (1) (ii) and its successors, and use, duplication, disclosure by the Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at 252-227-7013.

Warranties

Limited Warranty on Media and Damages Disclaimer

FlowPoint or its distributors or resellers will repair or replace free of charge any defective recording medium on which the Software is recorded if the medium is returned to FlowPoint or its distributor or reseller within ninety (90) days after the purchase of License for the Software. This warranty does NOT cover defects due to accident, or abuse occurring after your receipt of the Software. THIS WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS AND YOU MAY ALSO HAVE OTHER RIGHTS WHICH MAY VARY FROM STATE TO STATE.

Limited Warranty on Hardware

FlowPoint warrants that Products delivered hereunder shall be free from defects in materials and workmanship for a period of one (1) year from the date of purchase. The liability of FlowPoint is limited to replacing or repairing, at Manufacturer's option, any defective Products that are returned F.O.B. Manufacturer's factory, California. In no case are Products to be returned without first obtaining permission and a customer return material authorization number from Manufacturer.

THIS WARRANTY DOES NOT APPLY TO DEFECTS DUE DIRECTLY OR INDIRECTLY TO MISUSE, ABUSE, NEGLIGENCE, ACCIDENT, REPAIRS OR ALTERATIONS MADE BY THE CUSTOMER OR ANOTHER PARTY OR IF THE FLOWPOINT SERIAL NUMBER HAS BEEN REMOVED OR DEFACED. THIS WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS AND YOU MAY ALSO HAVE OTHER RIGHTS WHICH MAY VARY FROM STATE TO STATE.

EXCEPT FOR THE WARRANTY SET FORTH HEREIN, MANUFACTURER DISCLAIMS ALL WARRANTIES WITH REGARD TO THE PRODUCTS, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTIBILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Hardware and Software Limitations

FlowPoint does not warrant that the Software will be free from error or will meet your specific requirements. You assume complete responsibility for decisions made or actions taken based on information obtained using the Software. Any statements made concerning the utility of the Software are not to be construed as unexpressed or implied warranties.

FLOWPOINT SHALL NOT BE RESPONSIBLE OR LIABLE WITH RESPECT TO ANY SUBJECT MATTER OF THIS SOFTWARE LICENSE AGREEMENT, THE HARDWARE, OR THE AGREEMENTS OF WHICH THEY ARE A PART OR ANY MEDIA ATTACHMENT, PRODUCT ORDER, SCHEDULE OR TERMS OR CONDITIONS RELATED THERETO UNDER ANY CONTRACT, NEGLIGENCE, STRICT LIABILITY OR OTHER THEORY: A) FOR LOSS OR INACCURACY OF DATA OR (EXCEPT FOR RETURN OF AMOUNTS PAID TO FLOWPOINT THEREFORE), COST OF PROCUREMENT OF SUBSTITUTE GOODS, SERVICES, OR TECHNOLOGY, B) FOR ANY INDIRECT, INCIDENTAL OR CONSEQUENTIAL DAMAGES INCLUDING BUT NOT LIMITED TO LOSS OF REVENUES AND LOSS OF PROFITS; HOWEVER CAUSED, WHETHER FOR BREACH OF WARRANTY, BREACH OF CONTRACT, REPUDIATION OF CONTRACT, NEGLIGENCE OR OTHERWISE.

NEITHER FLOWPOINT NOR ANY OF ITS REPRESENTATIVES, DISTRIBUTORS OR OTHER RESELLERS MAKES OR PASSES ON ANY WARRANTY OR REPRESENTATION ON BEHALF OF FLOWPOINT'S THIRD PARTY SUPPLIERS.

Post Warranty Services

Contact FlowPoint for information regarding post-warranty hardware and software services.

Preface

About This Book

The *Command Line Interface* contains information on the syntax and use of the Command Line Interface for the family of DSL routers. It provides the steps and information needed to configure the Router software and troubleshoot problems using the Command Line Interface. Configuration of network connections, bridging, routing, and security features are essentially the same for all DSL routers, unless otherwise noted. The book also provides detailed information about the system's bridging, routing, addressing, and security operations.

This book is intended for small and home office users, remote office users, and other networking professionals who are installing and maintaining bridged and routed networks.

How This Book Is Organized

This guide is intended to help you configure and manage the router using the Command Line Interface. The guide assumes that you have read the information about the router and installed the hardware using the *Internet Quick Start Guide*. The guide is divided into eight parts:

Introduction

Describes the features of the Command Line Interface.

Advanced Topics

Contains additional information on topics such as interoperability, routing and bridging operations, PAP/CHAP security negotiation, bandwidth management, protocol conformance, and the file system.

Planning for Router Configuration

Provides information unique to configuration using the Command Line Interface including worksheets for collecting required information.

Configuring Router Software

Describes how to configure the router using the Command Line Interface.

Configuring Special Features

Describes how to configure features such as Bridging Filtering, RIP, DHCP, NAT, Management Security, Software Options Keys, Encryption, IP Filtering, and L2TP Tunneling.

Command Line Interface Reference

Describes the syntax of each command and the results when the command is entered.

Managing the Router

Describes SNMP management capabilities, TFTP client and server, TELNET support and how to upgrade the system software, boot code, backup and restore configuration files, FLASH memory recovery procedures, and batch file command execution.

Troubleshooting

Describes diagnostic tools used for identifying and correcting hardware and software problems.

Reference

User Guide
Contains an overview of the Router's software and hardware features and details on hardware installation and software configuration using the Windows-based Configuration Manager.

Quick Start Guide
Describes the configuration process involved in setting up a specific router model.

Typographic Conventions

The following figure summarizes the conventions used in this guide:

Item	Type Face	Example
Words defined in glossary, book titles, figure captions, command reference parameters.	Italics	Refer to *Advanced Features* system name *name*
Keywords in command reference instructions	Bold	Example: **save**
Examples showing you what to type and what is displayed on the terminal.	Mono-spaced font	Enter the following command: `remote listIpRoute hq`
File names	Upper case	Copy file CFGMGR.EXE

Table of Contents

Introduction

This guide provides steps and information needed to configure the DSL or Dual Ethernet router software, using the **Command Line Interface**[1].

The command Line Interface covers the following basic configuration topics:

- Setting of names, passwords, PVC numbers, and link and network parameters
- Configuration of specific details within a protocol, such as IP or IPX addresses and IP protocol controls
- Activation of bridging and routing protocols
- Enabling of the Internet Firewall filter with IP routing

The Command Line Interface also provides the following advanced features:

- Manage the router's file system
- Set bridging filters
- Configure the type of DSL technology specific to your router (ADSL, SDSL, etc.)
- Configure the Dual Ethernet router
- Issue online status commands
- Monitor error messages
- Set RIP options
- Configure DHCP
- Configure NAT
- Configure Telnet/SNMP security
- Configure host mapping
- Configure IP multicast
- Create and execute script files
- Configure encryption
- Configure IP filtering
- Configure L2TP tunneling
- Enable Software Options Keys

---

1. The Microsoft® Windows™-based **Configuration Manager** program (featuring an easy-to-use, point-and-click GUI interface) provides another way to configure the router's software. Please refer to the *User Guide* if you intend to use Configuration Manager as your primary configuration tool.

Command Line Interface Access

You can access the Command Line Interface from:

- A terminal session running under Windows (for local access)
- The terminal window from the Configuration Manager (for local access) (see note 2)
- An ASCII terminal (for local access)
- A TELNET session (for remote access)

Note 1: For local access, the PC or ASCII terminal is connected to the **Console** port. This connection and the required communications settings are thoroughly described in the *User Guide*, Appendix C, *Accessing the Command Line Interface*.

Note 2: If you wish to access the terminal window from within the Configuration Manager, click **Tools** and **Terminal Window** from the main menu. The menu selection Commands provide shortcuts to most of the commands described in this manual. These shortcuts will substantially reduce the amount of typing.

Chapter 1. Advanced Topics

This chapter provides information on advanced topics useful to network administrators. Refer to the *User Guide* for a general overview of the router basic features.

Interoperability

The router uses industry-wide standards to ensure compatibility with routers and equipment from other vendors. To interoperate, the router supports standard protocols on the physical level, data link level for frame type or encapsulation method, and network level. For two systems to communicate directly, they must use the same protocol at each level. Most protocols do not support negotiable options, except for PPP.

The physical protocol level includes hardware and electrical signaling characteristics. This support is provided by the router Ethernet and modem hardware interfaces.

The data link protocol level defines the transmission of data packets between two systems over the LAN or WAN physical link.

The frame type or encapsulation method defines a way to run multiple network-level protocols over a single LAN or WAN link. The router supports the following WAN encapsulations:

- PPP (VC multiplexing)
- PPP (LLC multiplexing)
- RFC 1483 (for ATM)
- RFC 1483 with MAC Encapsulation Routing (for ATM)
- FRF8 (for ATM)
- RFC 1490 (for Frame Relay)
- RFC 1490 with MAC Encapsulated Routing (for Frame Relay)

Routing

The network protocol provides a way to route user data from source to destination over different LAN and WAN links. Routing relies on routing address tables to determine the best path for each packet to take.

The routing tables can be seeded; i.e., addresses for remote destinations are placed in the table along with path details and the associated costs (path latency).

The routing tables are also built dynamically; i.e., the location of remote stations, hosts, and networks are updated from broadcast packet information.

Routing helps to increase network capacity by localizing traffic on LAN segments. It also provides security by isolating traffic on segmented LAN. Routing extends the reach of networks beyond the limits of each LAN segment.

Numerous network protocols have evolved and within each protocol are associated protocols for routing, error handling, network management, etc. The following chart displays the networking and associated protocols supported by the router.

Network Protocol	Associated Protocol	Description
Internet Protocol (IP)	Routing Information Protocol (RIP)	Protocol used to maintain a map of the network
	Address Resolution Protocol (ARP)	Maps IP addresses to datalink addresses
	Reverse Address Resolution Protocol (RARP)[a]	Maps data link addresses to IP addresses
	Internetwork Control Message Protocol (ICMP)	Diagnostic and error reporting/ recovery
	Simple Network Management Protocol (SNMP)	Network Management
Internet Packet Exchange (IPX)	Routing Information Protocol (RIP)[b]	Protocol used to maintain a map of the network
	Service Advertising Protocol (SAP)	Distributes information about service names and addresses

a Used only during a network boot
b IPX-RIP is a different protocol from IP-RIP and includes time delays

Most of the router's operation on each protocol level is transparent to the user. Some functions are influenced by configuration parameters and these are described in greater details in the following sections.

Bridging

Bridging connects two or more LANs together so that all devices share the same logical LAN segment and network number. The MAC layer header contains source and destination addresses used to transfer frames. An address table is dynamically built and updated with the location of devices when the frames are received.

Transparent bridging allows locally connected devices to send frames to all devices as if they are local.

Bridging allows frames to be sent to all destinations regardless of the network protocols used. It allows protocols that cannot be routed (such as NETBIOS) to be forwarded and allows optimizing internetwork capacity by localizing traffic on LAN segments. A bridge extends the physical reach of networks beyond the limits of each LAN segment. Bridging can increase network security with filtering.

The router bridging support includes the IEEE 802.1D standard for LAN-to-LAN bridging and the Spanning Tree Protocol for interoperability with other vendors' bridge/routers. Bridging is provided over PPP as well as adjacent LAN ports.
Most of the router's bridging operation is transparent. Some functions are influenced by configuration parameters and these are described in greater detail in the following sections.

Bridging and Routing Operation

The router can operate as a bridge, as a router, or as both (sometimes called a brouter).

- The router will operate as a router for network protocols that are enabled for routing (IP or IPX).
- The router will operate as a bridge for protocols that are not supported for routing.
- Routing takes precedence over bridging; i.e., when routing is active, the router uses the packet's protocol address information to route the packet.
- If the protocol is not supported, the router will use the MAC address information to forward the packet.

Operation of the router is influenced by routing and bridging controls and filters set during router configuration as well as automatic spoofing and filtering performed by the router. For example, general IP or IPX routing, and routing or bridging from specific remote routers are controls set during the configuration process.

Spoofing and filtering, which minimize the number of packets that flow across the WAN, are performed automatically by the router. For example, RIP routing packets and certain NetBEUI packets are spoofed even if only bridging is enabled.

Bridging and Routing Configuration Settings

The router can be configured to perform general routing and bridging while allowing you to set specific controls.

One remote router is designated as the outbound default bridging destination. All outbound bridging traffic, with an unknown destination, is sent to the default bridging destination. Bridging from specific remote routers can be controlled by enabling/disabling bridging from individual remote routers.

Routing is performed to all remote routers entered into the remote router database. All routing can be enabled/disabled with a system-wide control.

The following charts describe the operational characteristics of the router, based on configuration settings.

IP/IPX Routing ON Bridging To/From Remote Router OFF

Data Packets Carried	IP (TCP, UDP), IPX
Operational Characteristics	Basic IP, IPX connectivity
Typical Usage	When only IP/IPX traffic is to be routed and all other traffic is to be ignored. For IP, used for Internet access. **Note:** This is the most easily controlled configuration.

IP/IPX Routing ON **Bridging To/From Remote Router ON**

Data Packets Carried	IP/IPX routed; all other packets bridged
Operational Characteristics	IP/IPX routing and allows other protocols, such as NetBEUI (that can't be routed), to be bridged.
Typical Usage	When only IP/IPX traffic is to be routed but some non-routed protocol is required. Used for client/server configurations.

IP/IPX Routing OFF **Bridging To/From Remote Router ON**

Data Packets Carried	All packets bridged
Operational Characteristics	Allows protocols, such as NetBEUI (that can't be routed) to be bridged.
Typical Usage	Peer-to-peer bridging and when the remote end supports only bridging.

Point-To-Point Protocol (PPP)

PPP is an industry standard WAN protocol for transporting multi-protocol datagrams over point-to-point connections. PPP defines a set of protocols, such as security and network protocols, that can be negotiated over the connection. PPP includes the following protocols:

- Link Control Protocol (LCP) to negotiate PPP; i.e., establish, configure and test the datalink connection.
- Network Control Protocols (NCPs), such as:

  TCP/IP routing Internet Protocol Control Protocol (IPCP)

  IPX routing Control Protocol (IPXCP)

  Bridge Control Protocol (BNCP)
- Security Protocols including PAP and CHAP

For a more detailed description of the router's implementation of some of these protocols, please read the following section. A list of PPP protocol conformance is included in the section *Protocol Conformance*.

PAP/CHAP Security Authentication

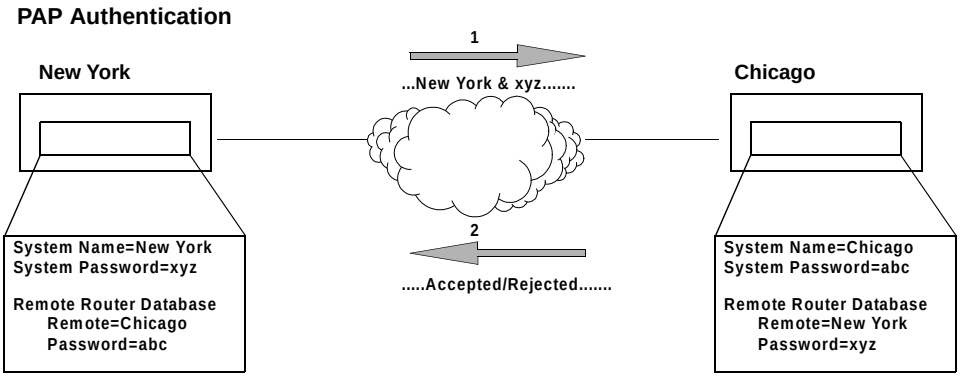
Password Authentication Protocol (PAP) and Challenge Handshake Authentication Protocol (CHAP) under PPP are supported by the router. However, security authentication may or may not be needed depending on the requirements of the remote end.

The nature of the connection in a DSL environment (traffic occurs on a dedicated line/virtual circuit) does not require authentication unless specifically required by the remote end, the ISP, or the NSP. When not required, security is disabled with the command **remote disauthen**.

General Security Authentication Information

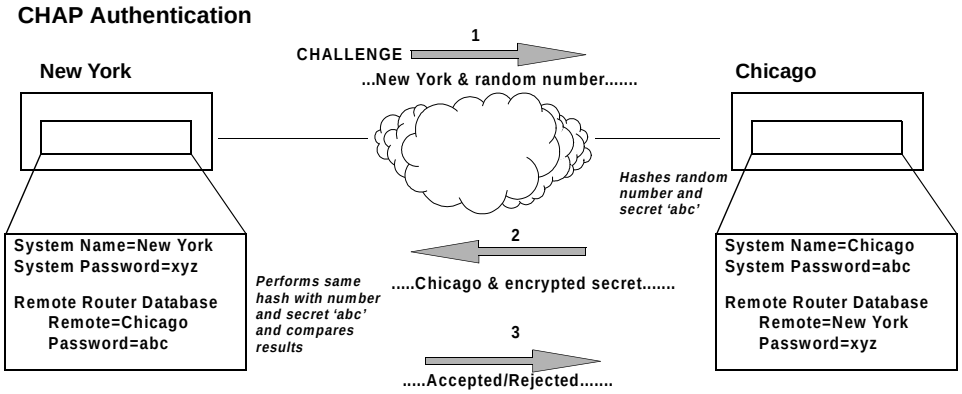
Security authentication may be required by the remote end. The following information describes how authentication occurs.

PAP provides verification of passwords between routers using a 2-way handshake. One router (peer) sends the system name and password to the other router. Then the other router (known as the authenticator) checks the peer's password against the configured remote router's password and returns acknowledgment.



CHAP is more secure than PAP as unencrypted passwords are not sent across the network. CHAP uses a 3-way handshake. One router (known as the authenticator) challenges the other router (known as the peer) by generating a random number and sending it along with the system name. The peer then applies a one-way hash algorithm to the random number and returns this encrypted information along with the system name.

The authenticator then runs the same algorithm and compares the result with the expected value. This authentication method depends upon a password or secret known only to both ends.



Security Configuration Settings

The router has one default system password used to access any remote router. This “system authentication password” is utilized by remote sites to authenticate the local site. The router also allows you to assign a unique “system override password” used only when connecting to a specific remote router for authentication by that remote site. Each remote router entered in the remote router database has a password used when the remote site attempts to gain access to the local router. This “remote authentication password” is utilized by the router to authenticate the remote site.

Each remote router entered in the remote router database also has a minimum security level, known as the “remote authentication protocol”, that must be negotiated before the remote router gains access to the local router. In addition, a system-wide control, “system authentication protocol”, is available for overriding the minimum security level in the entire remote router database.

Authentication Process

The authentication process occurs regardless of whether a remote router connects to the local router or vice versa, and even if the remote end does not request authentication. It is a bi-directional process, where each end can authenticate the other using the protocol of its choice (provided the other end supports it).

During link negotiation (LCP), each side of the link negotiates what protocol is to be used for authentication during the connection. If both the system and the remote router have PAP authentication, then PAP authentication is negotiated.

Otherwise, the router *always* requests CHAP authentication first; if refused, PAP will be negotiated. If the remote end does not accept either PAP or CHAP, the link is dropped; i.e., the router does not communicate without a minimum security level. On the other hand, the router will accept any authentication scheme required by the remote node, including no authentication at all.

During the authentication phase, each side of the link can request authentication using the method they negotiated during LCP.

For CHAP, the router issues a CHAP challenge request to the remote side. The challenge includes the system name and random number. The remote end, using a hash algorithm associated with CHAP, transforms the name and number into a response value. When the remote end returns the challenge response, the router can validate the response challenge value using the entry in the remote router database. If the response is invalid, the call is disconnected. If the other end negotiated CHAP, the remote end can, similarly, request authentication from the local router. The router uses its system name and password to respond to CHAP challenge.

For PAP, when a PAP login request is received from the remote end, the router checks the remote router PAP security using the remote router database. If the remote router is not in the remote router database or the remote router password is invalid, the call is disconnected. If the remote router and password are valid, the local router acknowledges the PAP login request.

If PAP was negotiated by the remote end for the remote-side authentication, the router will issue PAP login requests *only* if it knows the identity of the remote end. The identity is known if the call was initiated from the router or the remote end returned a successful CHAP challenge response. For security reasons, the router will *never* identify itself using PAP without first knowing the identity of the remote router.
If PAP was negotiated by the remote end for the local side of the authentication process and the minimum security level is CHAP, as configured in the remote router database, the link is dropped for a security violation.

Protocol Conformance

Protocol Standards

The router conforms to RFCs designed to address performance, authentication, and multi-protocol encapsulation. The following RFCs are supported:

- RFC 1058 Routing Information Protocol (RIP)
- RFC 1144 Compressing TCP/IP headers (Van Jacobson)
- RFC 1220 Bridging Control Protocol (BNCP)
- RFC 1332 IP Control Protocol (IPCP)
- RFC 1334 Password Authentication Protocol & Challenge Handshake Authentication Protocol (PAP/CHAP)
- RFC 1483 Multiprotocol Encapsulation over ATM Adaptation Layer 5
- RFC 1490 Multiprotocol Interconnect over Frame Relay
- RFC 1552 Novell IPX Control Protocol (IPXCP)
- RFC 1577 Classical IP and ARP over ATM
- RFC 1661 Point-to-Point Protocol (PPP)
- RFC 1723 RIP Version 2
- RFC 1962 PPP Compression Control Protocol (CCP)
- RFC 1973 PPP in Frame Relay
- RFC 1974 Stac LZS compression protocol
- RFC 1990 Multi-Link Protocol (MLP)
- RFC 2131 and 2132 Dynamic Host Configuration Protocol (DHCP)

IP Routing

IP routing support, in conformance with RFC 791, provides the ability to process TCP/IP frames at the network layer for routing. IP routing support includes the Routing Interface Protocol (RIP), in conformance with RFC 1058 (RIP v.1) and RFC 1723 (RIP v.2).

IPX Routing

IPX routing conforms to the Novell® NetWare™ IPX Router Development Guide, Version 1.10.

Encapsulation Options

This section describes in technical terms the format of each packet associated with a particular encapsulation option supported by the router.

The encapsulation type for each remote entry is defined using the **remote setProtocol** command.

PPP

Each packet begins with a one or two-byte protocol ID. Typical IDs are:

0xc021 -- LCP

0x8021 -- IPCP

0x0021 -- IP

0x002d -- Van Jacobson compressed TCP/IP

0x002f -- Van Jacobson uncompressed TCP/IP

0x8031 -- Bridge NCP

0x0031 -- Bridge Frame

The command for this encapsulation option is:

remote setProtocol PPP <*remoteName*>

Note: With PPP over ATM, the address and control fields (i.e. FF03) are never present, including in LCP packets.

PPPLLC

This protocol (LLC-multiplexed) allows PPP traffic to be carried simultaneously with other traffic on a single virtual circuit (as opposed to the PPP method of encapsulation – VC multiplexing - which dedicates a virtual circuit to PPP traffic only).

Each PPP packet is prepended with the sequence 0xFEFE03CF. Thus, an LLC packet has the format: 0xFEFE03CF 0xC021.

The command for this encapsulation option is:

remote setProtocol PPPLLC <*remoteName*>

RFC 1483 or RFC 1490

Bridging

User data packets are prepended by the sequence 0xAAAA0300 0x80c20007 0x0000 followed by the Ethernet frame containing the packet.

802.1D Spanning Tree packets are prepended with the header 0xAAAA0300 0x80C2000E.

Routing

IP packets are prepended with the header 0xAAAA0300 0x00000800.

IPX packets are prepended with the header 0xAAAA0300 0x00008137.

The commands for this encapsulation option are:

remote setProtocol RFC1483 <*remoteName*> (for ATM)

remote setProtocol FR <*remoteName*> (for Frame Relay - RFC 1490)

MAC Encapsulated Routing: RFC 1483MER (ATM) or RFC 1490MER (Frame Relay)

MER encapsulation allows IP packets to be carried as bridged frames, but does not prevent bridged frames from being sent as well, in their normal encapsulation format: RFC 1483 (ATM) or RFC 1490 (Frame Relay).

If IP routing is enabled, then IP packets are prepended with the sequence 0xAAAA0300 0x80c20007 0x0000 and sent as bridged frames. If IP routing is not enabled, then the packets appear as bridged frames.

The commands for this encapsulation option are:

remote setProtocol RFC1483MER <*remoteName*> (for ATM)

remote setProtocol MER (for Frame Relay)

FRF8

IP Packets have prepended to them the following sequence: 0x03CC.

The command for this encapsulation option is:

remote setprotocol FRF8 <*remoteName*>

Note: This protocol allows sending ATM over Frame Relay

rawIP

IP packets do not have any protocol headers prepended to them; they appear as IP packets on the wire. Only IP packets can be transported since there is no possible method to discriminate other types of packets (bridged frames or IPX).

The command for this encapsulation option is: **remote setProtocol rawIP** <*remoteName*>

System Files

The router's file system is a DOS-compatible file system. The following list describes the contents of the file system:

- **SYSTEM.CNF**
  Configuration files containing:

  DOD Remote Router Database
  SYS System Settings: name, message, authentication method, and passwords
  ETH Ethernet LAN configuration settings

- **DHCP.DAT**
  DHCP files

- **FILTER.DAT**
  Bridge filters

- **KERNEL.F2K**
  Router system software (KERNEL.FP1 for IDSL routers)

- **ETH.DEF**
  File used by the manufacturer to set a default Ethernet configuration

- **ASIC.AIC**
  Firmware for the xDSL modem or ATM interface

- **ATM.DAT**
  ATM configuration file

Any file contained within the system may be retrieved or replaced using the TFTP protocol. Specifically, configuration files and the operating system upgrades can be updated. Only one copy for the router software is allowed in the router's FLASH memory.

Refer to *Chapter 6. Managing the Router* for details on software upgrades, booting router software, copying configuration files, and restoring router software to FLASH memory.

Bridging Filtering

You can control the flow of packets across the router using bridging filtering. Bridging filtering lets you “deny”or “allow” packets to cross the network based on position and hexadecimal content within the packet. This enables you to restrict or forward messages with a specified address, protocol, or data content. Common uses are to prevent access to remote networks, control unauthorized access to the local network, and limit unnecessary traffic.

For example, it might be necessary to restrict remote access for specific users on the local network. In this case, bridging filters are defined using the local MAC address for each user to be restricted. Each bridging filter is specified as a “deny” filter based on the MAC address and position of the address within the packet. Deny filtering mode is then enabled to initiate bridge filtering. Every packet with one of the MAC addresses would not be bridged across the router until the deny filtering mode was disabled.

Similarly, protocol filtering can be used to prevent a specific protocol from being bridged. In this case, the protocol ID field in a packet is used to deny or allow a packet. You can also restrict, for example, the bridging of specific broadcast packets.

Unique System Passwords

As described in the section *Security Configuration Settings* of this chapter, you can specify a unique system override password for a remote router with the command **remote SetOurPasswd**. This “system override password” is used instead of the general system password *only* when connecting to a specific remote router. This allows you to set a unique CHAP or PAP authentication password for authentication of the local site by the remote site *only* when the router connects to that remote site.

A common use would be to set a password assigned to you by Internet Service Providers (ISPs). Similarly, the system name of the local router can be overridden when connecting to a specific remote with the command **remote setoursysname.**

Chapter 2. Planning For Router Configuration

This chapter describes the terminology and the information that you need to collect before configuring the router. The information needed to configure the router is contingent on the chosen Link Protocol. It is therefore important to know which Link Protocol you are using (this is determined by your Network Service Provider) to be able to refer to the configuration sections that apply to your setup.

When configuring the router using the Command Line Interface, planning is similar to the process described for Configuration Manager with very few exceptions.

Important Terminology

You should familiarize yourself with the following terminology as it will be used throughout this chapter.

Target router
Router that you are configuring. Also referred to as **local** router.

Remote routers
All the routers to which the target (local) router may connect.

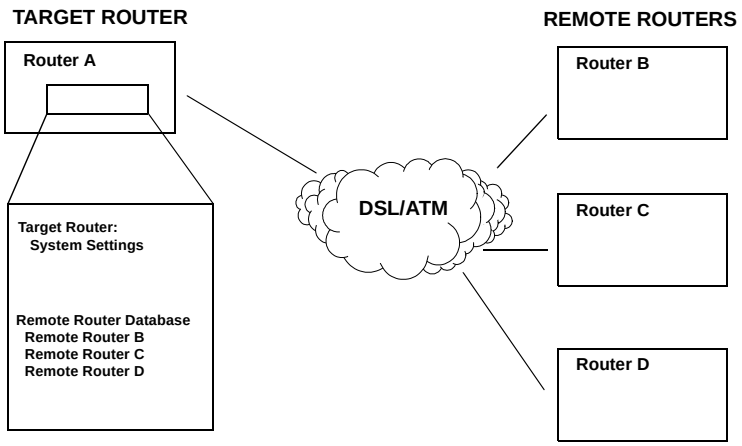
Remote router database
Database which resides in the target router and contains information about the remote routers to which the target router may connect.

Remote router entry
Entry about a remote router in the target router database. A remote router entry defines:

- Connection parameters
- Security features
- Route addressing and bridging functions

The following diagram illustrates these key words and concepts.

Configuration Process for Router A



Collect your Configuration Information

This section describes the configuration information associated with each Link Protocol/Network Protocol combination and also provides configuration information for the Dual Ethernet router. It is organized as follows:

Link Protocols/Network Protocols Configurations

1. Determine which Link Protocol/Network Protocol association you are using. This information is obtained from your Network Service Provider (NSP).

2. To locate the Link/Network information that applies to your situation, use the following listing:

   PPP with:

   - *IP Routing Network Protocol, page 28*
   - *IP Routing Network Protocol, page 28*
   - *Bridging Network Protocol, page 32*

   RFC 1483 or RFC 1490 with:

   - *IP Routing Network Protocol, page 40*
   - *IPX Routing Network Protocol, page 35*
   - *Bridging Network Protocol, page 32*

   MAC Encapsulated Routing: RFC 1483MER or RFC 1490MER with:

   - *IP Routing Network Protocol, page 40*

   FRF8 Link Protocol with:

   - *IP Routing Network Protocol, page 40*

3. Collect the information applicable to your Link/Network Protocol association. This information will be used later in conjunction with the *Configuration Tables* for easy configuration of your router based on your Link/Network protocol. These configuration tables provide step-by-step instructions for a basic configuration for each Link/Network protocol.

   Note: Use the blank Network Information Worksheets in Appendix A to collect your network information.

Dual Ethernet Router Configuration

Configure the router as a Bridge, page 42

Configure the router for IP Routing, page 42

PPP Link Protocol (over ATM or Frame Relay)

The PPP Link Protocol is an encapsulation method that can be used over ATM (for ATM routers) or Frame Relay (for Frame Relay routers)

Combined with the IP, IPX, or Bridging Network Protocols, PPP over ATM and PPP over Frame Relay share the same configuration characteristics, except for the connection identifiers: VPI/VCI numbers are used for ATM and a DLCI number is used for Frame Relay.

Select the Network Protocol that applies to your situation: IP, or IPX, or Bridging. Collect the information described in the appropriate section. This data will be later used to configure your router using the Command Line Interface commands (see *Configuration Tables*, Chapter 3).

IP Routing Network Protocol

♦ System Names and Authentication Passwords

For the Target Router

This information is defined by the user.
You must choose a name and authentication password for the target router. They are used by a remote router to authenticate the target router.

For the Remote Site(s)

This information is obtained from the Network Service Provider.
For each remote site, you must have the site name and its authentication password. They are used by the target router to authenticate the remote end. The name and password are used in both PAP and CHAP authentication. Please refer to the diagram on page 19 to see how this information is used.

Note 1: A sample configuration containing Names and Passwords is provided in the section *Sample Configuration 1 — PPP with IP and IPX*, Chapter 3

Note 2: If the ISP does not support the authentication of the ISP system by the caller, use the command **remote disauthen** <*remoteName*> to disable the authentication.

♦ VPI and VCI Numbers (for ATM routers)

Your router may have been preconfigured with VPI/VCI numbers. If not, you will have to obtain these numbers from your Network Service Provider and then configure them.

If you are connecting to multiple remote sites, you will need to obtain additional VPI and VCI numbers from your Network Service Provider and/or Network Access Provider. These numbers identify the remote destination and must, therefore, be unique for each remote.

♦ DLCI (for Frame Relay routers)

The DLCI number applies to Frame Relay routers only.

Your Network Service Provider or your Network Access Provider will provide you with a DLCI (Data Link Connection Identifier). The DLCI is an address identifying your connection.

♦ DNS Internet Account Information (optional)

This information is obtained from your Network Service Provider.

Consult with you Network Service Provider to find out if you need to enter the following information:

- DNS Server Address
- DNS Second Server Address
- DNS Domain Name

♦ IP Routing Addresses

For the Ethernet Interface

This information is defined by the user or your Network Administrator.

Ethernet IP Address (local LAN)
An Ethernet LAN IP address and subnet mask are required for the router's local Ethernet LAN connection.

Note: An **Ethernet route** is usually defined when there are multiple routers on the Ethernet which cannot exchange routing information between them. This feature is normally not used except in very special circumstances.

For the WAN Interface

This information is defined by the Network Service Provider.

Source (Target/Local) WAN Port Address
If Network Address Translation (NAT) is enabled, you must specify a source WAN IP address for the WAN connection to the remote router if IP address negotiation under PPP does not provide one.

Check with your system administrator for details on whether the router must communicate in numbered or unnumbered mode and what addresses are required.

Remote WAN Address
You may need to specify a remote WAN IP address for the WAN connection to the remote router depending on IP address negotiation under PPP.

Check with your system administrator for details on whether the router must communicate in numbered or unnumbered mode and what addresses are required.

TCP/IP Remote Routes
An IP route includes an IP address, subnet mask, and metric (a number representing the perceived cost in reaching the remote network or station).

A **TCP/IP Default Route** should be designated in the routing table for all traffic that cannot be directed to other specific routes.

You will need to define the default route to a remote router or, if required due to special circumstances, define an Ethernet gateway. There can be only one default route specified.

IPX Routing Network Protocol

♦ System Names and Authentication Passwords

For the Target Router

This information is defined by the user.
You must choose a name and authentication password for the target router. They are used by a remote router to authenticate the target router.

For the Remote Site(s)

This information is obtained from the Network Service Provider.
For each remote site, you must have the site name and its authentication password. They are used by this target router to authenticate the remote end. The name and password are used in both PAP and CHAP authentication. Please refer to the diagram on page 19 to see how this information is used.

Note 1: A sample configuration containing Names and Passwords is provided in the section *Sample Configurations,* Chapter 3

Note 2: If the ISP does not support the authentication of the ISP system by the caller, use the command **remote disauthen** <*remoteName*> to disable the authentication.

♦ VPI and VCI Numbers

Your router may have been preconfigured with VPI/VCI numbers. If not, you will have to obtain these numbers from your Network Service Provider and then configure them.

If you are connecting to multiple remote sites, you will need to obtain additional VPI and VCI numbers from your Network Service Provider and/or Network Access Provider. These numbers identify the remote destination and must, therefore, be unique for each remote.

♦ DLCI (for Frame Relay routers)

The DLCI number applies to Frame Relay routers only.

Your Network Service Provider or your Network Access Provider will provide you with a DLCI (Data Link Connection Identifier). The DLCI is an address identifying your connection.

♦ IPX Routing Entries

These numbers are defined by the Network Administrator.
You will need to obtain the following information (most likely from your network administrator) for IPX Routing.

Note: IPX routes define a path to a specific destination. They are primarily needed by the routers to allow the servers and clients to exchange packets. A path to a file server will be based on the Internal Network Number of the server.

A path to a client will be based on the External Network Number (Ethernet) of the client.

Internal Network Number

It is a logical network number that identifies an individual Novell server. It is needed to specify a route to the services (i.e., file services, print services) that Novell offers. It must be a unique number.

External Network (a.k.a. IPX Network Number)

It refers to a physical LAN/wire network segment to which servers, routers, and PCs are connected (Ethernet cable-to-router segment). It must be a unique number.

WAN Network Number

Important: This number is part of the routing information. It is used to identify the WAN segment between the two routers only. Note that only both routers need to have the WAN Network Number configured.

SAP (Service Advertisement Protocol)

SAP entries should reflect primary logon servers for the clients on the local LAN. Only the servers on the remote side of the link have to be entered. Local servers do not need to be entered.

Frame Type

With local servers on your LAN, make sure to select the proper frame type for the IPX network number. To determine this, consult with your network administrator. When you have only NetWare clients on your LAN, leave the default (802.2) selected as most clients support any type.

The frame type choices are:

802.2	Default recommended by Novell
802.3	Other most common type
DIX	For DEC, Intel, Xerox; this setting is also referred to as “Ethernet II”, and is rapidly becoming obsolete.

Note: Appendix B provides step-by-step information on how to configure IPX routing.

Bridging Network Protocol

♦ System Names and Authentication Passwords

For the Target Router

This information is defined by the user.

You must choose a name and authentication password for the target router. They are used by a remote router to authenticate the target router.

♦ For the Remote Site(s)

This information is obtained from the Network Service Provider.
For each remote site, you must have the site name and its authentication password. They are used by the target router to authenticate the remote end. The name and password are used in both PAP and CHAP authentication. Please refer to the diagram on page 19 to see how this information is used.

Note 1: A sample configuration containing Names and Passwords is provided in the section *Sample Configuration 1 — PPP with IP and IPX*, Chapter 3

Note 2: If the ISP does not support the authentication of the ISP system by the caller, use the command **remote disauthen** <*remoteName*> to disable the authentication.

♦ VPI and VCI Numbers

Your router may have been preconfigured with VPI/VCI numbers. If not, you will have to obtain these numbers from your Network Service Provider and then configure them.

If you are connecting to multiple remote sites, you will need to obtain additional VPI and VCI numbers from your Network Service Provider and/or Network Access Provider. These numbers identify the remote destination and must, therefore, be unique for each remote.

♦ DLCI (for Frame Relay routers)

The DLCI number applies to Frame Relay routers only.

Your Network Service Provider or your Network Access Provider will provide you with a DLCI (Data Link Connection Identifier). The DLCI is an address identifying your connection.

♦ DNS Internet Account Information (optional)

This information is obtained from the Network Service Provider.
Consult with you Network Service Provider to find out if you need to enter the following information:

- DNS Server Address
- DNS Second Server Address
- DNS Domain Name

Note: If you intend to connect to the Internet only, enter this information using the Internet Quick Start configurator.

RFC 1483 / RFC 1490 Link Protocols

The Link Protocol RFC 1483 is a multiprotocol encapsulation method over ATM and is used by ATM routers. RFC 1490 is a multiprotocol encapsulation method over Frame Relay and is used by Frame Relay routers.

RFC 1483 and RFC 1490 combined with the IP, IPX, or Bridging Network Protocols share the same configuration characteristics, except for the connection identifiers: VPI/VCI numbers are used for RFC 1483 and a DLCI number is used for RFC 1490.

Collect the information as described in the appropriate section. This data will be later used to configure your router using the Command Line Interface (see *Configuration Tables*, Chapter 3).

IP Routing Network Protocol

♦ VPI and VCI Numbers (for RFC 1483)

The VPI and VCI numbers apply to ATM routers only.

Your router may have been preconfigured with VPI/VCI numbers. If not, you will have to obtain these numbers from your Network Service Provider and then configure them.

If you are connecting to multiple remote sites, you will need to obtain additional VPI and VCI numbers from your Network Service Provider and/or Network Access Provider. These numbers identify the remote destination and must, therefore, be unique for each remote.

♦ DLCI (for RFC 1490)

The DLCI number applies to Frame Relay routers only.

Your Network Service Provider or your Network Access Provider will provide you with a DLCI (Data Link Connection Identifier). The DLCI is an address identifying your connection.

♦ DNS Internet Account Information (optional)

This information is obtained from the Network Service Provider.

Consult with you Network Service Provider to find out if you need to enter the following information:

- DNS Server Address
- DNS Second Server Address
- DNS Domain Name

♦ IP Routing Entries

For the Ethernet Interface

This information is defined by the user or the Network Administrator.

Ethernet IP Address (Local LAN)
An Ethernet LAN IP address and subnet mask are required for the router's local Ethernet LAN connection.

TCP/IP Ethernet Routes

You normally do not need to define an Ethernet IP route. An Ethernet IP route consists of an IP address, a mask, a metric, and a gateway. An Ethernet route is usually defined when there are multiple routers on the Ethernet which cannot exchange routing information between them.

For the WAN Interface

This information is obtained from the Network Administrator.

Source (Target/Local) WAN Port Address
If NAT is enabled, you must specify a source WAN IP address for the WAN connection to the remote router. Check with your system administrator for details.

If NAT is not enabled, you may need to specify a source WAN IP address for the WAN connection to the remote router. Check with your system administrator for details.

TCP/IP Remote Routes
An IP route includes an IP address, subnet mask, and metric (a number representing the perceived cost in reaching the remote network or station).

A **TCP/IP Default Route** default route should be designated in the routing table for all traffic that cannot be directed to other specific routes. You will need to define the default route to a remote router or, if required due to special circumstances, define an Ethernet gateway. There can be only one default route specified.

IPX Routing Network Protocol

♦ VPI and VCI Numbers (for RFC 1483)

The VPI and VCI numbers apply to ATM routers only.

Your router may have been preconfigured with VPI/VCI numbers. If not, you will have to obtain these numbers from your Network Service Provider and then configure them.

If you are connecting to multiple remote sites, you will need to obtain additional VPI and VCI numbers from your Network Service Provider and/or Network Access Provider. These numbers identify the remote destination and must, therefore, be unique for each remote.

♦ DLCI (for RFC 1490)

The DLCI number applies to Frame Relay routers only.

Your Network Service Provider or your Network Access Provider will provide you with a DLCI (Data Link Connection Identifier). The DLCI is an address identifying your connection.

♦ IPX Routing Entries

The user or the Network Administrator defines this information.

Note: IPX routes define a path to a specific destination. They are primarily needed by the routers to allow the servers and clients to exchange packets. A path to a file server will be based on the Internal Network Number of the server. A path to a client will be based on the External Network Number (Ethernet) of the client.

Internal Network Number

This is a logical network number that identifies an individual Novell server. It is needed to specify a route to the services (i.e. file services, print services) that Novell offers. It must be a unique number.

External Network (a.k.a. IPX Network Number)

This number refers to a physical LAN/wire network segment to which servers, routers, and PCs are connected (Ethernet cable-to-router segment). It must be a unique number.

WAN Network Number

Important: This number is part of the routing information. It is used to identify the WAN segment between the two routers only.

Note: only both routers need to have the WAN Network Number configured.

SAP (Service Advertisement Protocol)

SAP entries should reflect primary logon servers for the clients on the local LAN. Only the servers on the remote side of the link have to be entered. Local servers do not need to be entered.

Frame Type

With local servers on your LAN, make sure to select the proper frame type for the IPX network number. To determine this, consult with your network administrator. When you have only NetWare clients on your LAN, leave the default (802.2) selected as most clients support any type.
The frame type choices are:

802.2 Default recommended by Novell
802.3 Other most common type
DIX For DEC, Intel, Xerox; this setting is also referred to as “Ethernet II”, and is rapidly becoming obsolete.

Bridging Network Protocol

♦ VPI and VCI Numbers (with RFC 1483)

The VPI and VCI numbers apply to ATM routers only.

Your router may have been preconfigured with VPI/VCI numbers. If not, you will have to obtain these numbers from your Network Service Provider and then configure them.

If you are connecting to multiple remote sites, you will need to obtain additional VPI and VCI numbers from your Network Service Provider and/or Network Access Provider. These numbers identify the remote destination and must, therefore, be unique for each remote.

♦ DLCI (with RFC 1490)

The DLCI number applies to Frame Relay routers only.

Your Network Service Provider or your Network Access Provider will provide you with a DLCI (Data Link Connection Identifier). The DLCI is an address identifying your connection.

♦ DNS Internet Account Information (optional)

This information is obtained from the Network Service Provider.

Consult with your Network Service Provider to find out if you need to enter the following information:

- DNS Server Address
- DNS Second Server Address
- DNS Domain Name

MAC Encapsulated Routing: RFC 1483MER / RFC 1490MER Link Protocols

MAC Encapsulated Routing (MER) allows IP packets to be carried as bridged frames (bridged format). The Link Protocol RFC 1483 with MER (referred to as RFC 1483MER) is a multiprotocol encapsulation method over ATM used by ATM routers. RFC 1490 with MER (referred to as RFC 1490MER) is a multiprotocol encapsulation method over Frame Relay used by Frame Relay routers.

RFC 1483MER and RFC 1490MER combined with the IP, IPX, or Bridging Network Protocols share the same configuration characteristics, except for the connection identifiers: VPI/VCI numbers are used for RFC 1483MER and a DLCI number is used for RFC 1490.

Collect the information as described in the appropriate section. This data will be later used to configure your router using the Command Line Interface (see *Configuration Tables*, Chapter 3).

IP Routing Network Protocol

♦ VPI and VCI Numbers (for RFC 1483MER)

The VPI and VCI numbers apply to ATM routers only.

Your router may have been preconfigured with VPI/VCI numbers. If not, you will have to obtain these numbers from your Network Service Provider and then configure them.

If you are connecting to multiple remote sites, you will need to obtain additional VPI and VCI numbers from your Network Service Provider and/or Network Access Provider. These numbers identify the remote destination and must, therefore, be unique for each remote.

♦ DLCI (for RFC 1490MER)

The DLCI number applies to Frame Relay routers only.

Your Network Service Provider or your Network Access Provider will provide you with a DLCI (Data Link Connection Identifier). The DLCI is an address identifying your connection.

♦ DNS Internet Account Information (optional)

This information is obtained from the Network Service Provider.

Consult with your Network Service Provider to find out if you need to enter the following information:

- DNS Server Address
- DNS Second Server Address
- DNS Domain Name

Note: If you intend to connect to the Internet only, enter this information using the Internet Quick Start configurator.

♦ IP Routing Entries

For the Ethernet Interface

This information is defined by the user or the Network Administrator.

Ethernet IP Address (Local LAN)
An Ethernet LAN IP address and subnet mask are required for the router's local Ethernet LAN connection.

TCP/IP Ethernet Routes

You normally do not need to define an Ethernet IP route. An Ethernet IP route consists of an IP address, a mask, a metric, and a gateway. An Ethernet route is usually defined when there are multiple routers on the Ethernet which cannot exchange routing information between them.

For the ATM WAN Interface

This information is obtained from the Network Administrator or the NSP.

Source (Target/Local) WAN Port Address and Mask
You must specify a Source WAN IP address for the WAN connection to the remote router (whether or not NAT is enabled). The Source WAN Address is the address of the local router on the remote network. The mask is the mask used on the remote network. Check with your system administrator for details.

TCP/IP Remote Routes
When using RFC 1483MER or RFC 1490MER, the IP route includes an IP address, subnet mask, metric (a number representing the perceived cost in reaching the remote network or station), and a gateway. The gateway address that you enter is the address of a router on the remote LAN. Check with your system administrator for details.

A **TCP/IP Default Route** default route should be designated in the routing table for all traffic that cannot be directed to other specific routes.
You will need to define the default route to a remote router or, if required due to special circumstances, define an Ethernet gateway. There can be only one default route specified.

FRF8 Link Protocol

The FRF8 Link Protocol is an encapsulation method, which allows an ATM router to interoperate with a Frame Relay network.

FRF8 is only used in conjunction with the IP Network Protocol. Collect the information described below. This data will be later used to configure your router using the Command Line Interface (see *Configuration Tables*, Chapter 3).

IP Routing Network Protocol

♦ VPI and VCI Numbers

Your router may have been preconfigured with VPI/VCI numbers. If not, you will have to obtain these numbers from your Network Service Provider and then configure them.

If you are connecting to multiple remote sites, you will need to obtain additional VPI and VCI numbers from your Network Service Provider and/or Network Access Provider. These numbers identify the remote destination and must, therefore, be unique for each remote.

♦ DNS Internet Account Information (optional)

This information is obtained from the Network Service Provider.

Consult with your Network Service Provider to find out if you need to enter the following information:

- DNS Server Address
- DNS Second Server Address
- DNS Domain Name

Note: If you intend to connect to the Internet only, enter this information using the Internet Quick Start configurator.

♦ IP Routing Entries

For the Ethernet Interface

This information is defined by the user or the Network Administrator.

Ethernet IP Address (Local LAN)
An Ethernet LAN IP address and subnet mask are required for the router's local Ethernet LAN connection.

TCP/IP Ethernet Routes
You normally do not need to define an Ethernet IP route. An Ethernet IP route consists of an IP address, a mask, a metric, and a gateway. An Ethernet route is usually defined when there are multiple routers on the Ethernet which cannot exchange routing information between them.

For the ATM WAN Interface

This information is obtained from the Network Administrator or the NSP.

Source (Target/Local) WAN Port Address and Mask
You must specify a Source WAN IP address for the WAN connection to the remote router (whether or not NAT is enabled). The Source WAN Address is the address of the local router on the remote network. The mask is the mask used on the remote network. Check with your system administrator for details.

TCP/IP Remote Routes
When using FRF8, the IP route includes an IP address, subnet mask, metric (a number representing the perceived cost in reaching the remote network or station. Check with your system administrator for details.

A **TCP/IP Default Route** default route should be designated in the routing table for all traffic that cannot be directed to other specific routes.
You will need to define the default route to a remote router or, if required due to special circumstances, define an Ethernet gateway. There can be only one default route specified.

Dual Ethernet Router Configuration

General Information

To configure the Dual Ethernet router, access the router using the Command Line Interface (CLI). The CLI can be accessed from a Telnet or a console session (using the console cable) connected to the router's default IP address of 192.169.254.254. You can also configure the router using the Web browser GUI. Refer to the *Dual Ethernet Router Quick Start Guide*.

The Dual Ethernet router has two interfaces:

ETH/0 - refers to the router's hub with four 10Base-T connectors

ETH/1 - Refers to the single 10Base-T connector

Bridging is enabled by default when the router boots up. IP and IPX routing are disabled.

The router's default IP address is 192.168.254.254

DHCP is enabled by default and the router's DHCP server issues IP addresses to any PC request. The DHCP default IP pool is 192.168.254. 2 through 192.168.254.20.

To connect to the router, use the router's default IP address using a Telnet session, for example, and any 10Base-T port on the router.

Warning: You cannot boot from the ETH/1 interface.

Configuring the Dual Ethernet Router as a Bridge

This router is configured by default as a bridge and no configuration steps are needed. The user needs only establish a connection to the remote location (the ISP, for example).

Bridging is enabled by default when the router boots up. IP and IPX routing are disabled.

Configuring the Dual Ethernet Router for IP Routing

The ETH commands are used to configure the Dual Ethernet router for IP Routing. Please refer to the *Dual Ethernet Router Commands* section, page 169, for usage and syntax information.

The last argument of each ETH command determines which interface is being configured (0 for ETH/0, 1 for ETH/1).

Each interface (ETH/0 and ETH/1) must be set. A minimum of one route must be defined to have a working configuration. This is generally a default route on the ETH/1 interface where all traffic otherwise specified is automatically forwarded. This default route is: 0.0.0.0 255.255.255.255 1.

The Gateway address is the IP address supplied by the your ISP or Network Administrator.

You can customize your router by using the scripting feature which loads batch files of preset configuration commands into the router (refer to page page 220, *Batch File Command Execution*).

A Dual Ethernet Router sample configuration with IP Routing is provided on page 71.

Chapter 3. Configuring Router Software

This chapter covers the following configuration topics:

Configuration Tables

Configuration commands are outlined for each **Link Protocol/Network Protocol** supported by the router. The information needed to configure the router is contingent on the chosen Link Protocol. It is therefore important to know which Link Protocol you are using (this is determined by your Network Service Provider) to be able to refer to the configuration sections that apply to your setup.

A configuration table for the **Dual Ethernet Router** (with IP Routing enabled) is also provided.

Verify the Router Configuration

This section describes how to test IP, IPX, or Bridging.

Sample Configurations

This chapter provides two configuration examples with diagrams, commands, and list outputs.

Notes

1 - For usage conventions and a complete description of the commands mentioned in this chapter, refer to Chapter 5. *Command Line Interface Reference*.

2 - Command Line Interface
To configure the router software, the Command Line Interface is available to you at all times after you have installed the router hardware, connected a PC with a terminal emulation session (or ASCII terminal), and powered the unit on. This chapter assumes that you have successfully installed the router hardware as described in the User Guide.

If you intend to use the Command Line Interface through Configuration Manager, it is assumed that you have installed the Configuration Manager software and can access the terminal window (refer to the *User Guide*, chapter 1, *Installing and Accessing Configuration Manager*).

3 - Network Information Worksheets
Worksheets are provided in Appendix A so that you can enter details about your target router and remote routers. The worksheets list the commands associated with setting the features.

To configure the **target router**, you need to fill out one Target Router chart for the target router and one Remote Router chart for each remote router to be entered into the remote router database.

If you are setting up both ends of the network, you will need a mirror image of the information listed below for configuring the router on the other end of the link.

Important — Changing any the of the following settings requires a “reboot and save” of the router to take effect:

Ethernet LAN: Ethernet IP or IPX Address, TCP/IP Routing, IPX Routing

Bridging: Bridging, Filters

Remote Router: TCP/IP Route Addresses, IPX Routes, IPX SAPs and Bridging control, enable, disable or add remote routers

Configuration Tables

The following tables give you step-by-step instructions for standard configurations of the following Network Protocols / Link Protocol associations, as well as a configuration table for a Dual Ethernet Router:

- PPP Link Protocol with IP Routing Network Protocol
- PPP Link Protocol with IPX Routing Network Protocol
- PPP Link Protocol with Bridging Network Protocol
- RFC 1483/RFC 1490 Link Protocols with IP Routing Network Protocol
- RFC 1483/RFC 1490 Link Protocols with IPX Routing Network Protocol
- RFC 1483/RFC 1490 Link Protocols with Bridging Network Protocol
- RFC 1483MER/RFC 1490MER Link Protocols with IP Network Protocol
- FRF8 Link Protocol with IP Routing Network Protocol
- Mixed Network Protocols (Combinations of two or three Network protocols)
- Dual Ethernet Router with IP Routing

Note: Blank Network Configuration Worksheets are available in *Appendix A*.

How to use the tables

1. Find the configuration table that fits your particular Network Protocol/Link Protocol association. These tables are designed to provide easy step-by-step instructions.
2. Use the blank Network Configuration Worksheets provided at the end of the chapter to enter the commands in the order that they are given in the Configuration tables' **COMMANDS** column.
3. You may want to refer to the sample configurations at the end of the chapter.

Configuring PPP with IP Routing

This table outlines configuration commands for the PPP Link Protocol with the IP Routing Network Protocol.

PPP with IP Routing		
STEPS	**SETTINGS**	**COMMANDS**
System Settings		
System Name	Required	**system name** <*name*>
System Message	Optional	**system msg** <*message*>
Authentication Password	Required	**system passwd** <*password*>
Ethernet IP Address	As required	**eth ip addr** <*ipaddr*> <*ipnetmask*> [<*port#*>]
DHCP Settings	Already enabled; additional settings may be required	**dhcp set valueoption domainname** <*domainname*> **dhcp set valueoption domainnameserver** <*ipaddr*>
Change Login	Optional	**system admin** <*password*>
Remote Routers		
New Entry	Enter: Remote Name	**remote add** <*remoteName*>
Link Protocol/PVC[a] (for ATM routers)	Select: PPP Enter: VPI/VCI numbers	**remote setProtocol PPP** <*remoteName*> **remote setPVC** <*vpi number*>*****<*vci number*> <*remoteName*>
Link Protocol/DLCI[b] (for Frame Relay routers)	Select: PPP Enter: DLCI number	**remote setProtocol PPP** <*remoteName*> **remote setDLCI** <*number*> <*remoteName*>
Security [c] Remote's Password	Choose security level Enter: password	**remote setAuthen** <*protocol*> <*remoteName*> **remote setOurPasswd** <*password*> <*remoteName*>
Bridging On/Off	Must be OFF	**remote disBridge** <*remoteName*>
TCP/IP Route Address	Enter: Explicit or default route	**remote addIproute** <*ipnet*> <*ipnetmask*> <*hop*s> <*remoteName*>
If NAT is enabled:	To enable NAT, use:	**remote setIpTranslate** on <*remoteName*>
	Enter: You may need to enter a Source WAN Port Address	**remote setSrcIpAddr** <*ipaddr*> <*mask*> <*remoteName*>
If NAT is not enabled:	Enter: You may need to enter a Source WAN Port Address	**remote setSrcIpAddr** <*ipaddr*> <*mask*> <*remoteName*>
IP and IPX Routing		
TCP/IP Routing (Internet Firewall)	Must be enabled (Optional)	**eth ip enable** **eth firewall** <on | off>
IPX Routing	Must be disabled	**eth ipx disable**
Store **Reboot**		**save** **reboot**

a Enter this information if you are using PPP in an ATM environment.
b Enter this information if you are using PPP in a Frame Relay environment.
c If the ISP does not support the authentication of the ISP system by the caller, use the command **remote disauthen** <*remoteName*> to disable the authentication.

Configuring PPP with IPX Routing

This table outlines configuration commands for the PPP Link Protocol with the IPX Routing Network Protocol.

Note: Appendix B provides step-by-step information on how to configure IPX routing.

PPP with IPX Routing		
STEPS	**SETTINGS**	**COMMANDS**
System Settings		
System Name	Required	**system name** <*name*>
System Message	Optional	**system msg** <*message*>
Authentication Password	Required	**system passwd** <*password*>
Ethernet IP Address	As required	**eth ip addr** <*ipaddr*> <*ipnetmask*>[<*port#*>]
Settings DHCP	Already enabled; addit. settings may be required	**dhcp set valueoption domainname** <*domainname*> **dhcp set valueoption domainnameserver** < *ipaddr* >
Change Login	Optional	**system admin** <*password*>
Ethernet IPX Network #	Enter: IPX Network # Frame Type (default: 802.2)	**eth ipx addr** <*ipxnet*> [<*port#*>] **eth ipx frame** <*type*>
Remote Routers		
New Entry	Enter: Remote Name	**remote add** <*remoteName*>
Link Protocol/PVC[a] (for ATM routers)	Select: PPP Enter: VPI/VCI numbers	**remote setProtocol PPP** <*remoteName*> **remote setPVC** <*vpi number*>*****<*vci number*> <*remoteName*>
Link Protocol/DLCI[b] (for Frame Relay routers)	Select: PPP Enter: DLCI number	**remote setProtocol** PPP <*remoteName*> **remote setDLCI** <*number*> <*remoteName*>
Security[c] Remote's Password	Choose security level Enter: Password	**remote setAuthen** <*protocol*> <*remoteName*> **remote setPasswd** <*password*> <*remoteName*>
Bridging On/Off	Must be OFF	**remote disBridge** <*remoteName*>
IPX Routes Add	Enter appropriate info	**remote addIpxroute** <*ipxNet*> <*metric*> <*ticks*> <*remoteName*>
IPX SAPs Add	Enter appropriate info	**remote addIpxsap** <*servicename*> <*ipxNet*> <*ipxNode*> <*socket*> <*type*> <*hops*> <*remoteName*>
WAN Network #	Enter appropriate info	**remote setIpxaddr** <*ipxNet*> <*remoteName*>
IP and IPX Routing		
TCP/IP Routing	Must be disabled	**eth ip disable**
IPX Routing	Must be enabled	**eth ipx enable**
Store **Reboot**		**save** **reboot**

a Enter this information if you are using PPP in an ATM environment.

b Enter this information if you are using PPP in a Frame Relay environment.

c If the ISP does not support the authentication of the ISP system by the caller, use the command:
remote disauthen <*remoteName*> to disable the authentication.

Configuring PPP with Bridging

This table outlines configuration commands for the PPP Link Protocol with the Bridging Network Protocol.

PPP with Bridging		
STEPS	**SETTINGS**	**COMMANDS**
System Settings		
System Name	Required	**system name** <*name*>
System Message	Optional	**system msg** <*message*>
Authorization Password	Required	**system passwd** <*password*>
DHCP Settings	Already enabled; additional settings may be required	**dhcp set valueoption domainname** <*domainname*> **dhcp set valueoption domainnameserver** < *ipaddr* >
Change Login	Optional	**system admin** <*password*>
Remote Routers		
New Entry	Enter: Remote Name	**remote add** <*remoteName*>
Link Protocol/PVC[a] (for ATM routers)	Select: PPP Enter: VPI/VCI	**remote setProtocol PPP** <*remoteName*> **remote setPVC** <*vpi number*>*****<*vci number*> <*remoteName*>
Link Protocol/DLCI[b] (for Frame Relay routers)	Select: PPP Enter: DLCI number	**remote setProtocol PPP** <*remoteName*> **remote setDLCI** <*number*> <*remoteName*>
Security[c] Remote's Password	Choose security level Enter: Password	**remote setAuthen** <*protocol*> <*remoteName*> **remote setOurPasswd** <*password*> <*remoteName*>
Bridging On/Off	Must be ON	**remote enaBridge** <*remoteName*>
IP and IPX Routing		
IP Routing	Must be disabled	**eth ip disable**
IPX Routing	Must be enabled	**eth ipx disable**
Store **Reboot**		**save** **reboot**

a Enter this information if you are using PPP in an ATM environment.

b Enter this information if you are using PPP in a Frame Relay environment.

c If the ISP does not support the authentication of the ISP system by the caller, use the command **remote disauthen** <*remoteName*> to disable the authentication.

Configuring RFC 1483 / RFC 1490 with IP Routing

This table outlines configuration commands for the RFC 1483 and the RFC 1490 Link Protocols with the IP Routing Network Protocol.

RFC 1483 / RFC 1490 with IP Routing		
STEPS	**SETTINGS**	**COMMANDS**
System Settings		
System Message	Optional	**system msg** <*message*>
Ethernet IP Address	As required	**eth ip addr** <*ipaddr*> <*ipnetmask*> [<*port#*>]
DHCP Settings	Already enabled; additional settings may be required	**dhcp set valueoption domainname** <*domainname*> **dhcp set valueoption domainnameserver** <*ipaddr*>
Change Login	Optional	**system admin** <*password*>
Remote Routers		
New Entry	Enter: Remote Name	**remote add** <*remoteName*>
Link Protocol/PVC[a] (for ATM routers)	Select: RFC 1483 Enter: VPI/VCI Numbers	**remote setProtocol RFC1483** <*remoteName*> **remote setPVC** <*vpi number*>*****<*vci number*> <*remoteName*>
Link Protocol/DLCI[b] (for Frame Relay routers)	Select: FR Enter: DLCI number	**remote setProtocol FR** <*remoteName*> **remote setDLCI** <*number*> <*remoteName*>
Bridging On/Off	Must be OFF	**remote disBridge** <*remoteName*>
TCP/IP Route Address	Enter: Explicit or default route with remote gateway	**remote addiproute** <*ipnet*> <*ipnetmask*> <*hops*> <*remoteName*>
If Address Translation (NAT) is enabled: TCP/IP Route Addresses	To enable NAT, use: Enter: Source WAN Port Address	**remote setIpTranslate on** <*remoteName*> **remote setSrcIpAddr** <*ipaddr*> <*mask*> <*remoteName*>
If NAT is OFF: TCP/IP Route Addresses	You may still need to enter a Source WAN Port Address	**remote setSrcIpAddr** <*ipaddr*> <*mask*> <*remoteName*>
IP and IPX Routing		
TCP/IP Routing (Internet Firewall)	Must be enabled (Optional)	**eth ip enable** **eth firewall** <on | off >
IPX Routing	Must be disabled	**eth ipx disable**
Store **Reboot**		**save** **reboot**

a Enter this information if you are using RFC 1483 in an ATM environment.

b Enter this information if you are using RFC 1490 in a Frame Relay environment.

Configuring RFC 1483 / RFC 1490 with IPX Routing

This table outlines configuration commands for the RFC 1483 and RFC 1490 Link Protocols with the IPX Routing Network Protocol.

Note: Appendix B provides step-by-step information on how to configure IPX routing.

RFC 1483 / RFC 1490 with IPX Routing		
STEPS	**SETTINGS**	**COMMANDS**
System Settings		
System Message	Optional	**system msg** <*message*>
Ethernet IP Address	As required	**eth ip addr** <*ipaddr*> <*ipnetmask*> [<*port#*>]
DHCP Settings	Already enabled; additional settings may be required	**dhcp set valueoption domainname** <*domainname*> **dhcp set valueoption domainnameserver** < *ipaddr* >
Ethernet IPX Network #	Enter: IPX Network # Frame Type (default is 802.2)	**eth ipx addr** <*ipxnet*> [<*port#*>] **eth ipx frame** <*type*>
Change Login	Optional	**system admin** <*password*>
Remote Routers		
New Entry	Enter: Remote Name	**remote add** <*remoteName*>
Link Protocol/PVC (for ATM routers)	Select: RFC 1483 Enter: VPI/VCI Numbers	**remote setProtocol RFC1483** <*remoteName*> **remote setPVC** <*vpi number*>***** <*vci number*> <*remoteName*>
Link Protocol/DLCI[a] (for Frame Relay routers)	Select: FR Enter: DLCI number	**remote setProtocol FR** <*remoteName*> **remote setDLCI** < *number*> <*remoteName*>
Bridging On/Off	Must be OFF	**remote disBridge** <*remoteName*>
IPX Routes Add	Enter appropriate info	**remote addIpxroute** <*ipxNet*> <*metric*> <*ticks*> <*remoteName*>
IPX SAPs Add	Enter appropriate info	**remote addIpxsap** <*servicename*> <*ipxNet*> < *ipxNode*> <*socket*> <*type*> <*hops*> <*remoteName*>
WAN Network Number	Enter appropriate info	**remote setIpxaddr** <*ipxNet*> <*remoteName*>
IP and IPX Routing		
TCP/IP Routing (Internet Firewall)	Must be disabled (Optional)	**eth ip disable** **eth firewall** <on | off >
IPX Routing	Must be enabled	**eth ipx enable**
Store **Reboot**		**save** **reboot**

a Enter this information if you are using RFC 1490 in a Frame Relay environment.

Configuring RFC 1483 / RFC 1490 with Bridging

This table outlines configuration commands for the RFC 1483 and RFC 1490 Link Protocols with the Bridging Network Protocol.

RFC 1483 / RFC 1490 with Bridging		
STEPS	**SETTINGS**	**COMMANDS**
System Settings		
System Message	Optional	**system msg** <*message*>
DHCP Settings	Already enabled; additional settings may be required	**dhcp set valueoption domainname** <*domainname*> **dhcp set valueoption domainnameserver** <*ipaddr*>
Change Login	Optional	**system admin** <*password*>
Remote Routers		
New Entry	Enter: Remote Name	**remote add** <*remoteName*>
Link Protocol/PVC (for ATM routers)	Select: RFC 1483 Enter: VPI/VCI Numbers	**remote setProtocol RFC1483** <*remoteName*> **remote setPVC** <*vpi number*>*****<*vci number*> <*remoteName*>
Link Protocol/DLCI[a] (for Frame Relay routers)	Select: FR Enter: DLCI number	**remote setProtocol FR** <*remoteName*> **remote setDLCI** <*number*> <*remoteName*>
Bridging On/Off	Must be ON	**remote enaBridge** <*remoteName*>
IP and IPX Routing		
IP Routing	Must be disabled	**eth ip disable**
IPX Routing	Must be disabled	**eth ipx disable**
Store **Reboot**		**save** **reboot**

a Enter this information if you are using RFC 1490 in a Frame Relay environment.

Configuring MAC Encapsulated Routing: RFC 1483MER / RFC 1490MER with IP Routing

This table outlines configuration commands for the RFC 1483MER and RFC 1490MER Link Protocols with the IP Routing Network Protocol.

RFC 1483MER / RFC 1490 MER with IP Routing		
STEPS	**SETTINGS**	**COMMANDS**
System Settings		
System Message	Optional	**system msg** <message>
Ethernet IP Address	As required	**eth ip addr** <*ipnet*> <*ipnetmask*> [<*port#*>]
DHCP Settings	Already enabled; additional settings may be required	**dhcp set valueoption domainname** <*domainname*> **dhcp set valueoption domainnameserver** <*ipaddr*>
Change Login	Optional	**system admin** <*password*>
Remote Routers		
New Entry	Enter: Remote Name	**remote add** <*remoteName*>
Link Protocol/PVC[a] (for ATM routers)	Select: RFC 1483MER Enter: VPI/VCI Numbers	**remote setProtocol** RFC1483MER <*remoteName*> **remote setPVC** <*vpi number*>*****<*vci number*> <*remoteName*>
Link Protocol/DLCI[b] (for Frame Relay routers)	Select: MER Enter: DLCI number	**remote setProtocol MER** <*remoteName*> **remote setDLCI** <*number*> <*remoteName*>
Bridging On/Off	Must be OFF	**remote disBridge** <*remoteName*>
TCP/IP Route Address	Enter: Explicit or default route with remote gateway	**remote addiproute** <*ipnet*> <*ipnetmask*><*ipGateway*> <*ipGateway*> <*remoteName*>
If NAT is enabled:	To enable NAT, use:	**remote setIpTranslate on** <*remoteName*>
If NAT is OFF: TCP/IP Route Addresses	Enter: Source WAN Port Address + mask of the remote network	**remote setSrcIpAddr** <*ipaddr*> <*mask*><*remoteName*>
	Enter a Source WAN Port Address + mask of the remote network's mask	**remote setSrcIpAddr** <*ipaddr*> <*mask*> <*remoteName*>
IP and IPX Routing		
TCP/IP Routing (Internet Firewall)	Must be enabled (Optional)	**eth ip enable** **eth firewall** <on | off >
IPX Routing	Must be disabled	**eth ipx disable**
Store **Reboot**		**save** **reboot**

a Enter this information if you are using RFC 1483 in an ATM environment.

b Enter this information if you are using RFC 1490 in a Frame Relay environment.

Configuring FRF8 with IP Routing

This table outlines configuration commands for the FRF8 Link Protocol with the IP Routing Network Protocol.

FRF8 with IP Routing		
STEPS	**SETTINGS**	**COMMANDS**
System Settings		
System Message	Optional	**system msg** <*message*>
Ethernet IP Address	As required	**eth ip addr** <*ipaddr*> <*ipnetmask*> [<*port#*>]
DHCP Settings	Already enabled; additional settings may be required	**dhcp set valueoption domainname** <*domainname*> **dhcp set valueoption domainnameserver** <*ipaddr*>
Change Login	Optional	**system admin** <*password*>
Remote Routers		
New Entry	Enter: Remote Name	**remote add** <*remoteName*>
Link Protocol/PVC	Select: FRF8 Enter: VPI/VCI Numbers	**remote setProtocol FRF8** <*remoteName*> **remote setPVC** <*vpi number*>*****<*vci number*> <*remoteName*>
Bridging On/Off	Must be OFF	
TCP/IP Route Address	Enter: Explicit or default route	**remote disBridge** <*remoteName*> **remote addIproute** <*ipnet*> <*ipnetmask*> <*hops*> <*remoteName*>
If Address Translation (NAT) is enabled:	To enable NAT, use:	**remote setIpTranslate on** <*remoteName*>
	Enter: Source WAN Port Address + mask of the remote network	**remote setSrcIpAddr** <*ipaddr*> <*mask*><*remoteName*>
If NAT is OFF: TCP/IP Route Addresses	Enter a Source WAN Port Address + mask of the remote network	**remote setSrcIpAddr** <*ipaddr*> <*mask*><*remoteName*>
IP and IPX Routing		
TCP/IP Routing (Internet Firewall)	Must be enabled (Optional)	**eth ip enable** **eth firewall** <on | off >
IPX Routing	Must be disabled	**eth ipx disable**
Store **Reboot**		**save** **reboot**

Configuring Mixed Network Protocols

Several Network Protocols can be configured concurrently in the same router. The possible combinations are:

- Bridging + IP Routing
- Bridging + IPX Routing
- Bridging + IP Routing + IPX Routing
- IP Routing + IPX Routing

General Configuration Rules:

IP (and IPX) Routing takes precedence over Bridging.

Each Network Protocol in the combination is individually configured as described in the preceding tables.

When configuring multiple network protocols, **make sure that they are all enabled** (even though the preceding individual configuration tables show them to be mutually exclusive).

Example:

To configure Bridging + IP Routing (both with Link Protocol RFC 1483), refer to the preceding *RFC 1483 with Bridging* and *RFC 1483 with IP Routing* tables. Follow the instructions described in the tables, except for the Bridging and IP Routing settings: Since you are configuring both Bridging and IP Routing, make sure that these two protocols are both enabled (even though the individual configuration tables you are referring to are showing them to be mutually exclusive).

Configure Bridging and then IP Routing.

Remember that IP Routing has precedence over Bridging.

Configuring a Dual Ethernet Router for IP Routing

This table outlines commands used to configure a Dual Ethernet router for IP Routing.

Dual Ethernet Router - IP Routing		
Steps	**Settings**	**Commands**
System Settings System Name	Optional	**system name** <*name*>
System Settings Message	Optional	**system msg <***message***>**
Ethernet Settings Routing/ Bridging Controls	Enable IP Routing Disable Bridging	**eth ip enable** **eth br disable**
Ethernet Settings ETH/0 IP Address ETH/1 IP Address TCP/IP default route address	Define ETH/0 IP address for the hub side Define ETH/1 IP address for the single 10Base-T side ETH/0 sends all traffic to ETH/1	**eth ip addr** <*ipaddr*> <*ipnetmask*> [<*port#*>] **eth ip addr** <*ipaddr*> <*ipnetmask*> [<*port#*>] **eth ip addroute** <*ipaddr*> <*ipnetmask*> <*gateway*> <*hops*> [<*port#*>]
DHCP Settings DHCP Settings	Already enabled; additional settings may be required Define DHCP network for ETH/1 Create an address pool for ETH/1 DNS Domain Name DNS Server WINS Server Address	**dhcp add | <***net***>** <*mask*> **|** <*ipaddr*> **|** <*code*> <*min*> <*max*> <*type*> **dhcp set addresses** <*first ipaddr*> <*last ipaddr*> **dhcp set valueoption domainname <***domainname***>** **dhcp set valueoption domainnameserver** <*ipaddr*> **dhcp set valueoption winsserver** <*ipaddr*>

Verify the Router Configuration

Test IP Routing

Test IP Routing over the Local Ethernet LAN (from PC)

- Use the TCP/IP **ping** command or a similar method to contact the configured target router specifying the Ethernet LAN IP address.
- If you cannot contact the router, verify that the Ethernet IP address and subnet mask are correct and check the cable connections.
- Make sure that you have saved and rebooted after setting the IP address.
- Check Network TCP/IP properties under Windows 95. If you are running Windows 3.1, check that you have a TCP/IP driver installed.

Test IP Routing to a Remote Destination

- Using the TCP/IP **ping** command, contact a remote router from a local LAN-connected PC. When you enter the **ping** command, the router will connect to the remote router using the DSL line.
- If remote or local WAN IP Addresses are required, verify that they are valid.
- Use the **iproutes** command to check, first, the contents of the IP routing table and, second, that you have specified a default route as well.

Test Routing from a Remote Destination

- Have a remote router contact the target router using a similar method.

Test TCP/IP Routes

- Contact a station, subnetwork or host located on the network beyond a remote router to verify the TCP/IP route addresses entered in the remote router database.
- Verify that you configured correct static IP routes.
- Use the **iproutes** command to check the contents of the IP routing table.

Test Bridging to a Remote Destination

Use any application from a local LAN-attached station that accesses a server or disk using a protocol that is being bridged on the remote network beyond the remote router. If you cannot access the server:

- Verify that you have specified a default destination remote router.
- Make sure that you have enabled bridging to the remote router.
- Check that bridging filtering does not restrict access from the local station.

Test IPX Routing

One way to test IPX Routing is to check for access to servers on the remote LAN. Under Windows, use the "NetWare Connections" selection provided with NetWare User Tools. Under DOS, use the command **pconsole** or type **login** on the login drive (usually F:)**.** Select the printer server and verify that the server you have defined is listed. When you attempt to access the server, the router will connect to the remote router using the DSL line.

If you cannot access the remote server:

- Check that the local Ethernet LAN IPX network number is correct.
- Verify that the WAN link network number is the same as the remote WAN link network number.
- Check cable connections and pinouts.
- Verify that the IPX Routes and IPX SAPs you have specified are correct.
- List the contents of the routing and services tables using the **ipxroutes** and **ipxsaps** commands, respectively.
- Make sure that the security authentication method and password that you configured matches the remote router.

Sample Configurations

Sample Configuration 1 — PPP with IP and IPX

This configuration example comprises:

- A scenario describing the configuration
- A diagram showing the configuration of the SOHO router
- Tables containing the configuration settings for this example
- Several “list” commands outputs that are used to check the information entered for this particular configuration.
- Information about the Names and Passwords that are used in this configuration example (required for PPP)

Note: Blank Network Information Worksheets are available to fill in the information for your own configuration in *Appendix A.*

Scenario

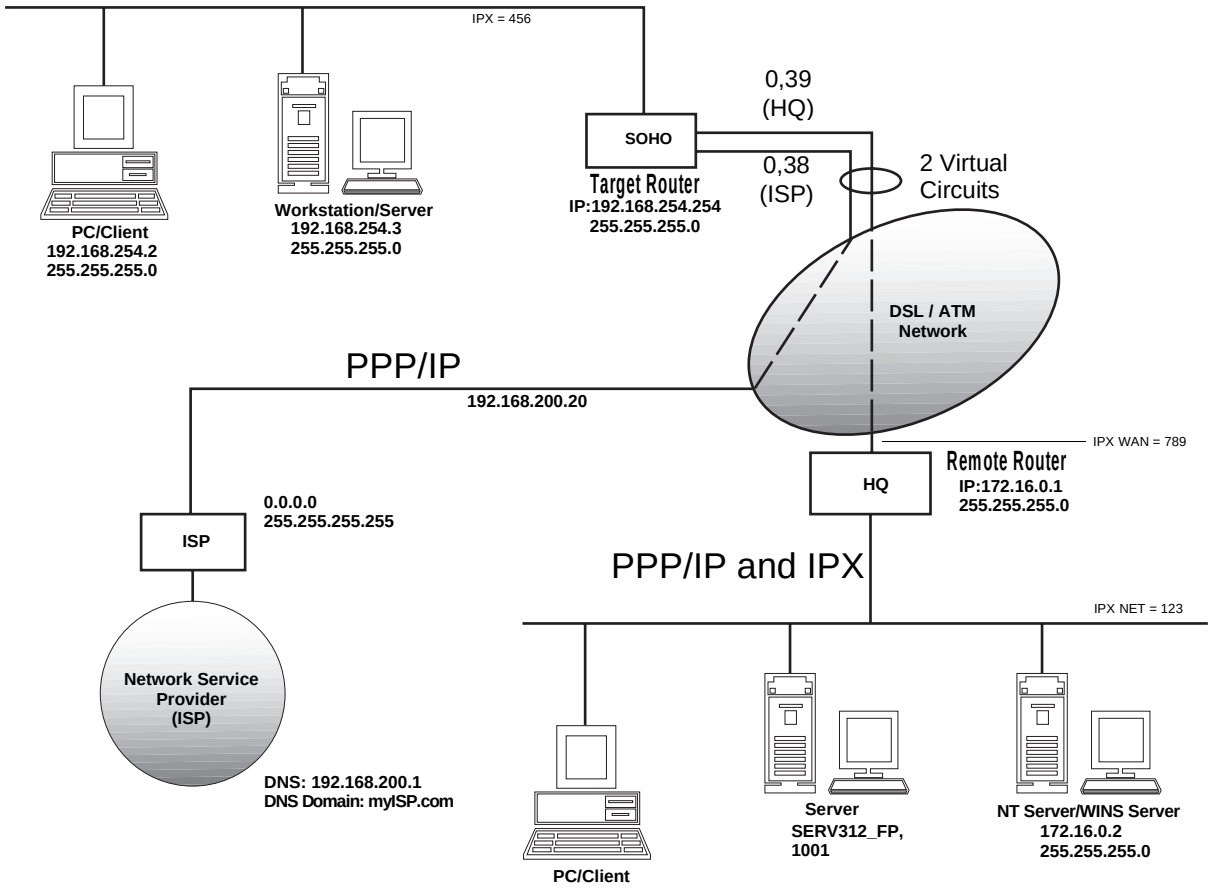
In this configuration example of a hypothetical network, a small office/home office **(SOHO)** will access:

- The Internet through an Internet Service Provider **(ISP);** it uses PPP as the Link Protocol with IP Routing as the Network Protocol. Network Address Translation (NAT) is enabled to the ISP, since the ISP assigned the SOHO only one IP address.
- A central site (HQ) through a Network Service Provider (NSP provides access to the DSL/ATM Wide Area Network); it uses PPP as the Link Protocol with IP and IPX as its Network Protocols.

  IP addresses are issued by the DHCP server. DHCP will be set up to issue DNS information to the SOHO LAN.

Sample Configuration 1 — Diagram for Target Router (SOHO)

Small Home Office SOHO (Target/Local Router)



Network Service Provider
(HQ)

Sample Configuration 1 — Tables For Target Router (SOHO)

SOHO SYSTEM SETTINGS		
Configuration Section	**Item**	**Commands**
System Settings Name	System Name	**system name SOHO**
System Settings Message	Message (optional)	**system msg Configured_Dec_1998**
System Settings Authentication Password	Authentication Password	**system password SOHOpasswd**
System Settings Ethernet IP Address Ethernet IPX Network	Ethernet IP Address and Subnet Mask (default IP address) Ethernet IPX Network Number	**eth ip addr 192.168.254.254 255.255.255.0** **eth ipx addr 456**
System Settings DHCP Settings	DNS Domain Name DNS Server WINS Server Address	**dhcp set valueoption domainname myISP.com** **dhcp set valueoption domainnameserver 192.168.200.1** **dhcp set valueoption winsserver 172.16.0.2**

SOHO REMOTE ROUTER DATABASE ENTRY: HQ		
Configuration Section	**Item**	**Commands**
Remote Routers New Entry	Remote Router's Name	**remote add HQ**
Remote Routers Link Protocol PVC	Link Protocol VPI Number/VCI Number	**remote setProtocol PPP HQ** **remote setPVC 0*39 HQ**
Remote Routers Security	Minimum Authentication (PAP is the default) Remote Router's Password	**remote setauthen PAP HQ** **remote setpasswd HQpasswd HQ**
Remote Routers Bridging	Bridging On/Off **(**Bridging is OFF by default)	**remote disbridge HQ**
Remote Routers TCP/IP Route Addresses	Remote Network's IP Addresses, Subnet Masks, and Metric	**remote addiproute 172.16.0.0 255.255.255.0 1 HQ**
Remote Routers IPX Address	Network #, Hop Count, Ticks	**remote addipxroute 1001 1 4 HQ**
Remote Routers IPX SAPs	SAPS: Server Name, Server Type, Network #, Node #, Sockets, type, hops WAN Network #	**remote addipxsap SERV312_FP 4 1001 00-00-00-00-00-01 451 3 1 HQ** **remote setipxaddr 789 HQ**

Note: Fill in one worksheet for each remote router in the Remote Router Database

SOHO REMOTE ROUTER DATABASE ENTRY: ISP		
Configuration Section	**Item**	**Commands**
Remote Routers New Entry	Remote Router's Name	**remote add ISP**
Remote Routers Link Protocol PVC	Link Protocol VPI Number/VCI Number	**remote setProtocol PPP ISP** **remote setPVC 0*38 ISP**
Remote Routers Security	Minimum Authentication (PAP is the default) Remote Router's Password	**remote setauthen PAP ISP** **remote setpasswd ISPpasswd ISP**
Remote Routers Bridging	Bridging On/Off **(**Bridging is OFF by default)	**remote disbridge ISP**
Remote Routers TCP/IP Route Addresses	Remote Network's IP Addresses, Subnet Masks, and Metric Network Address Translation In Advanced: Source WAN IP Address and Subnet Mask[a]	**remote addiproute 0.0.0.0 255.255.255.255 1 ISP (Default Route)** **remote setiptranslate on ISP** **remote setsrcipaddr 192.168.200.20 255.255.255 255 ISP**

a This is needed only if the ISP does not assign an IP address automatically.
Note: Fill in one worksheet for each remote router in the remote router database

SOHO ROUTING CONTROLS		
Configuration Section	**Item**	**Commands**
IP and IPX Routing	TCP/IP Routing On/Off IPX Routing On/Off Internet Firewall On/Off (Firewall is ON by default)	**eth ip enable** **eth ipx enable** **eth ip firewall on**

Sample Configuration 1 - Check the Configuration with the “list” Commands

Type the following **commands** to obtain a list of your configuration.

system list

```
GENERAL INFORMATION FOR <SOHO>
  System started on..................... 12/1/1998 at 17:41
  Authentication override.............. NONE
 WAN to WAN Forwarding................ yes
  BOOTP/DHCP Server address............ none
  Telnet Port.......................... default (23)
SNMP Port............................ default (161)
  System message: configured Dec-1998
```

remote list

```
INFORMATION FOR <HQ>
  Status.............................. enabled
  Protocol in use...................... PPP
  Authentication....................... enabled
  Authentication level required........ PAP
  Connection Identifier (VPI*VCI)...... 0*39
  IP address translation............... off
  Compression Negotiation.............. off
  Source IP address/subnet mask........ 0.0.0.0/0.0.0.0
  Remote IP address/subnet mask........ 0.0.0.0/0.0.0.0
  Send IP RIP to this dest............. no
    Send IP default route if known..... no
  Receive IP RIP from this dest........ no
    Receive IP default route by RIP.... no
  Keep this IP destination private..... yes
  Total IP remote routes............... 1
           172.16.0.0/255.255.255.0/1
  IPX network number................... 00000789
  Total IPX remote routes.............. 1
           00001001/1/4
  Total IPX SAPs....................... 1
    SERV312_FP 00001001 00:00:00:00:00:01 0451 0003 1
  Bridging enabled..................... no
    Exchange spanning tree with dest... yes

INFORMATION FOR <ISP>
  Status.............................. enabled
  Protocol in use...................... PPP
  Authentication....................... enabled
  Authentication level required........ PAP
  Connection Identifier (VPI*VCI)...... 0*38
  IP address translation............... on
  Compression Negotiation.............. off
  Source IP address/subnet mask........ 192.168.200.20/255.255.255.255
  Remote IP address/subnet mask........ 0.0.0.0/0.0.0.0
  Send IP RIP to this dest............. no
    Send IP default route if known..... no
  Receive IP RIP from this dest........ no
    Receive IP default route by RIP.... no
  Keep this IP destination private..... yes
```

```
  Total IP remote routes..... 1
           0.0.0.0/255.255.255.255/1
  IPX network number..... 00000000
  Total IPX remote routes..... 0
  Total IPX SAPs..... 0
  Bridging enabled..... no
    Exchange spanning tree with dest... yes
```

dhcp list

```
       bootp server ..... none
       bootp file ..... n/a

       DOMAINNAMESERVER (6) ..... 192.168.200.1
       DOMAINNAME (15) ..... myISP.com
       WINSSERVER (44) ..... 172.16.0.2

Subnet 192.168.254.0, disabled - other DHCP servers detected
       When DHCP servers are active . stop
       Mask ..... 255.255.255.0
       first ip address ..... 192.168.254.2
       last ip address ..... 192.168.254.20
       lease ..... default
       bootp ..... not allowed
       bootp server ..... none
       bootp file ..... n/a
```

eth list

```
ETHERNET INFORMATION FOR <ETHERNET/0>
  Hardware MAC address..... 00:20:6F:02:A1:BF
  Bridging enabled..... no
  IP Routing enabled..... yes
    Firewall filter enabled ..... yes
    Send IP RIP to the LAN..... rip-1 compatible
      Advertise me as default router... yes
    Process IP RIP packets received.... rip-1 compatible
      Receive default route by RIP..... yes
  RIP Multicast address..... default
  IP address/subnet mask..... 192.168.254.254/255.255.255.0
  IP static default gateway..... none
  IPX Routing enabled..... yes
    External network number..... 00000456
    Frame type..... 802.2
```

Information About Names And Passwords

In this configuration example, the PPP Link Protocol requires using systems names and passwords.

♦ System Passwords

SOHO has a system password "SOHOpasswd" This password is used when SOHO communicates with HQ for authentication by that site, and at any time when HQ challenges SOHO.

HQ has a system password "HQpasswd" which is, likewise, used when HQ communicates with site SOHO for authentication by SOHO, and at any time SOHO challenges HQ.
ISP has a system password "ISPpasswd" used for the same purpose.

♦ Remote Passwords

Each router has a remote router's password for each remote router defined in its Remote Router Database. The router will use the remote password to authenticate the remote router when the remote router communicates with or is challenged by the local site.

For example, SOHO has remote router entries for HQ and ISP, and defined in each table entry are the respective remote router's password.

The following table shows the names and passwords for each router that must be defined for authentication to be performed correctly. (This assumes that all three systems use some form of authentication protocol.)

Note: If you have trouble with passwords, we recommend that you set the remote router security to "**disable authentication**" to simplify the process.

	Names & passwords configured in SOHO Router	**Names & passwords configured in HQ Router**	**Names & passwords configured in ISP Router**
System Name	SOHO	HQ	ISP
System Password	SOHOpasswd	HQpasswd	ISPpasswd
Remote Router Database	HQpasswd ISPpasswd	SOHOpasswd	SOHOpasswd

Sample Configuration 2 — RFC 1483 with IP and Bridging

This configuration example comprises:

- A scenario describing this configuration of the router SOHO
- A diagram showing the configuration information needed for this example
- Tables containing the configuration settings for this example
- Several “list” commands outputs that are used to check the information entered for this particular configuration.

Note 1: Names and Passwords are not required with the RFC 1483 Link Protocol.

Note 2: Blank Network Information Worksheets are available to fill in the information for your own configuration in Appendix A.

Scenario

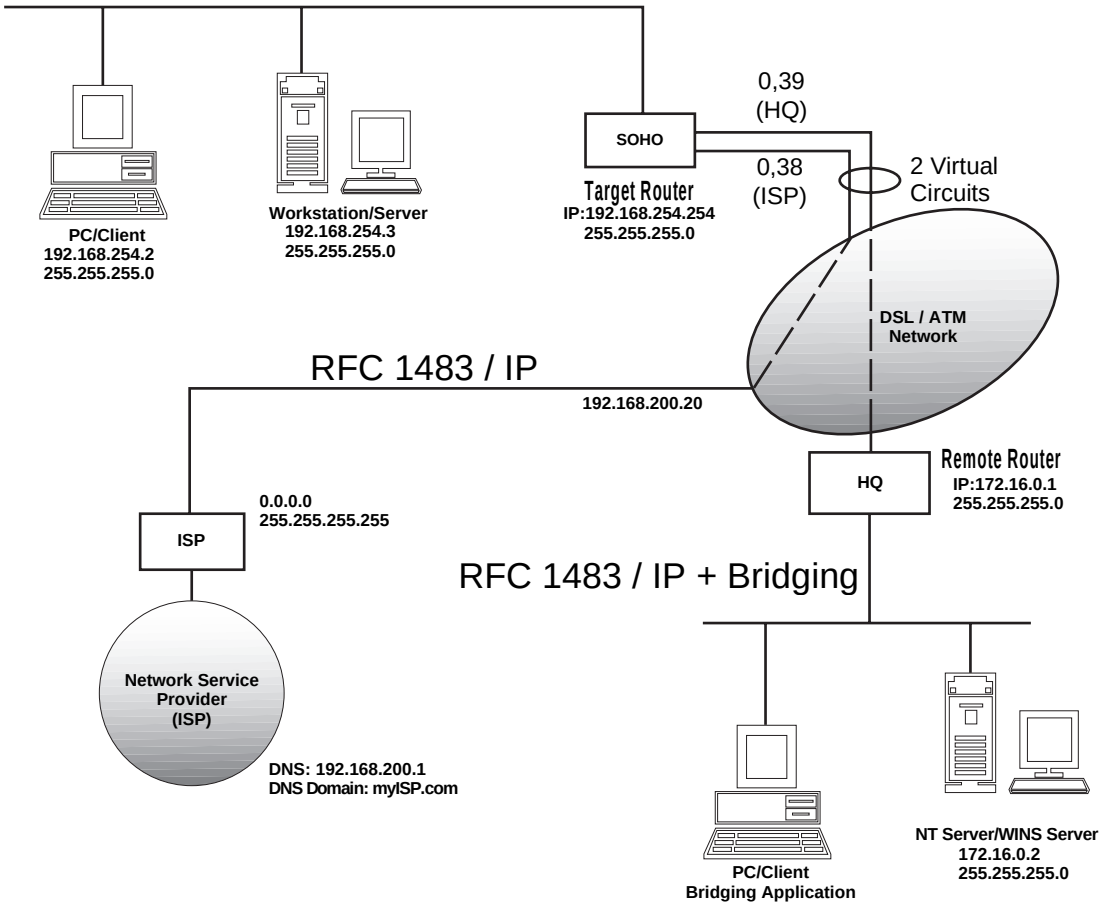
In this configuration example of a hypothetical network, a small office/home office **(SOHO)** will access:

- The Internet through an Internet Service Provider **(ISP);** it uses RFC 1483 as the Link Protocol with IP Routing as the Network Protocol. Network Address Translation (NAT) is enabled to the ISP, since the ISP assigned SOHO only one IP address.
- A central site (HQ) through a Network Service Provider (NSP provides access to the DSL/ATM Wide Area Network); it uses RFC 1483 as the Link Protocol with Bridging and IP Routing as its Network Protocols.

IP addresses are issued by the DHCP server. DHCP will be set up to issue DNS information to the SOHO LAN.

Sample Configuration 2 — Diagram for Target Router SOHO

Small Home Office SOHO (Target Router)



Network Service Provider (HQ)

Sample Configuration 2 — Tables For Target Router (SOHO)

SOHO SYSTEM SETTINGS		
Configuration Section	**Item**	**Commands**
System Settings Message	Message (optional)	**system msg RFC1483_dec98**
System Settings Ethernet IP Address	Ethernet IP Address and Subnet Mask (default IP address)	**eth ip addr 192.168.254.254 255.255.255.0**
System Settings DHCP Settings	DNS Domain Name DNS Server WINS Server address	**dhcp set valueoption domainname myISP.com** **dhcp set valueoption domainnameserver 192.168.200.1** **dhcp set valueoption winsserver 172.16.0.2**

SOHO REMOTE ROUTER DATABASE ENTRY: HQ		
Configuration Section	**Item**	**Commands**
Remote Routers New Entry	Remote Router's Name	**remote add HQ**
Remote Routers Link Protocol PVC	Link Protocol VPI Number/VCI Number	**remote setProtocol RFC1483 HQ** **remote setPVC 0*39 HQ**
Remote Routers Bridging	Bridging On/Off	**remote enabridge HQ**
Remote Routers TCP/IP Route Addresses	Remote Network's IP Addresses, Subnet Masks, and Metric	**remote addiproute 172.16.0.0 255.255.255.0 1 HQ**

SOHO REMOTE ROUTER DATABASE ENTRY: ISP		
Configuration Section	**Item**	**Commands**
Remote Routers New Entry	Remote Router's Name	**remote add ISP**
Remote Routers Link Protocol PVC	Link Protocol VPI Number/VCI Number	**remote setProtocol RFC1483 ISP** **remote setPVC 0*38 ISP**
Remote Routers Bridging	Bridging On/Off (Bridging is Off by default)	**remote disbridge ISP**
Remote Routers TCP/IP Route Addresses	Remote Network's IP Addresses, Subnet Masks, and Metric Network Address Translation (NAT) In Advanced: Source WAN IP Address and Subnet Mask	**remote addiproute 0.0.0.0 255.255.255.255 1 ISP** **remote setiptranslate on ISP** **remote setsrcipaddr 192.168.200.20 255.255.255.255 ISP**

SOHO ROUTING CONTROLS		
Configuration Section	**Item**	**Commands**
IP and IPX Routing	TCP/IP Routing On/Off IPX Routing On/Off (IPX routing is OFF by default) Internet Firewall On/Off (Firewall is ON by default)	**eth ip enable** **eth ipx disable** **eth ip firewall on**

Sample Configuration 2 - Check the Configuration with the “list” Commands

system list

```
GENERAL INFORMATION FOR <SOHO>
  System started on..................... 12/1/1998 at 17:48
  Authentication override.............. NONE
WAN to WAN Forwarding................. yes
  BOOTP/DHCP Server address............ none
  Telnet Port.......................... default (23)
SNMP Port............................ default (161)
  System message: ADSL RFC1483 sample
```

eth list

```
ETHERNET INFORMATION FOR <ETHERNET/0>
  Hardware MAC address................. 00:20:6F:02:A1:BF
  Bridging enabled..................... yes
  IP Routing enabled................... yes
    Firewall filter enabled ........... yes
    Send IP RIP to the LAN.............. rip-1 compatible
      Advertise me as default router... yes
    Process IP RIP packets received.... rip-1 compatible
      Receive default route by RIP..... yes
  RIP Multicast address................ default
  IP address/subnet mask............... 192.168.254.254/255.255.255.0
    IP static default gateway............ none
  IPX Routing enabled.................. no
    External network number............ 00000000
    Frame type......................... 802.2
```

remote list

```
INFORMATION FOR <HQ>
  Status............................... enabled
  Protocol in use...................... RFC1483 (SNAP)
  Connection Identifier (VPI*VCI)...... 0*39
  IP address translation............... off
  Compression Negotiation.............. off
  Source IP address/subnet mask........ 0.0.0.0/0.0.0.0
  Remote IP address/subnet mask........ 0.0.0.0/0.0.0.0
  Send IP RIP to this dest............. no
    Send IP default route if known..... no
  Receive IP RIP from this dest........ no
    Receive IP default route by RIP.... no
  Keep this IP destination private..... yes
  Total IP remote routes............... 1
           172.16.0.0/255.255.255.0/1
  IPX network number................... 00000000
  Total IPX remote routes.............. 0
  Total IPX SAPs....................... 0
  Bridging enabled..................... yes
    Exchange spanning tree with dest... yes

INFORMATION FOR <ISP>
  Status............................... enabled
  Protocol in use...................... RFC1483 (SNAP)
```

```
  Connection Identifier (VPI*VCI)...... 0*38
  IP address translation.............. on
  Compression Negotiation.............. off
  Source IP address/subnet mask........ 192.168.200.20/255.255.255.255
  Remote IP address/subnet mask........ 0.0.0.0/0.0.0.0
  Send IP RIP to this dest............. no
    Send IP default route if known..... no

Receive IP RIP from this dest........ no
    Receive IP default route by RIP.... no
  Keep this IP destination private..... yes
  Total IP remote routes............... 1
           0.0.0.0/255.255.255.255/1
  IPX network number................... 00000000
  Total IPX remote routes.............. 0
  Total IPX SAPs....................... 0
  Bridging enabled..................... no
    Exchange spanning tree with dest... yes
```

dhcp list

```
       bootp server ................. none
       bootp file ................... n/a

       DOMAINNAMESERVER (6) ......... 192.168.200.1
       DOMAINNAME (15) .............. myISP.com
       WINSSERVER (44) .............. 172.16.0.2

Subnet 192.168.254.0, disabled - other DHCP servers detected
       When DHCP servers are active . stop
       Mask ......................... 255.255.255.0
       first ip address ............. 192.168.254.2
       last ip address .............. 192.168.254.20
       lease ........................ default
       bootp ........................ not allowed
       bootp server ................. none
       bootp file ................... n/a
```

Sample Configuration 3 — Configuring a Dual Ethernet Router for IP Routing

Scenario

The following example provides a simple sample configuration for a Dual Ethernet router with IP Routing enabled.

The router's hub (ETH/0) belongs to the 192.168.254.0 subnet. The router's ETH/1 belongs to the 192.168.253.0 subnet.

ETH/0 will route packets to ETH/1 at the address 192.168.253.254. DHCP is enabled for both subnets.

ETH_ROUTER CONFIGURATION		
Configuration Section	**Item**	**Commands**
System Settings Name	System Name (optional)	**system name eth_router**
System Settings Message	Message (optional)	**system msg Configured_Jan_1999**
Ethernet Settings Routing/ Bridging Controls	Enable IP Routing Disable Bridging	**eth ip enable** **eth br disable**
Ethernet Settings ETH/0 IP Address ETH/1 IP Address TCP/IP default route address	Define ETH/0 IP address for the hub side Define ETH/1 IP address for the single 10Base-T side ETH/0 sends all traffic to ETH/1	**eth ip addr 192.168.254.254 255.255.255.0 0** **eth ip addr 192.168.253.254 255.255.255.0 1** **eth ip addroute 0.0.0.0 255.255.255.255 192.168.253.254 1 1**
DHCP Settings DHCP Settings	Define DHCP network for ETH/1 Create an address pool for ETH/1 DNS Domain Name DNS Server WINS Server Address	**dhcp add 192.168.253.0 255.255.255.0** **dhcp set addresses 192.168.253.2 192.168.253.20** **dhcp set valueoption domainname myISP.com** **dhcp set valueoption domainnameserver 192.168.200.1** **dhcp set valueoption winsserver 172.16.0.2**

Chapter 4. Configuring Special Features

The features described in this chapter are advanced topics. They are primarily intended for experienced users and network administrators to perform network management and more complex configurations.

- IP Firewall and Bridging Filtering
- IP (RIP) Protocol Controls
- DHCP (Dynamic Host Configuration Protocol)
- NAT (Network Address Translation)
- Management Security
- Software Options Keys
- Encryption
- IP filtering
- L2TP tunneling

Bridging Filtering and IP Firewall

General Information

You can control the flow of packets across the router using bridging filtering. Bridging filtering lets you ‘deny’ or ‘allow’ packets to cross the network based on position and hexadecimal content within the packet. This enables you to restrict or forward messages with a specified address, protocol or data content. Common uses are to prevent access to remote networks, control unauthorized access to the local network and limit unnecessary traffic.

For example, it might be necessary to restrict remote access for specific users on the local network. In this case, bridging filters are defined using the local MAC address for each user to be restricted. Each bridging filter is specified as a ‘deny’ filter based on the MAC address and position of the address within the packet. Deny filtering mode is then enabled to initiate bridge filtering. Every packet with one of the MAC addresses would not be bridged across the router until “deny” filtering mode was disabled.

Similarly, protocol filtering can be used to prevent a specific protocol from being bridged. In this case, the protocol id field in a packet is used to deny or allow a packet. You can also restrict, for example, the bridging of specific broadcast packets.

Configure Bridging Filtering

Bridging filtering allows you to control the packets transferred across the router. This feature can be used to enhance security or improve performance. Filtering occurs based on matched patterns within the packet at a specified offset. Two filtering modes are available:

- “Deny” mode will discard any packet matched to the “deny” filters in the filter database and let all other packets pass.

- “Allow” mode will only pass the packets that match the “allow” filters in the filter database and discard all others.

Up to 40 “allow” filters or 40 “deny” filters can be activated from the filter database.

You enter the filters, including the pattern, offset, and filter mode, into a filter database. If you intend to restrict specific stations or subnetworks from bridging, then add the filters with a “deny’ designation. Then enable filtering for deny. If you wish to allow only specific stations or subnetworks to bridge, then add the filters with an “allow” designation and enable filtering for “allow”. Add each filter with the following command:

filter br add [*pos*][*data*]deny|allow

where [*pos*] is the byte offset within a packet (number from 0-127) to a [*data*] (a hex number up to 6 bytes). This data and offset number can be used to identify an address, protocol id or data content. After you have entered all of the filters, verify your entries with the following command:

filter br list

If you have entered an incorrect filter, delete the filter using the **filter br del** command. When you are satisfied with the filter list, save the filtering database with the **save filter** command. You must reboot the router to load the filtering database. Then enable bridging filtering with the following command:

filter br use none|deny|allow

Test the filtering configuration by accessing a remote destination identified in the filter.

Enable/Disable Internet Firewall Filtering

The router supports IP Internet Firewall Filtering to prevent unauthorized access to your system and network resources from the Internet. This filter discards packets received from the WAN which have a source IP address recognized as a local LAN address. You can set Internet Firewall Filtering using the command:

eth ip firewall on|off|list

The Internet Firewall defaults to ON during initial configuration and is active *only* when Ethernet LAN IP routing is on.

As described earlier, Ethernet LAN IP routing is controlled by the commands:
eth ip enable
eth ip disable

Therefore, at initial configuration, you need only enable IP routing to activate the Internet Firewall Filter. If you do not wish the router to perform IP Internet Firewall Filtering while IP routing, you must turn OFF the Firewall Filter. Remember to save and reboot if you alter IP routing status.

IP (RIP) Protocol Controls

You can configure the router to send and receive RIP packet information to and from, respectively, the remote router. This means that the local site will 'learn' all about the routes beyond the remote router and the remote router will 'learn' all about the local site's routes. You may not want this to occur in some cases. For example, if you are connecting to a site outside of your company, such as the Internet, you may want to keep knowledge about your local site's routes private.

The default is to not send or receive IP RIP packets. If RIP packets are not allowed to flow on the WAN link, you must use the **remote addiproute** command to configure static routes for this WAN link. You can also advertise the local site's existence. The default is to keep the local site's existence private.

If you wish to allow sending or receiving RIP packets or default routes, or advertise the local site's existence, use the following command:

remote setipoptions <*option*> [**on**|**off**] <*remoteName*>
where <*option*> is:

rxrip	Receive IP RIP packets from the remote destination
rxrip1	Receive and process RIP-1 packets only
rxrip2	Receive and process RIP-2 packet only
rxdef	Receive the remote site's default route
txrip	Send IP RIP packets to the remote destination
txrip1	Send RIP-1 packets only
txrip2	Send RIP-2 packets only
txdef	Send the local site's default route
private	Keep the local site's existence private

RIP can be set on the LAN interface as well. See the eth ip options commands for more information.

DHCP (Dynamic Host Configuration Protocol)

This section describes how to configure DHCP using the Command Line Interface. Configuring DHCP can be a complex process; this section is therefore intended for network managers. Please refer to Chapter 4 for a complete list of the DHCP commands.

General Information

The router supports DHCP and acts as the DHCP server. DHCP is a service that allocates IP addresses automatically to any DHCP client (any device attached to your network such as your PC) requesting an IP address.

DHCP is used to acquire IP addresses and options (such as the subnet mask, DNS, gateway, etc.) automatically. On the practical level, acquiring these initialization parameters with DHCP translates into avoiding the more involved router/PC manual initialization process (reconfiguration of router and/or PC addresses to be in the same network).

To configure DHCP for a network, the network administrator defines a range of valid IP addresses to be used in the subnetwork as well as options and other parameters. Once DHCP is configured for the network, each DHCP client (your PC for example) can easily request an IP address from the pool of valid IP addresses. The DHCP client will learn part or all of the network parameters automatically. IP addresses and options assigned to a client are collectively called the lease. The lease is only valid for a certain period of time and is automatically renewed by the client. Note that the **Quick Start** configurator does a basic configuration of the DHCP server by asking for some common options.

Before becoming active, the router's DHCP server attempts to locate other active DHCP servers on the network such as Windows NT servers. If one is detected, the router's DHCP server disables itself.

DHCP administration and configuration is divided into the following parts:

- Manipulating subnetworks and explicit client leases
- Setting option values
- BootP
- Defining option types
- Configuring BootP/DHCP Relays
- Other information

Note 1: The TCP/IP stack has to be installed on the PCs for DHCP to work.

Note 2: In Windows, DHCP is enabled by selecting it on your PC (under **Settings, Control Panel, Network**, and **TCP/IP** in the **Configuration** tab page).

Note 3: To save the DHCP configuration or changes to FLASH memory in the router, make sure to use the command: **dhcp save**.

Manipulating Subnetworks and Explicit Client Leases

Enabling/disabling a subnetwork or a client lease

To enable/disable a subnetwork or a client lease, use the commands:

dhcp enable all | <*net*> <*ipaddr*>
dhcp disable all | <*net*> <*ipaddr*>

Examples:

To enable the subnetwork 192.168.254.0 if that subnetwork exists, type:
`dhcp enable 192.168.254.0`

`To enable the client lease 192.168.254.17 if that client lease exists, enter:`
`dhcp enable 192.168.254.17`

To disable the client lease 192.168.254.18 if that client lease exists, type:
`dhcp disable 192.168.254.18`

To check the results of these commands, use:

dhcp list

If the client lease does NOT exist, it must be explicitly created.

Adding subnetworks and client leases

♦ Adding a subnetwork

The following commands are used to add/delete subnetworks. Only one subnetwork with one pool of IP addresses may be defined for a subnet.

To add a subnetwork, use:

dhcp add <*net*> <*mask*>

To remove a subnetwork, use:

dhcp del <*net*>

Note: All client leases associated with this subnetwork are automatically deleted.

Examples:

The following command will create a subnetwork 192.168.254.0 with a subnet mask of 255.255.255.0:
`dhcp add 192.168.254.0 255.255.255.0`

The following command will delete the subnetwork 192.168.254.0 and will delete all client leases associated with that subnetwork:
`dhcp del 192.168.254.0`

♦ Adding explicit or dynamic client leases

Client leases may either be created dynamically or explicitly. Usually client leases are created dynamically when PCs boot and ask for IP addresses.

Explicit client leases

To add an explicit client lease, a subnetwork MUST already exist (use **dhcp add** <*net*> <*mask*> to add the subnetwork) before the client lease may be added. Use the command:

dhcp add <*ipaddr*>

To remove a client lease, use:

dhcp del <*ipaddr*>

Note: An administrator MAY create a client lease that is part of a subnet but does not fall within the pool of IP addresses.

Examples:

To explicitly add the client lease 192.168.254.31, type:
`dhcp add 192.168.254.31`

To delete the client lease 192.168.254.31, type:
`dhcp del 192.168.254.31`

Dynamic Client Leases

Dynamic client leases are created from the pool of IP addresses associated with that subnetwork. To set or change the pool, use:

dhcp set addresses <*first ip addr*> <*last ip addr*>

To clear the values from the pool, use:

dhcp clear addresses <*net*>

Note: Any client leases that currently exist will NOT be affected.

To remove a client lease that was dynamically created, use:

dhcp del <*ipaddr*>

Caution: If <*ipaddr*> is a subnet, you will delete the entire subnet.

Setting the lease time

♦ Concepts

The information given by the DHCP server (router) to your PC is leased for a specific amount of time. The client lease has already been selected. The DHCP server will select the lease time based on the option defined for the client lease as described by this algorithm:

1. If the client lease option is a specific number or is infinite, then the server uses the specified lease time associated with this client lease.

2. If the client lease option is "default", then the server goes up one level (to the subnetwork) and uses the lease time explicitly specified for the subnetwork.

3. If the client and subnetwork lease options are both "default", then the server goes up one level (global) and uses the lease time defined at the global level (server).

4. Lease time:
   The minimum lease time is 1 hour.
   The global default is 168 hours.

♦ Commands

The following commands are used by network administrators to control lease time.

To set the lease time explicitly for the client lease, use:

dhcp set lease <*ipaddr*> <*hours*>

To set the lease time explicitly for the subnetwork lease, use:

dhcp set lease <*net*> <*hours*>

To set the lease time explicitly for the global lease, use:

dhcp set lease <*hours*>

Examples:

To set the lease time to "default" for the client 192.168.254.17, type:

```
dhcp set lease 192.168.254.17 default
```

To set the subnetwork lease time to infinite for the subnet 192.168.254.0, type:

```
dhcp set lease 192.168.254.0 infinite
```

To set the global lease time to 2 hours, type:

```
dhcp set lease 2
```

Manually changing client leases

Administrators will generally NOT need to change client leases manually. However, if the need arises to do so, use the following commands.

WARNING: The client will not be aware that the administrator has changed or released a client lease!

This command will change the client lease expiration time to a given value:

dhcp set expire <*ipaddr*> <*hours*>

Setting the expiration time to “default” will cause the server to compute the lease time using the algorithm as described in section C, *Setting the lease time.*

Use this command to release the client lease so it becomes available for other assignments:

dhcp clear expire <*ipaddr*>

Setting Option Values

Administrators will want to set the values for global options, for options specific to a subnetwork, or for options specific to a client lease.

Note: See RFC 2131/2132 for the description of various options.

Concepts

The server returns values for options explicitly requested in the client request. It selects the values to return based on the following algorithm:

1. If the value is defined for the client, then the server will return the requested value for an option.
2. If the value for the option has not been set for the client, then the server returns the value option if it has been defined for the subnetwork.
3. If the valueoption does not exist for the client AND does not exist for the subnetwork, then the server returns the value option if it has been defined globally.
4. If the value option is not defined anywhere, the server will NOT return any value for that option in its reply to the client request.

Important: When replying to a client request, the server does:

- Not return any option values NOT requested by the client.
- Not support the definition of a "class" of clients.
- Not return any non-default option values UNLESS the client requests the option value AND the server has a value defined for that option.
- Not return any non-default values on the clients subnet UNLESS the client requests the value for that option.

Commands for global option values

To set the value for a global option, use:

dhcp set valueoption <*code*> <*value*>**...**

The code can be a number between 1 and 61 or a keyword.

To see the list of predefined and user-defined options, use:

dhcp list definedoptions

To clear the value for a global option, use:

dhcp clear valueoption <*code*>

Example:

To set the global value for the domain name server option, enter:

```
dhcp set valueoption domainnameserver 192.168.254.2 192.168.254.3
```

Commands for specific option values for a subnetwork

To set the value for an option associated with a subnetwork, use:

dhcp set valueoption <*net*> <*code*> <*value*>**...**

To clear the value for an option associated with a subnetwork, use:

dhcp clear valueoption <*net*> <*code*>

Examples:

```
dhcp set valueoption 192.168.254.0 gateway 192.168.254.254
dhcp set valueoption 6 192.84.210.75 192.84.210.68
```

Commands for specific option values for a client lease

To set the value for an option associated with a specific client, use:

dhcp set valueoption <*ipaddr*> <*code*> <*value*>**...**

To clear the value for an option associated with a specific client, use:

dhcp clear valueoption <*ipaddr*> <*code*>

Example:

```
dhcp set valueoption 192.168.254.251 winserver 192.168.254.7
```

Commands for listing and checking option values

To list the values for global options as well as subnet and client lease information, use:

dhcp list

To list options that are set for that subnet/client lease as well as subnet/client lease information, use:

dhcp list <*net*>|<*ipaddr*>

This command lists all available options (predefined and user-defined options):

dhcp list definedoptions

This command lists all available options starting with the string "name".

dhcp list definedoptions name

To list the lease time use:

dhcp list lease

Example:

This command lists the subnet 192.168.254.0 including any options set specifically for that subnet:

```
dhcp list 192.168.254.0
```

BootP

Administrators may wish to specify that certain client leases AND certain subnetworks can satisfy BootP requests.

About BootP and DHCP

BootP and DHCP provide services that are very similar. However, as an older service, BootP offers only a subset of the services provided by DHCP.

The main difference between BootP and DHCP is that the client lease expiration for a BootP client is always INFINITE.

Caution: Remember that when BootP is enabled, the client assumes that the lease is infinite.

By default, the DHCP server will NOT satisfy BootP requests unless the administrator has explicitly enabled BootP (at the subnetwork or lease level).

Enable/Disable BootP

To allow BootP request processing for a particular client/subnet, use the command:

dhcp bootp allow <*net*>|<*ipaddr*>

To disallow BootP request processing for a particular client/subnet, type:

dhcp bootp disallow <*net*>|<*ipaddr*>

Use BootP to specify the boot server

The following commands let the administrator specify the TFTP server (boot server) and boot file name. The administrator will first configure the IP address of the TFTP server and file name (kernel) from which to boot. This is particularly useful if the kernel in the router's flash is corrupt or does not exist.

To set the IP address of the server and the file to boot from, use the command:

dhcp bootp tftpserver [<*net*>|<*ipaddr*>] <*tftpserver ipaddr*>

dhcp bootp file [<*net*>|<*ipaddr*>] <*file name*>

To clear the IP address of the server and the file to boot from, use:

dhcp bootp tftpserver [<net>|<*ipaddr*>] 0.0.0.0

Examples:

To set the global BootP server IP address to 192.168.254.7:

```
dhcp bootp tftpserver 192.168.254.7
```

To set the subnet 192.168.254.0 server IP address to 192.168.254.8:

```
dhcp bootp tftpserver 192.168.254.0 192.168.254.8
```

To set the client 192.168.254.21 server IP address to 192.168.254.9

```
dhcp bootp tftpserver 192.168.254.21 192.168.254.9
```

To set the subnet 192.168.254.0 boot file to "kernel.100":

```
dhcp bootp file 192.168.254.0 kernel.100
```

To clear the global BootP server IP address and file name:

```
dhcp bootp tftpserver 0.0.0.0
```

To clear the subnet 192.168.254.0 server IP address and file name:

```
dhcp bootp tftpserver 192.168.254.0 0.0.0.0
```

Defining Option Types

Concepts

A DHCP option is a code, length, or value. An option also has a "type" (byte, word, long, longint, binary, IP address, string).

The subnet mask, router gateway, domain name, domain name servers, NETBIOS name servers are all DHCP options. Please refer to RFC 1533 if you require more information.

Usually users will not need to define their own option types. The list of predefined option types based on RFC 1533 can be shown by typing:

dhcp list definedoptions

Commands

The following commands are available for adding/deleting option types:

dhcp add <*code*> <*min*> <*max*> <*type*>

To list option types that are currently defined, use:

dhcp list definedoptions...

To list the definitions for all known options, use:

dhcp list definedoptions

To get help information, use:

dhcp list definedoptions?

To list the definition for option 1, if option 1 is defined, type:

```
dhcp list definedoptions 1
```

To list the definition for all options that are well-known AND have a name starting with 'h', type:

```
dhcp list definedoptions h
```

Example:

To define a new option with a code of 128, a minimum number of IP addresses of 1, a maximum number of IP addresses of 4, of type “IP address”, type:

```
dhcp add 128 1 4 ipAddress
```

This information implies that:

- Some DHCP client will know about the option with code 128.
- Option 128 allows IP addresses.
- The server can have a minimum of 1 IP address.
- The server can have up to 4 IP addresses.
- The administrator will still need to set the option value either globally, specific to a subnetwork, or specific to a client for the option to have any meaning.

To delete the definition of the option with code 128, type:

```
dhcp del 128
```

The values for this option that have been set globally, specific to a subnetwork, or specific to a client will NOT be removed. The administrator must remove those values explicitly. Well-known type option codes CANNOT be changed or deleted.

Configuring BootP/DHCP Relays

BootP/DHCP Relays are used by system administrators when the DHCP configuration parameters are acquired from a BootP/DHCP server other than the router’s DHCP server.

This feature allows configuration information to be centrally controlled. Enabling a BootP/DHCP Relay disables DHCP on the router since (by definition) only one policy mechanism can be supported.

BootP/DHCP Relays are enabled and disabled using the command:

system bootpserver

Other Information

DHCP information is kept in the file DHCP.DAT. This file is self contained.

This file contains ALL of the DHCP information including:

- the option definitions
- the subnetwork that have been added
- the client lease information
- the option values that have been set
- This file can be uploaded/downloaded from one router to another.

NAT (Network Address Translation)

The router supports classic NAT (one NAT IP address assigned to one PC IP address) and a NAT technique known as masquerading (one single NAT IP address assigned to many PC IP addresses).

General NAT Rules

1. IP Routing must be enabled.
2. NAT can be run on a per-remote-router basis.
3. Any number of PCs on the LAN may be going to the same or different remote routers at the same time. In reality, the number of PCs on the LAN that can be supported is limited by how much memory the router consumes maintaining table information -AND- by how many connections are currently active.
4. Some operations will NOT work. Specifically, services that place IP address/port information in the data MAY NOT WORK until the router examines their packets and figures out what information in the data needs to be changed. Remember that the router is remapping both IP addresses and ports.
5. When using NAT with a remote router, either the remote ISP MUST supply the IP address for NAT translation -or- the user MUST configure the IP address for NAT translation locally.
6. Any number of PCs on the LAN may have a connection to the same or different remote routers at the same time. In reality, the number of PCs on the LAN that can be supported is limited by the amount of memory consumed by the router to maintain table information -AND- by the number of connections the router “thinks” are currently active. Theoretically, up to 64,000 active connections per protocol type - TCP/UDP - can be concurrently running, if the table space is available.

Masquerading (one single NAT IP address shared by many PC IP addresses)

With this form of NAT, multiple local (PC) IP addresses are mapped to a single global IP address. Many local (PCs) IP addresses are therefore hidden behind a single global IP address. The advantage of this type of NAT is that users only need one global IP address, but the entire local LAN can still access the Internet. This NAT technique requires not only remapping IP addresses but also TCP and UDP ports.

Each PC on the LAN side has an IP address and mask. When the router connects to an ISP, the router appears to be a HOST with one IP address and mask. The IP address that the router uses to communicate with the ISP is obtained dynamically (with PPP/IPCP or DHCP) or is statically configured. When the PC connects to the ISP, the IP address and Port used by the PC are remapped to the IP address assigned to the router. This remapping is done dynamically.

Client Configuration

♦ Enable NAT

To enable NAT, use the commands:

remote setIpTranslate on <*remoteName*>
save

The **save** command makes the above changes persistent across boots which turn NAT on when connected to this remote router.

♦ Obtain an IP Address for NAT Translation

The IP address (the IP address "known" by the remote ISP) used for this type of NAT translation can be assigned in two ways.

The ISP dynamically assigns the IP address. Use the commands:

remote setSrcIpAddr 0.0.0.0 0.0.0.0 <*remoteName*>
save

The IP address is assigned locally. Use the commands:

remote setSrcIpAddr *ww.xx.yy.zz* 255.255.255.255 <*remoteName*>
save

Note: *ww.xx.yy.zz* is the IP address the user on the local LAN is assigning.

Server Configuration

This section is intended for users and network administrators who wish to allow WAN access to a Web server, FTP server, SMTP server, etc., on their local LAN, while using NAT.

NAT needs a way to identify which local PC (local IP address(es)) should receive these server requests. The servers can be configured on a per-remote-router basis as well as globally.

♦ Remote Commands

The following two commands are used to enable/disable a local IP address (on your LAN) as the server for a particular protocol for the remote router <*remoteName*>.

remote addServer <*ipaddr*> |discard|me <*protocolid*> tcp|udp <*first port*> ftp|telnet|smtp|snmp|http [<*last port*>[<*first private port*>]] <*remoteNam*e>

remote delServer <*ipaddr*> |discard|me <*protocolid*> tcp|udp <*first port*> ftp|telnet|smtp|snmp|http [<*last port*>[<*first private port*>]] <*remoteName*>

first port: it is the first or only port as seen by the remote end.

last port: if specified, it is used with <first port> to specify a range of ports as seen by the remote end for the server on your LAN.

first private port: if specified, it is a port remapping of the incoming request from the remote end.
first port maps to *first private port*.
first port + 1 maps to *first private port* + 1.

last port maps to *first private port* + *last port* - *first port*

first port through *last port* are the ports as seen by the remote end.
first private port through *first private port* + *last port* - *first port* are the equivalent ports the server on your local LAN will receive the request.

This command is used to view all of the remote entries, including the changes.
remote list <*remoteName*>

Remember to type **`save`** to make the changes persistent across boots.

Example 1:

Assume that the local LAN network is 192.168.1.0 255.255.255.0. The following commands are typed to enable a Telnet server on the local LAN with the IP address 192.168.1.3, and an FTP server with the IP address 192.168.1.2.

```
remote addServer 192.168.1.3 tcp telnet router1
remote addServer 192.168.1.2 tcp ftp router1
```

When receiving a request from *router1* to communicate with the local Telnet server, the local router will send the request to 192.168.1.3. If *router1* asks to talk to the local FTP server, the local router will send the request to 192.168.1.2.

Example 2:

Assume that the local LAN network is 192.168.1.0 255.255.255.0. When the port value of 0 (zero) is used, it directs all ports of the specified protocol to the IP address specified.

```
remote addServer 192.168.1.4 tcp 0 router1
```

Note: **addserver** commands using specific port numbers take priority over the port # 0 setting. 192.168.1.4 will be asked to serve requests coming from *router1* to the local router. If the local router also has the same Telnet and FTP entries from the previous example, 192.168.1.3 will serve the Telnet request, 192.168.1.2 will serve the FTP request, and 192.168.1.4 will serve any other request, including HTTP, SMTP, etc.

Example 3:

```
remote addServer 192.168.1.10 tcp 9000 9000 telnet route-in
remote addServer 192.168.1.11 tcp 9001 9001 telnet route-in
```

In this example, an incoming request on tcp port 9000 will be sent to 192.168.1.10 with the port changed from 9000 to the telnet (port 23).

An incoming request on tcp port 9001 will be sent to 192.168.1.11 with the port changed from 9001 to the telnet port.

“Failed to add server” error message

The error message *“Failed to add server"* is printed if a server entry could not be created. Possible reasons are as follows:

- Port overlap: One or more of the ports that would be visible to the remote end overlap.

  Example:

  ```
  remote addserver 192.168.1.10 tcp 9000 9000 telnet router1
  ```

  Let us assume this command is accepted.

  ```
  remote addserver 192.168.1.11 tcp 9000 9000 telnet router1
  ```

  Let us assume this command gets an error.

  For the remote end sending a server request to port 9000, it is impossible to know to which server, 192.168.1.10 -or- 192.168.1.11, to send the request, if both entries exist.

- Not enough memory was available to create an entry. This condition should not happen. The amount of memory needed for a server entry is less than 30 bytes; understandably, if this problem occurs, a lot of problems/failures will arise.

♦ System Commands

The following two commands are used to globally enable/disable a local IP address (on your LAN) as the server for that particular protocol.

system addServer <*ipaddr*> discard|me <*protocolid*> tcp|udp <*first port*> ftp|telnet|smtp|snmp|http [<*last port*>[<first *private port*>]]

system delServer <*ipaddr*> discard|me <*protocolid*> tcp|udp <*first port*> ftp|telnet|smtp|snmp|http [<*last port*>[<first *private port*>]]

first port: it is the first or only port as seen by the remote end.

last port: if specified, it is used with <*first port*> to specify a range of ports as seen by the remote end for the server on your LAN.

first private port: if specified, it is a port remapping of the incoming request from the remote end.
first port maps to *first private port.*
first port + 1 maps to *first private port* + 1

last port maps to *first private port* + *last port* - *first port*

first port through *last port* are the ports as seen by the remote end.
first private port through *first private port* + *last port* - *first port* are the equivalent ports the server on your local lan will receive the request.
Remember to type **save** to make the changes persistent across boots.

Examples:

```
system addserver 192.168.1.5 tcp smtp
system addserver 192.168.1.6 tcp 0
system addserver 192.168.1.6 udp 0
```

The router sends a server request for SMTP to 192.168.1.5 when such a request comes from any remote router running NAT. The router sends any other server request (tcp or udp) to 192.168.1.6.

♦ Server Request Hierarchy

When handling a request from a remote router (to which the local router has NAT enabled), the local router selects a server based on the following priority (order) algorithm:

1. **remote addserver** — The local router selects a server for the remote router that handles that particular protocol/port.
2. **system addserver** — The local router selects a global server that handles that particular protocol/port.
3. **remote addserver** with *port* 0 — The local router selects a server for the remote router that handles that particular protocol (such as tcp/udp) and ANY port.
4. **system addserver** with *port* 0 — The local router selects a global server that handles that particular protocol and ANY port.
5. If an IP address is used for true NAT host remapping as well as for IP address/port translation, the IP address of the local remapped host as the server is selected.
6. Router's **IP address** — The local router selects itself (the local router) as the server.

Classic NAT (one NAT IP address assigned per one PC IP address)

With classic NAT, one PC IP address is translated to one NAT IP address. This NAT technique is primarily used to make certain hosts on a private LAN globally visible and give them the ability to remap these IP addresses as well.

Client Configuration

Classic NAT requires that you first enable NAT Masquerading as described in the previous section; thus, for the Classic and Masquerading forms of NAT, the clients are configured in the same way. Please, refer to the Client Configuration section, page 86.

Host Remapping

♦ Remote Commands

Use these commands to enable or disable host remapping on a-per-remote basis:

remote addHostMapping <*first private addr*> <*second private addr*> <*first public addr*> <*remoteName*>

remote delHostMapping <*first private addr*> <*second private addr*> <*first public addr*> <*remoteName*>

Use **remote addHostMapping** when a host on the local LAN is known by different IP addresses to different remotes.

♦ System Commands

Use these commands to enable or disable host remapping systemwide:

system addHostMapping <*first private addr*> <*second private addr*> <*first public addr*>
system delHostMapping <*first private addr*> <*second private addr*> <*first public addr*>

Use the **system addHostMapping** when a host on the local LAN is known by the same IP address on all remotes.

♦ IP Address Range

The range of local LAN IP addresses to be remapped is defined by <*first private addr*> *to* <*second private addr*> inclusive. These addresses are mapped one-to-one to the public addresses.

The range of public IP addresses is defined by <*first public addr*> only. The rest of the range is computed automatically (from <*first public addr*> to <*first public addr*> + number of addresses remapped - 1) inclusive.

♦ Multiple Host Remapping Entries

Users may have as many host remapping entries as they wish.

Example:

```
remote addHostMapping 192.168.207.40 192.168.207.49 10.0.20.11
remoteName
remote addHostMapping 192.168.207.93 192.168.207.99 10.0.20.4
remoteName
remote addHostMapping 192.168.209.71 192.168.209.80 10.12.14.16
remoteName
```

The above entries create three mappings:

192.168.207.40 through 192.168.207.49 are mapped to 10.0.20.11 through 10.0.20.20
192.168.207.93 through 192.168.207.99 are mapped to 10.0.20.4 through 10.0.20.10
192.168.209.71 through 192.168.209.80 are mapped to 10.12.14.16 through 10.12.14.25

♦ Range Overlap Rules

With **remote addHostMapping,** private IP address ranges cannot overlap for a remote router.
With **remote addHostMapping**, public IP address ranges cannot overlap for a remote router.

With **system addHostMapping**, private IP address ranges cannot overlap for a system.

With **system addHostMapping**, public IP address ranges cannot overlap for a system.

If a private IP address range for a remote router and a private IP address range for the system overlap, the private IP address range for the remote has precedence.

If a public IP address range for a remote and the public IP address range for the system overlap, the public IP address range for the remote has precedence.

Private IP addresses and public IP addresses can be the same.

For example, to enable IP/port translation to a remote router and make the IP addresses 10.1.1.7 through 10.1.1.10 globally visible, it is permissible to use either one of the following commands:

```
remote addHostMapping 10.1.1.7 10.1.1.10 10.1.1.7 remoteName
system addHostMapping 10.1.1.7 10.1.1.10 10.1.1.7
```

If the remapped host's IP address (classic NAT, one-to-one IP address translation) and the "masquerading" IP address (many-to-one IP address translation) are the same, then NAT masquerading has precedence over classic NAT.

Management Security

With the following security control features, the user can prevent the router from being remotely managed via Telnet and/or SNMP. Disabling SNMP will stop the Configuration Manager from accessing the router. In some environments this is desirable.

Disable Telnet and SNMP

To completely disable remote management, the following commands should be entered from the command line.

login admin
system telnetport disable
system snmpport disable
save
reboot

Restore Telnet and SNMP

In order to reestablish the Telnet and SNMP services, you should restore the default values with the commands:

system telnetport default
system snmpport default

Validation of Telnet and SNMP clients

The following commands are used to validate Telnet, SNMP, or HTTP clients. They define a range of IP addresses that are allowed to access the router via Telnet, SNMP, or HTTP. Only the IP addresses in the range specified for Telnet, SNMP, or HTTP can access the router via Telnet, SNMP, or HTTP. This validation feature is **off** by default.

system addtelnetFilter <*first ip addr*> [<*last ip addr*>] | LAN

system addSNMPFilter <*first ip addr*> [<*last ip addr*>] | LAN

system addHTTPFilter <*first ip addr*> [<*last ip addr*>] | LAN

Where:

first ip addr	First IP address of the client range
last ip addr	Last IP address of the client range. May be omitted if the range contains only one IP address.
LAN	Local Ethernet LAN

`Example:`

```
system addsnmpfilter 192.168.1.5 192.168.1.12
```

Multiple range can be specified for Telnet and SNMP clients. If no range is defined, then access to the router is through the LAN or WAN.

Note 1: These commands do not require a reboot and are effective immediately.

Note 2: The following commands are used to delete client ranges previously defined by the **system addtelnetFilter, system addSNMPFilter, system addHTTPFilter** commands:

system deltelnetFilter <*first ip addr*> [<*last ip addr*>] | LAN

system delSNMPFilter <*first ip addr*> [<*last ip addr*>] | LAN

system delHTTPFilter <*first ip addr*> [<*last ip addr*>] | LAN

Note 3: To list the range of allowed clients, use the command **system list** when logged in with read and write permission (login with password).

Restrict Remote Access

To allow management via SNMP or Telnet, while making it more difficult for non-authorized personnel to access the router, the Telnet and SNMP ports may be redefined to a non well-known value. When Network Address Translation (NAT) is used, this port redefinition feature also allows to continue using the standard Telnet and SNMP ports with another device on the LAN (provided the appropriate NAT server ports commands are issued), while simultaneously managing the router (with non-standard ports). The following commands show how this is done.

Example:

```
login admin
system telnetport 4321
system snmpport 3214
```

Changing the SNMP Community Name

Changing the SNMP community name from its default value of “public” to another string may further enhance SNMP security. This string then acts like a password, but this password is sent in the clear over the WAN/LAN, in accordance with the SNMP specification.

The SNMP community name is changed using the following commands:

login admin
system community <*snmp community name*> **-- (eg:** system community fred)
save
reboot

Disable WAN Management

It may be desirable to allow management of the router on the local LAN, but not over the WAN Network. If the router has been configured to use NAT, then by defining two servers, that DO NOT exist, on the LAN side to handle WAN SNMP and Telnet requests, WAN management of the router cannot occur. The following commands show how this could be done.

Example:

```
login admin
system addServer 192.168.254.128 udp snmp - (no computer at 192.168.254.128)
system addServer 192.168.254.128 tcp telnet
save
reboot
```

Software Options Keys

This router has several optional software features that can be purchased as software options keys, when ordering the router. These optional features are:

- DES encryption (For more information on this feature, refer to *Encryption*, page 95)
- IP filters (For more information on this feature, refer to *IP Filtering*, page 98)
- L2TP Tunneling (For more information on this feature, refer to *L2TP Tunneling - Virtual Dial-Up*, page 101)

These options are usually ordered with the router.

To find out which software options are installed on your router, use the **vers** command. The following provides a sample output of the **vers** command:

```
Maximum users: unlimited
Options: SDSL, IP, ~IP FILTERING, IP TRANS, HOST MAPPING, DHCP, ~L2TP,
~ENCRYPT, BRIDGE, IPX
```

The features present in the firmware, but not enabled are preceded by a "~". These features can be enabled by purchasing a software key from your distributor.

To install a software options key purchased separately, follow the instructions provided with that software key.

Encryption

Note: Encryption is a software option. The following section applies only for routers with this option.

For routers shipped with the following encryption options, two variants of encrypted data links over PPP have been implemented:

- PPP DES (RFC1969)
- Diffie-Hellman

Encryption requires PPP.

Caution: DES and Diffie-Hellman encryption options are not available for export outside of the United States or Canada.

PPP DES (RFC 1969) Encryption

PPP DES (Data Encryption Standard) implementation uses a 56-bit key with fixed transmit and received keys that are specified in each router. With RFC 1969, users must manage the keys. This implementation has been tested for interoperability with other PPP DES vendors such as IBM, Network Express (part of Cabletron), and a few others.

Configuration Notes

Simply add the encryption commands to your standard configuration. For PPP DES, the encryption commands are:

remote setEncryption dese rx <*key*> <*remoteName*>
remote setEncryption dese tx <key> <*remoteName*>

Observe the following guidelines:

- PPP DES can only be configured using the Command Line Interface (CLI).
- The choice of keys should be carefully considered: they must have eight hexadecimal digits and values that are considered cryptographically weak should be avoided. Consult a security expert for advice.
- Use the console port or a telnet port (use the system log command) to view error messages and progress. If you see 'Unknown protocol' errors, the router receive key and sender Tx key don't match.
- Different keys may be used with different remote destinations.
- For maximum security, as shown in the following configuration examples, Telnet and SNMP access should be disabled and PPP CHAP authentication should be used by both ends.

Sample Configuration

Refer to the section *Sample Configurations*, Chapter 3 of this manual, page 57. The routers SOHO (the target router) and HQ (the remote router) are configured in the same manner as shown in chapter 3, but the following encryption commands are added. Don't forget to save the configuration and reboot the router (**save** and **reboot** commands).

Remember that the transmit key (tx) of SOHO is the receive key (rx) of HQ. Inversely, the receive key of SOHO is the transmit key of HQ.

Use this sample configuration with the additional encryption commands as a guideline to configure your own routers.

♦ Enable encryption on the router HQ

```
Sample:
login admin
remote setEncryption dese rx 1111111111111111 SOHO
remote setEncryption dese tx 2222222222222222 SOHO
save
reboot
```

♦ Enable encryption for the router SOHO

```
Sample:
remote setEncryption dese tx 1111111111111111 HQ
login admin
remote setEncryption dese rx 2222222222222222 HQ
save
reboot
```

Diffie-Hellman Encryption

With Diffie-Hellman encryption, each router has an encryption file that is associated with a public key providing 768-bit security. The predefined keys can be replaced by the user. The key files have a suffix of "num" by convention (e.g.; dh96.num).

Configuration Notes

Simply add the encryption command to your standard configuration. For Diffie-Hellman, the encryption command is:

remote setEncryption DESE_1_KEY|DESE_2_KEY [<*fileName*>]| <*remoteName*>

Observe the following guidelines:

- DESE_1_KEY specifies that the same key is used in both directions and DESE_2_KEY specifies that the keys are different. Having the same keys in both directions can significantly reduce time needed to compute the DES keys from the Diffie-Hellman exchange.

- routers' "receive" key and "sender" Tx key don't match.
- Different keys and key files may be used with different remote destinations.
- For maximum security, as shown in these examples, Telnet and SNMP access should be disabled and Use the console port to view error messages and progress. If you see "Unknown protocol" errors, the PPP CHAP used.

Sample Configuration

The sample configuration is the same as the one provided in the preceding PPP DES Encryption example, but use the Diffie-Hellman encryption command instead of the PPP DES encryption commands.

Sample:

```
login admin
remote setEncryption DESE_1_KEY dh96.num SOHO
save
reboot
```

File Format for the Diffie-Hellman Number File

The file consists of 192 bytes, in binary format. There are two 96-byte numbers stored, with the most significant byte in the first position. For example, the number 0x12345678 would appear 000000...0012345678.

The first 96 bytes form the modulus. In the equation x' = g^x mod n, n is the modulus. According to Diffie and Hellman, the modulus should be prime, and (n-1)/2 should also be prime.

The second 96 bytes form the generator, or g in the above equation. The generator should be a primitive root mod n.

The remaining pieces of the encryption key (x and y) are randomly generated at connection time, and will change every time the device connects.

You should contact an encryption expert to obtain cryptographically sound generator and modulus pairs, should you wish to change the default values.

♦ Default Modulus

```
00000000: c9 b4 ed 33 ba 7f 00 9e - ce e0 83 5d a5 4c 19 25
00000010: e0 2d 99 44 e8 8d cd 16 - 02 0e 6c 26 6d 15 7c 95
00000020: 82 9a 8c 2b 19 d0 56 da - 9b 5b a9 cd cf fb 45 2b
00000030: c9 6a 3c 26 e5 b8 1a 25 - 07 b8 07 22 ed 15 8a 56
00000040: 8b f4 30 f2 28 fc 6b f1 - bf a4 3e 87 f0 be d6 1c
00000050: 33 92 b9 5e d1 b7 20 8c - 92 02 cb e5 26 45 02 1d
```

♦ Default Generator

```
00000000: 90 f0 09 78 cc 23 79 a8 - 6c 23 a8 65 e0 dc 0f 6d
00000010: fb a7 26 e8 63 0a 21 67 - 5a f8 0f 59 84 09 5c da
00000020: ef af af fc d2 5f 83 e2 - a7 27 05 34 17 94 1a 4f
00000030: b2 87 76 97 e7 48 43 db - 62 29 70 9e 7f eb 2c 6e
00000040: 5d 25 1d a1 65 f0 b4 e6 - 47 4d 25 23 0b 20 b9 93
00000050: 27 f0 56 12 5a 97 f6 c5 - 31 b6 19 fc 67 22 93 f5
```

IP Filtering

Note: Filtering is a software option. The following section applies only for routers with this option.

IP Filtering is a type of Firewall used to control network traffic: the process involves filtering packets received from one interface then and deciding whether to route them to another interface or discard them.

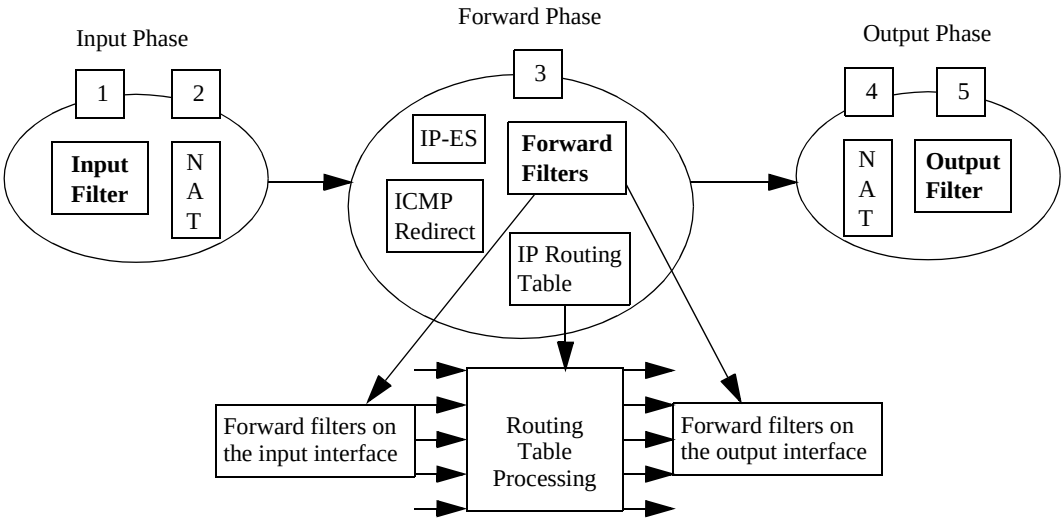
When filtering packets, the router examines information such as the source and destination address contained in the IP packet, the type of connection, etc., and then screens (filters) the packets based on this information: packets are either allowed to be forwarded from one interface to another interface or simply discarded.

IP filtering requires IP routing to be enabled. This type of filtering offers great flexibility and control of IP filters, but configuration of this feature requires using a series of commands that may appear complex to a casual user.

Filters and Interfaces

Filters are commands used to screen IP packets: packets are simply matched against a series of filters. As a result of this process, the packets are either allowed to come through the interface/link or are dropped. If no filter "matches" the incoming packet, the packet is accepted by default.

Filters "operate" at the interface level. Each particular interface has a series of IP filters associated with it and is defined by 3 types of filters: Input filters, Output filters, and Forward filters. A list of filters is created for each interface. The following illustrates the filter process.



In the following description of the Input, Forward, and Output phases, the reference numbers associated with filtering steps match the numbers used in the above illustration.

Input Phase

When an IP packet comes in through an interface (i.e., the Input interface), the router tries to recognize the packet. The router then examines the Input filters for this interface and based on the first Input filter that matches the IP packet, it decides how to handle the packet (forward or discard it).

If NAT translation is enabled for the Input interface, NAT translation is performed.

Forward Phase

At this stage, the router determines to which interface or link the packets will be sent out using its routing table; it then applies the Forward filters based on the Input interface information. The Forward filters based on the Output interface information are applied next.

Output Phase

If NAT translation is enabled for the Output interface, then NAT translation is performed.

The router examines the Output filters for this interface and based on the first Output filter that matches the IP packet, it decides how to handle the packet.

Configuring Filters with Network Address Translation (NAT) Enabled

General NAT Information

Network Address Translation is an IP address conversion feature that translates a PC's local (internal) address into a global (outside/Internet) IP address. NAT is needed when a PC (or several PCs) on a Local Area Network wants to connect to the Internet or get to a remote network which uses global, registered addresses: NAT swaps the local IP address to a global IP address: the IP address and Port information that the PC uses are remapped (changed) to the IP address that was assigned to the router and a new Port Number is assigned.

The preceding section, Filters and Interfaces, describes how NAT "behaves" for each filtering phase.

Filter Actions

For an IP packet to be forwarded successfully, a filter at each implementation point (Input, Forward, and Output) MUST accept the IP packet.

If NO filter at a particular point matches the incoming IP packet, it is assumed that the packet is accepted.

Each IP filter can initiate one of the following 3 possible actions:

Accept

When the packet is accepted at a filter interface (Input, Forward, or Output), the router lets it proceed for further processing.

Drop

With Deny, the packet is silently discarded.

Reject

With Reject, an ICMP REJECT (Internet Control Management Protocol) is sent to reject the packet.

IP filter commands

The following two commands are used respectively to define IP filters on the Ethernet interface and on the remote interface. For extensive information on the syntax of these two commands, please refer to the *Command Line Interface Reference* chapter.

eth ip filter <*command*> <*type*> <*action*> <*parameters*> *[<port#>]*

remote ipfilter <*command*> <*type*> <*action*> <*parameters*> <*remoteName*>

Special notes

IP filters of Input type are checked BEFORE the IP packet is redirected by ICMP. This could adversely affect local LANs that use ICMP redirect to dynamically learn IP routes. IP filters of Input type are checked BEFORE the IP packet is sent to the router itself as a host.

Example:

The following commands will stop ANY attempt by a host coming from the remote <internet> from sending an IP packet to the telnet port. Hence, the router will not see the packet; the packet will not be forwarded anywhere.

```
remote ipfilter insert input drop -p tcp -dp 23 internet
save
```

These commands will stop ANY attempt by a host coming from the remote <internet> from sending an IP packet to the telnet port "through" the router to a different interface. The router itself could still receive the IP packet so the remote host could telnet to the router itself.

```
remote ipfilter insert forward drop -p tcp -dp 23 internet
save
```

L2TP Tunneling - Virtual Dial-Up

This document has four parts:

- The *Introduction* provides a general overview of L2TP tunneling.
- The *L2TP Concepts* section explains LNS, L2TP client, LAC, dial user, tunnels, and sessions.
- *Configuration* describes preliminary configuration steps and verification steps and lists commands associated with the configuration of L2TP and PPP sessions.
- The *Sample Configurations* section provides two examples with step-by-step instructions: a simple L2TP client configuration example and a complete LNS and L2TP client configuration example.

Introduction

L2TP (Layer 2 Tunneling Protocol) is used to forward a PPP link from a remote site to a corporate site across the Internet, thus creating virtual paths called tunnels. Because tunneling involves encapsulating data, packets can be transported across networks using different protocols. The advantages for tunneling the PPP protocol are listed below:

- Different network protocols such as NetBEUI, IPX, and Appletalk can be transported through the Internet using a tunnel. The protocol packets are encapsulated and routed across the network through the Internet.
- Tunnels provide a way to reduce costs and complexity associated with remote dial-up networking by using a local ISP: users connect the remote site by dialing into their local ISP and let the Internet handle the long-distance connections, thus avoiding long-distance phone charges.
- Tunneling PPP allows compression of data through the entire tunnel, which translates into greater throughput.
- By allowing encryption over the PPP link, L2TP contributes to more secure networks over the Internet.
- Remote users can access the company network, even if there is a company firewall (provided, of course, that tunnels can come through the firewall).

Note: This feature can interoperate with any vendor that supports L2TP - Draft II.

L2TP Concepts

This section defines the major L2TP concepts such as LNS, L2TP client, LAC, and Dial user. These concepts are illustrated with L2TP client examples. Also described are tunnels and sessions' creations and destructions.

LNS, L2TP Client, LAC, and Dial User

An L2TP tunnel is created between an L2TP client and LNS. The L2TP client and LNS control the tunnel using the L2TP protocol.

Since routers are more often configured as L2TP clients or LNS than as LACs, this document, therefore, emphasizes L2TP client- and LNS-related information.

♦ LNS (L2TP Network Server)

The LNS is the point where the call is actually managed and terminated (e.g. within a corporate network).

♦ L2TP Client

With an L2TP client, the dial user and LAC are combined in the same hardware device. In this case, the PPP session is between the LAC and the LNS.

As shown in the following illustration (figure 1), an L2TP client is used to tunnel a PPP session between a small office (our router) and a corporate office through the Internet.

♦ LAC (L2TP Access Concentrator)

The LAC can be envisioned as the physical hardware (e.g. a router) used for placing and receiving phone calls.

♦ Dial User

A dial user is the remote system or router that is either placing the call to the LAC or receiving the call from the LAC.

The dial user does not actually dial in to the LNS or receive a call from the LNS, since this is a virtual connection.

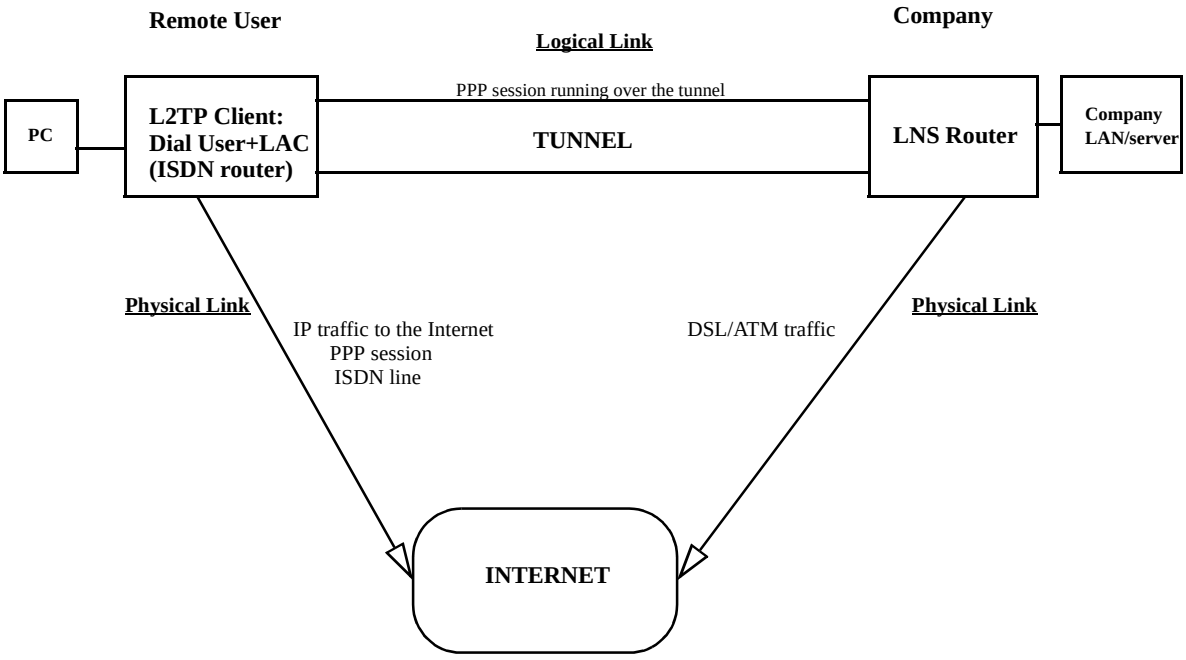
The dial user is one end of a PPP session. The LNS is the other end of the PPP session.

L2TP Client Example

The tunnel uses UDP/IP traffic as the transport medium over IP. This implementation of L2TP as illustrated below shows a tunnel from a remote user's perspective.

Note: There is one PPP session over ISDN and another PPP session over the tunnel.

Figure 1



LNS and L2TP Client Relationship

The LNS acts as the supervising system. The L2TP client acts both as the dial user and the LAC.

One end of the tunnel terminates at the L2TP client. The other end of the tunnel terminates at the LNS.

One end of the PPP session going through the tunnel terminates at the L2TP client acting as the dial user, the other end terminates at the LNS.

Tunnels

Tunnels are virtual paths that exist between an L2TP client and LNS.

An LNS can communicate simultaneously with more than one L2TP client.

An L2TP client can communicate simultaneously with more than one LNS.

Some L2TP implementations including the one discussed in this section allow the SAME router to act as BOTH a L2TP client and LNS simultaneously, if so configured.

Caution: Verify that the IP address of the other end of the tunnel is correctly routed through the right, local interface/remote and will not appear to be routed through the tunnel. An attempt to route the tunnel endpoint within itself will fail.

Sessions

Sessions can be thought of as switched virtual circuit “calls” carried within a tunnel and can only exist within tunnels. One session carries one “call”. This “call” is one PPP session. Multiple sessions can exist within a tunnel. The following briefly discusses how sessions are created and destroyed.

♦ Session creation

Traffic destined to a remote entry (located at the end of the tunnel) will cause a tunnel session to be initiated. When the L2TP client wishes to establish a session to an LNS, the L2TP client assumes the role of a LAC and sends control packets containing incoming call information to the LNS over the tunnel.

♦ Session destruction

A tunnel session will automatically time out after the data session stops. When instructed to destroy a session, the L2TP client closes any PPP session associated with that session. The L2TP client may also send control messages to the LNS indicating that the L2TP client wishes to end the PPP session.

When the LNS wants to hang up the call, it sends control messages destroying the session.

Configuration

Preliminary Steps to Configure a Tunnel

The following logical steps should be considered before configuring a tunnel:

1. Decide if the router will act as an L2TP Client or LNS.
2. Decide if one side or both sides of the connection can initiate a tunnel.
3. Create the L2TP Tunnel Entry with these characteristics:
   - A L2TP client host name
   - A LNS host name
   - A Tunnel CHAP secret (both side of the connection must use the same secret)
   - The IP address of the other party must be provided to the initiating side of the tunnel
   - Type of flow control (pacing, sequence numbers or not)
4. Create a remote entry for the PPP session. Associate the remote entry with the Tunnel.

Verification Steps

1. Verify that the IP address of the other end of the tunnel is correctly routed through the right, local interface/remote and will not appear to be routed through the tunnel. An attempt to route the tunnel endpoint within itself will fail.

2. Trying to establish IP connectivity (using the **ping** or **tracert** commands).

    a. "Pinging" from the L2TP client or LNS to the opposite tunnel endpoint will succeed (this tests the tunnel path).
    b. "Pinging" from a tunnel endpoint IP address to an IP address within the tunnel will probably fail due to the existence of the IP firewall..

Configuration Commands

There are two categories of L2TP commands and they are respectively associated with:

- Tunnels and the L2TP protocol
- The PPP session

♦ Commands associated with tunnels and the L2TP Protocol

These commands are used to configure L2TP tunnels. For additional information on the syntax of the commands listed below, please refer to the L2TP commands section in the Command Line Interface Reference chapter.

L2TP tunnel entry

l2tp add <*TunnelName*>

The remote tunnel host name

l2tp set remoteName <*name*> <*TunnelName*>

The local tunnel host name

l2tp set ourTunnelName <*name*> <*TunnelName*>

CHAP Secret

l2tp set CHAPSecret <*secret*> <*TunnelName*>

Tunnel Authentication

l2tp set authen on|off <*TunnelName*>

Type of L2TP support for tunnel

A tunnel entry can be configured to act as a LAC, an LNS, both a LAC and LNS, or disabled.

l2tp set type all|lns|l2tpclient|disabled <*TunnelName*>

Remote tunnel IP address

l2tp set address <*ipaddr*> <*TunnelName*>

Note: Verify that the IP address of the other end of the tunnel is correctly routed. It should not be routed through the tunnel itself, but over a physical link.

Our PPP system name and secret/password

The following commands specify the router's name and password/secret for authentication purposes on a per-tunnel basis.

Note: For more information on names and password usage, refer to the *Names and Passwords Rules* section, found later in this document.

l2tp set ourSysName <*name*> <*TunnelName*>

l2tp set ourPassword <*password*> <*TunnelName*>

Miscellaneous commands

Commands used to delete a tunnel, close a tunnel, or set up advanced L2TP configuration features such as traffic performance fine-tuning are discussed in the L2TP command section of the Command Line Interface Reference chapter.

PPP Session Configuration

Two commands are used to extend a PPP link from a remote site to a corporate site across the Internet and establish a tunnel. For additional information on the syntax of the commands listed below, please refer to the Command Line Interface Reference chapter, in the Remote command section.

remote setLNS <*TunnelName*> <*remoteName*>

remote setl2tpclient <*TunnelName*><*remoteName*>

Sample Configurations

Two sample configurations are described in this section:

- A simple configuration. This example describes the information needed to configure one side of the tunnel (the client side).
- A complete configuration. This example describes the information needed to configure both sides of the tunnel (client and server sides).

Simple L2TP Client Configuration Example

This example shows how a telecommuter working at home (client side) would configure his/her router SOHO to tunnel to the company's LAN (server side).

The information given in the Configuration Process section below provides a framework reference for this type of L2TP Client configuration.

♦ Assumptions

In this example, the following information is assumed:

- The server side (the company) has an LNS router connected to the Internet.
- The client side has an existing route to the Internet with the remote "internet" (Refer to Note 1, if you need sample configuration commands).
- IP routing is enabled (Refer to Note 1, if you need sample configuration commands).

Note: Below is an example of configuration commands that would be used to enable IP routing and establish a route to the Internet.

```
remote add internet
remote disauthen internet
remote setoursysname name_isp_expects internet
remote setourpass secret_isp_expects internet
remote addiproute 0.0.0.0 0.0.0.0 1 internet
remote setphone isdn 1 5551000 internet
remote setphone isdn 2 5553000 internet
eth ip enable
eth ip address 192.168.254.254 255.255.255.0
```

♦ Configuration Process

The following sets of questions, answers, and configuration commands specific to the L2TP tunnel and the PPP remote will assist you in configuring the client side router SOHO (also referred to as home router). Note that the server side is referred to as either company router or router at work.

L2TP tunnel configuration

L2TP tunnel-specific questions

1. What is the host name of the router at home that the user is configuring?
2. What is the host name of the company router at work to which the user will tunnel?
3. What is the shared CHAP secret used for tunneling between the home router (client) and the company router (server)?
4. What is the IP address of the company router to which the user will tunnel?

L2TP tunnel answers

For our example, let's assume the answers to the above Tunnel-Specific Questions are as follows:

1. Home_Router
2. Work_Router
3. Shared_Secret
4. 10.0.0.1

L2TP tunnel configuration commands

These commands would be used to set up the L2TP tunnel information for our example:

```
l2tp add Work_Router
l2tp set ourtunnel Home_Router Work_Router
l2tp set chapsecret Shared_Secret Work_Router
l2tp set address 10.0.0.1 Work_Router
```

PPP remote configuration

PPP remote-specific questions:

1. What is the home router's name for PPP authentication?

2. What is the home router's secret for PPP authentication?

3. Does the home router need PPP authentication for the remote router (company router)?

   If yes:

   a. What is the remote router's name for PPP authentication?
   b. What is the remote router's secret for PPP authentication?

   If no:

   a. Use the command **remote disauthen** <*remoteName*> where <*remoteName*> is the name used to refer to the company's router.

4. Does the remote router dynamically assign an IP address for this PPP session?

   If yes:

   a. Use IP address translation (NAT)

   If no and the home router is to behave as a LAN at home:

   a. Which IP address and network mask does the home router use for its LAN at home? Use the **eth ip addr** command to set the LAN at home. Do not enable IP address translation (NAT) for the remote (company) router.

   If no and the home router is to behave as a host at home:

   b. Which IP address does it use at home? Assuming an IP address of www.xxx.yyy.zzz, use the command:

   remote setsrcipaddr www.xxx.yyy.zzz 255.255.255.255 <*remoteName*>

   remote setiptranslate on <*remoteName*>

5. Which IP and network addresses does the home router access at work through this PPP session?

PPP remote answers

For our example, let us assume the answers to the above PPP Remote-Specific Questions are as follows:

1. ppp_soho

2. ppp_soho_secret

3. We assume that this router will authenticate the router at work with the following information:

   a) the company router's name is: ppp_work

   b) the company router's PPP secret is: ppp_work_secret

4. We assume that the company's router will dynamically assign an IP address to the home router.

5. 172.16.0.0/255.240.0.0

PPP remote configuration commands

For our example, these commands would be used to set up the PPP remote information for tunneling to work:

```
remote add ppp_work
```

```
remote setlns Work_Router ppp_work
remote setpasswd ppp_work_secret ppp_work
remote setiptranslate on ppp_work
remote addiproute 172.16.0.0 255.240.0.0 1 ppp_work

l2tp set oursysname ppp_soho Work_Router
l2tp set ourpassword ppp_soho_secret Work_Router
```

Complete LNS and L2TP Client Configuration Example

The following provides a configuration example of an LNS and L2TP Client.

♦ Assumptions

IP Addresses

The LNS server's LAN IP address is 192.168.100.1 (**LNSserver**) with a mask of 255.255.255.0.

The LNS has a WAN IP address of 192.168.110.1, which is used as the tunnel endpoint.

The LNS connects to the remote **internet**.

The L2TP Client's LAN IP address is 192.168.101.1 (**soho**) with a mask of 255.255.255.0. Additionally, 192.168.101.1 is also the tunnel endpoint within the L2TP client. The router **soho** connects to the remote **isp**.

Secret/password

A shared tunnel secret of "tunnelsecret" will be used.

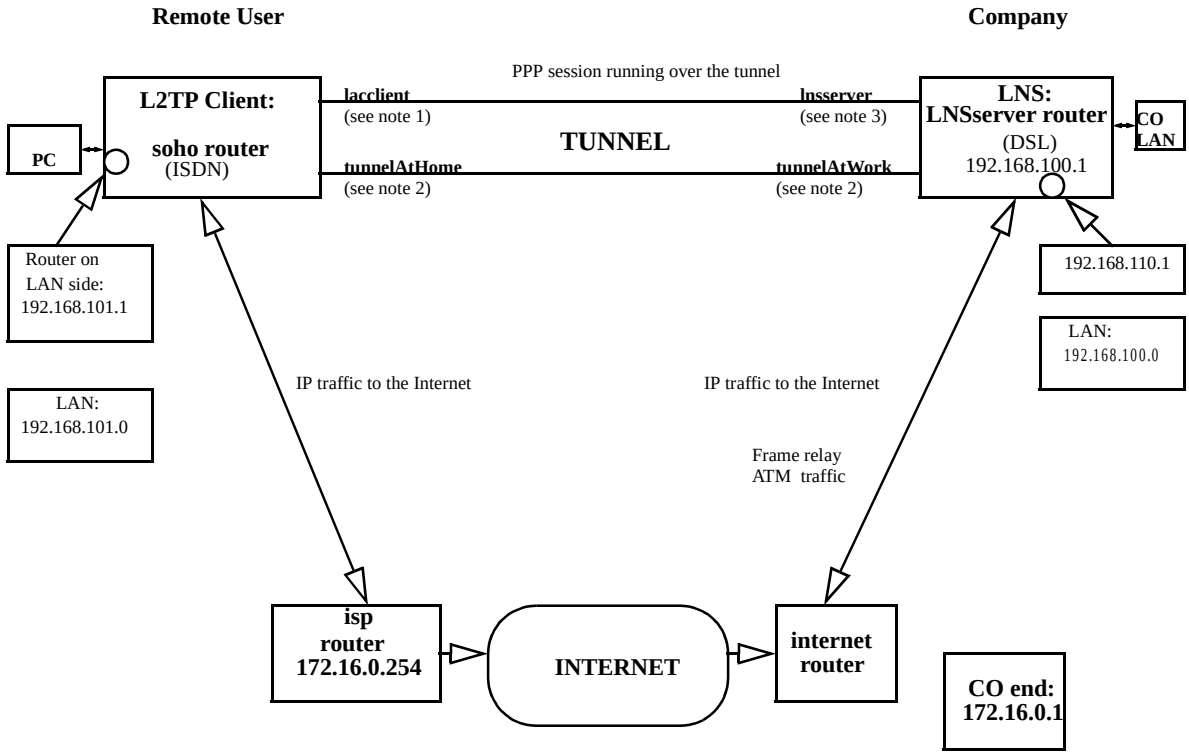
PPP Authentication

The LNS will authenticate the client using PPP. The client will not try to authenticate the LNS using PPP. For PPP authentication, the L2TP client will be known as "lacclient" with a password of "clientpassword".

Tunnel

Only the L2TP client (**soho**) will initiate the tunnel and make the connection. The tunnel is routed through the remote **internet** which is the default route. The LNS server never calls the L2TP client (**soho**).

Figure 2



Note 1: The CHAP secret is “clientPassword”.

Note 2: The CHAP secret is “tunnelSecret”.

Note 3: No CHAP secret is needed; the client does not authenticate the LNS server.

Configuration Process

The following sample scripts list the commands used to configure the routers **soho** (L2TP client), **LNSserver** (LNS), **internet**, and **isp**.

♦ Configuration commands for soho (L2TP client)

Note: soho is an ISDN router.

Define soho:

```
system name soho
system passwd sohopasswd
system msg configured_12/15/98
system securitytimer 60
```

Enable IP routing for soho:

```
eth ip enable
eth ip addr 192.168.101.1 255.255.255.0
```

Set up ISDN parameters:

```
isdn set switch ni1
isdn set dn 5551000 5553000
isdn set spids 0555100001 0555300001
```

Define DHCP settings for DNS servers, domain, wins server:

```
dhcp set value DOMAINNAMESERVER 192.168.100.68
dhcp set value DOMAINNAME flowpoint.com
dhcp set value WINSSERVER 192.168.100.73
```

Define a remote for the tunnel:

```
remote add lnsserver
remote disauthen lnsserver
remote setoursysname lacclient lnsserver
remote setourpasswd clientpassword lnsserver
remote setLNS tunnelAtWork lnsserver
remote addiproute 192.168.100.0 255.255.255.0 1 lnsserver
```

Define a remote isp:

```
remote add isp
remote setphone isdn 1 5552000 isp
remote setphone isdn 2 5554000 isp
remote disauthen internet remote addiproute 0.0.0.0 0.0.0.0 1 isp
```

Define the tunnel:

```
l2tp add tunnelAtWork
l2tp set chapsecret tunnelsecret tunnelAtWork
l2tp set ourtunnelname tunnelAtHome tunnelAtWork
l2tp set address 192.168.110.1 tunnelAtWork
save
reboot
```

♦ Configuration commands for internet

Note: internet is a DSL router. The router internet establishes a link to the LNS.

Define internet:

```
system name internet
system passwd internet
system msg configured_12/15/98
system securitytimer 60
```

Enable IP routing and add routes:

```
eth ip enable
eth ip addr 172.16.0.1 255.255.255.0
eth ip opt rxdef off
eth ip addroute 192.168.101.1 255.255.255.0 172.16.0.254 1
```

Create a DHCP pool of addresses:

```
dhcp add 172.16.0.0 255.255.255.0
dhcp del 192.168.254.0
dhcp set addr 172.16.0.2 172.16.0.20
```

Set up DSL parameters:

```
sd term co sd speed 1152
```

Define a remote LNSserver

```
remote add lnsserver
remote setauthen chap lnsserver
remote setpasswd serverpassword lnsserver
remote addiproute  192.168.110.1 255.255.255.255 1 lnsserver
remote setprotocol ppp lnsserver
remote setpvc 0*38 lnsserver
save
reboot
```

♦ Configuration commands for isp

Note: isp is an ISDN router. The router soho calls the router isp.

Define isp:

```
system name isp
system passwd isppasswd
system msg configured_12/15/98
system securitytimer 60
```

Enable IP routing:

```
eth ip enable
eth ip addr 172.16.0.254 255.255.255.0
```

Add a route to the other end of internet:

```
eth ip defgate 172.16.0.1
eth ip opt txdef off
```

Disable DHCP:

```
dhcp disable all
```

Set up ISDN parameters:

```
isdn set switch ni1
isdn set dn 5552000 5554000
isdn set spids 0555200001 0555400001
```

Define a remote (soho):

```
remote add soho
remote setauthen chap soho
remote setpassw sohopasswd soho
remote setphone isdn 1 5551000 soho
remote setphone isdn 2 5553000 soho
remote addiproute  192.168.101.0 255.255.255.0 1 soho
save
reboot
```

♦ Configuration commands for LNSserver

Note: LNSserver is a DSL router.

Define LNSserver:

```
system name lnsserver
system passwd serverpassword
system msg Script_for_LNS_called_HQ
system securitytimer 60
```

Enable IP routing:

```
eth ip enable
eth ip addr 192.168.100.1 255.255.255.0
```

Define DHCP settings for DNS servers, domain:

```
dhcp set value domainname flowpoint.com
dhcp set value domainnameserver 192.168.100.68
```

Set up DSL parameters:

```
sd speed 1152
```

Define a remote for the Tunnel:

```
remote add lacclient
remote setpass clientpassword lacclient
remote setLAC tunnelAtHome lacclient
remote setauthen chap lacclient
remote addiproute 192.168.101.0 255.255.255.0 1 lacclient
```

Define a remote (internet):

```
remote add internet
remote setphone isdn 1 5552000 internet
remote setphone isdn 2 5554000 internet
remote setauthen chap internet
remote setpasswd internet internet
remote addiproute 0.0.0.0 0.0.0.0 1 internet
remote setsrcipaddr 192.168.110.1 255.255.255.255 internet
remote addiproute 192.168.101.1 255.255.255.255 1 internet
remote setprotocol ppp internet
remote setpvc 0*38 internet
```

Define the actual tunnel:

```
l2tp add tunnelAtHome
l2tp set chapsecret tunnelsecret tunnelAtHome
l2tp set ourtunnelname tunnelAtWork tunnelAtHome
save
reboot
```

Chapter 5. Command Line Interface Reference

Command Line Interface Conventions

Command Input

The router Command Line Interface follows these conventions:

- Command line length may be up to 120 characters long.
- The Command Line Interface is not case-sensitive except for passwords and router names.
- Items that appear in bold type must be typed exactly as they appear.
  However, commands can be shortened to just those characters necessary to make the command unique.
- Items that appear in italic are placeholders representing specific information that you supply.
- Parameters in between characters < and > must be entered.
- Parameters in between characters [and] are optional.
- All commands are positional; i.e. each keyword/parameter must be entered in the order displayed.

Command Output

After execution of most commands, the system will return either of the following command prompts:

when you are logged in as an administrator, to indicate the end of command execution.

> to indicate the end of command execution when not logged in

Sample responses are shown in this chapter. In many cases, only the command prompt is returned. If you have not entered the correct parameters, the syntax of the command is displayed.

Command Organization

The commands are organized as follows:

- **System-level commands**
- **Router configuration commands:**
  system
  eth
  remote
  adsl
  atm
  eth (specific to the Dual Ethernet router)
  hdsl
  isdn
  sdsl

dhcp
l2tp
filters
save
erase

- **File system commands**

? or HELP

Lists the commands at the current level as well as subcommands. At the lowest level of the subcommand, entering a **?** may return the syntax of the command. Note that some commands require a character string and the **?** will be taken as the character string if entered in that position.

? or **help**

Example: `# ?`

Top-level commands:

```
?                   help              version
filter              login             logout
exit                reboot            mem
ps                  copy              dir
delete              rename            format
sync                msf               ifs
ipifs               iproutes          ipxroutes
ipxsaps             bi                system
eth                 save              erase
remote              call              ping
dhcp                atm               execute
l2tp                arp               tcp stats
```

System Level Commands

These commands are online action and status commands. They allow you to perform the following functions:

- log into and log out of configuration update mode
- display the router's configuration, the version and level numbers
- list running tasks, memory, communication interfaces
- connect to a remote router to test the line
- list IP routes, and IPX routes and SAPs, root bridge
- save the new configuration image
- reboot the system

ARP DELETE

Deletes the IP address of the entry in the ARP table.

arp delete <*ipaddr*>|all

ipaddr IP address in the format of 4 decimals separated by periods.

all Deletes all existing arp table entries

Example: `arp delete 128.1.2.0`

ARP LIST

Lists ARP table entries in an IP routing environment. ARP (Address Resolution Protocol) is a tool used to find the appropriate MAC addresses of devices based on the destination IP addresses.

arp list <*ipaddr*><*InterfaceName*> <*InterfaceUnit*>

ipaddr IP address associated with a MAC address for a device on the local interface in the format of 4 decimals separated by periods.

InterfaceName MAC address on the local network

InterfaceUnit For an Ethernet interface, can be a 1 or 0. For a DSL interface, this is a VPN number.

Example: `arp list`

Response:

```
IP Addr              Mac Address            Interface
192.84.210.148       00:05:02:00:80:A8      ETHERNET/0
```

BI

Lists the root bridge.

bi

Response:

```
# bi
GROUP 0Our ID=8000+00206f0249fc Root ID=8000+00206f0249fc
Port ETHERNET/0           00+00 FORWARDING
```

BI LIST

Lists MAC addresses and corresponding bridge ports as learned by the bridge function. This list includes several flags and the number of seconds elapsed since the last packet was received by the MAC address.

bi list

Response:

```
# bi list
BRIDGE GROUP 0:
00206F0249FC: P    US  SD  A
0180C2000000: P            A                MC
FFFFFFFFFFFF: P FLD        A                BC
00206F024A4F: ETHERNET/0                    1            FWD
00A024C6C594: ETHERNET/0                    1            FWD
00206F200008: ETHERNET/0                    1            FWD
0020AFC5697F: ETHERNET/0                    11           FWD
```

CALL

Dials a remote router. This command can be used to test the ISDN link and the remote router configuration settings.

call <*remoteName*>

Response:

```
# Request Queued
```

EXIT

Has the same function as **logout**, but will disconnect you from a Telnet session.

exit

IFS

Lists the communications interfaces installed in the router and the status of the interfaces.

ifs

Response:

```
# ifs
Interface     Speed       In %      Out %  Protocol    State     Connection
ETHERNET/0    10.0mb      0%/0%     0%/0%  (Ethernet)  OPENED
ATM_VC/1      25.6mb      0%/0%     0%/0%  (CLEAR)     OPENED    to HQ
ATM-25/0      25.6mb      0%/0%     0%/0%  (ATM)       OPENED
CONSOLE/0     9600 b      0%/0%     0%/0%  (TTY)       OPENED
```

IPIFS

Lists the IP interface.

ipifs

Response:

```
ATM_VC/1      192.168.254.1 (FFFFFF00) dest 192.168.254.2 sub 192.168.254.0
              net 192.168.254.0 (FFFFFF00) P-2-P
ETHERNET/0    192.84.210.12 (FFFFFF00) dest 0.0.0.0 sub 192.84.210.0
              net 192.84.210.0 (FFFFFF00) BROADCAST
```

IPROUTES

Lists the current entries in the IP routing table.

iproutes

Response:

```
# iproutes
IP route        /   Mask    -->  Gateway       Interface     Hops Flags

0.0.0.0         /ffffffff -->  0.0.0.0       [none]        0 NW PRIV
192.84.210.0    /ffffff00 -->  0.0.0.0       ETHERNET/0    1 NW FW DIR PERM
192.84.210.12   /ffffffff -->  0.0.0.0       ETHERNET/0    0 ME
192.168.254.0   /ffffff00 -->  0.0.0.0       [none]        0 NW PRIV
192.168.254.1   /ffffffff -->  HQ            ATM_VC/1      0 ME
192.168.254.2   /ffffffff -->  HQ            ATM_VC/1      1 FW DIR PRIV
224.0.0.9       /ffffffff -->  0.0.0.0       [none]        0 ME
255.255.255.255/ffffffff -->  0.0.0.0       [none]        0 NW PERM
```

where:	NW	Network
	PERM	Permanent (static)
	DOD	Initiate Link dial-up
	FW	Forward
	DIR	Direct
	ME	This Router

IPXROUTES

Lists the current entries in the IPX routing table.

ipxroutes

Response:

```
# ipxroutes
Network      Gateway     Interface      Hops Ticks Flags
00001001:    HQ            [down]       1      4      STATIC FORWARD DOD
00000456:    (DIRECT)    ETHERNET/0     0      1      FORWARD
```

where:	STATIC	Static Route
	DOD	Initiate Link dial-up
	FORWARD	
	DIRECT	

IPXSAPS

Lists the current services in the IPX SAPs table.

ipxsaps

Response:

```
# ipxsaps
Service Name             Type   Node number Network   Skt  Hops
SERV312_FP                 4    000000000001:00001001:045   1
```

LOGIN

Login is required whenever you intend to change any configuration settings or save an entire new configuration.

login <*password*>

password Mandatory password set using the **system admin** command or default (**admin**). If not specified, you will be shown the command syntax. The password is case sensitive.

Response: Logged in successfully!

or

```
Wrong password! Try logging in again.
```

After successfully logging in, the ‘#’ is used as the prompt character to indicate that you are logged in as an administrator.

LOGOUT

Logs out to reinstate administrative security after you have completed changing the router's configuration.

logout

MEM

Lists memory and buffer usage.

mem

Response:

```
# mem
Small buffers used.......18  (7% of 256 used)
Large buffers used.......41  (16% of 256 used)
Buffer descriptors used..59  (7% of 768 used)
Number of waiters s/l.... 0/0

Table memory allocation statistics:
Sizes     16    32    64   128   256   512  1024  2048
Used      34    18    12     3     8     9     8     7
Free       3     1     4     0     1     1     1     1

Sizes   4096  8192
Used       3     1
Free       1     0
Total in use: 51936, total free: 857368 (8272 + 849096)
```

MLP SUMMARY

Lists the status of the any protocols negotiated for an active remote connection. The following lists the most common protocols:

- MLP (PPP Link Protocol)
- IPNCP (IP routing Network Protocol)
- CCP (Compression Network Protocol)
- BNCP (Bridging Network Protocol)
- IPXCP (IPX Network Protocol)

"Open" indicates that the protocol is in ready state.
"Stopped" means that the protocol is defined but did not successfully negotiate with the remote end.
No message means that the link is not active.

mlp summary

Example: `mlp summary`

PING

An echo message, available within the TCP/IP protocol suite, sent to a remote node and returned; it is used to test connectivity to the remote node and is particularly useful for locating connection problems on a network.
By default, the router will try to “ping” the remote device for five consecutive times and will issue status messages.

ping [**-c count**] [**-i wait**] [- **s size** (or **-l size**)] <*ipaddr*>

-c count — Number of packets; count is a value between 1 and 10.

-i wait — Wait period in seconds between packets; wait is a value between 1 and 10.

-s size — Packet data length “size” bytes; size is a value between 0 and 972.

-l size — Same as -s size

ipaddr — IP address in the format of 4 decimals separated by periods.

Example:
```
ping -c  8  -i  7 -s 34 192.168.254.2
```

Response:
```
ping: reply from 192.168.254.2: bytes=34 (data), time<5 ms
ping: reply from 192.168.254.2: bytes=34 (data), time<5 ms
ping: reply from 192.168.254.2: bytes=34 (data), time<5 ms
ping: reply from 192.168.254.2: bytes=34 (data), time<5 ms
ping: reply from 192.168.254.2: bytes=34 (data), time<5 ms
ping: reply from 192.168.254.2: bytes=34 (data), time<5 ms
ping: reply from 192.168.254.2: bytes=34 (data), time<5 ms
ping: reply from 192.168.254.2: bytes=34 (data), time<5 ms
ping: packets sent 8, packets received 8
```

PS

Lists all of the tasks (processes) running in the system and the status of the tasks.

ps

Response:
```
# ps
TID:          NAME         FL P BOTTOM CURRENT SIZE
1:IDLE                     02 7 1208f0 121008  2032
3:MSFS_SYNC                03 6 1224a0 122ba8  2032
4:SYSTEM LOGGER            03 5 122cd0 1233d8  2032
5:LL_PPP                   03 5 126750 126e58  2032
6:NL_IP                    03 5 126fe0 1272e0  1000
7:TL_IP_UDP                03 3 127460 127768  1000|
8:TL_IP_TCP                03 3 1278c0 127fd0  2032
9:IP_RIP                   03 4 128120 128420  1000
10:TELNETD                 03 5 128550 128838  1000
11:DUM                     03 5 12b580 12bc88  2032
12:ATM25                   03 1 12c0a0 12c790  2032
13:SNMPD                   03 5 124b60 125a70  4080
14:BOOTP                   03 5 12e3d0 12e6c0  1000
15:CMD                     01 6 12cba0 12d9f8  4080
```

TID:	task ID field
NAME:	name of the task
FL:	flag field
P:	number from 1 to 7 with the highest priority equal to 1.
BOTTOM:	address of the task stack
CURRENT:	current stack pointer
SIZE:	stack size in byte

REBOOT

This command causes a reboot of the system. You must perform a reboot after you have configured the router the first time or when you modify the configuration. Reboot is *always* required when the following configuration settings are modified:

- System Settings Ethernet IP Address
- Ethernet IPX Network Number
- TCP/IP and IPX Routing
- Remote Router Default Bridging Destination
- TCP/IP Route Addresses
- IPX Routes
- SAPs and Bridging

Reboot is also required when adding a new remote entry in the remote database.

Reboot also ensures that all file system updates are completed. There is a time lag between the **save...** commands and the time the data is safely stored in FLASH memory. If the power goes off during this time, data can be lost. Always reboot before powering off the router. Alternatively, use the **sync** command.

Caution: This command erases all of the configuration data in the router.

reboot [default]

default	This option deletes the system configuration file, and restores the router to its original defaults (before any configuration was entered). **Note:** *Default* must be fully spelled out.

TCP STATS

Displays the TCP statistics and open connections.

tcp stats

Example: `tcp stats`

VERS

Displays the software version level, source, software options, and amount of elapsed time the router has been running.

vers

Response:

```
FlowPoint/2025 ATM25 Router
FlowPoint-2000 BOOT/POST V3.0.0 (12-Dec-98 18:10)
Software version 3.0.1 (built Wed Jan 7 13:17:37 PST 1999
18:36:15 PST 1999
Maximum users: unlimited
Options: ATM25, IP, ~IP FILTERING, IP TRANS, HOST MAPPING, DHCP, ~L2TP,
~ENCRYPT, BRIDGE, IPX
Up for 0 days 0 hours 20 minutes (started 1/7/1999 at 13:28)
```

Note: Features present in the firmware, but not yet enabled, are preceded by a "~". These features can be activated by purchasing a software key from your distributor**.**

Router Configuration Commands

Configuration commands are used to set configuration information for each functional capability of the router. Each functional capability has a specific prefix for its associated commands:

- **system**: target router system commands
- **eth ip**: Ethernet IP routing commands
- **remote**: remote router database commands
- **adsl:** Asymmetric Digital Subscriber Line commands (ADSL routers only)
- **atm:** Asynchronous Transfer Mode commands (ATM routers only)
- **eth** Dual Ethernet router commands only
- **hdsl:** High-speed Digital Subscriber Line commands (HDSL routers only)
- **isdn (for IDSL):** ISDN Digital Subscriber Line (IDSL routers only)
- **sdsl:** Symmetric Digital Subscriber Line commands (SDSL routers only)
- **dhcp:** Dynamic Host Configuration Protocol commands
- **l2tp** Layer 2 Tunneling Protocol commands
- **save**: save configuration to FLASH memory commands
- **filter**: filtering commands
- **? or help**: summary of available commands

Target Router System Configuration Commands (SYSTEM)

The following commands set basic router configuration information:

- name of the router
- optional system message
- authentication password
- security authentication protocol
- management security
- system administration password
- IP address translation
- NAT configuration
- Host mapping
- WAN-to-WAN forwarding
- filters

SYSTEM ?

Lists the supported keywords.

system ?

Response:

```
?                   msg                 name
passwd              authen              community
list                admin               history
log                 addServer           addHostMapping
delServer           delHostMapping      addTelnetFilter
delTelnetFilter     addSNMPFilter       delSMNPFilter
bootpServer         supportTrace        telnetport
snmpport            wan2wanforwarding   addUDPrelay
delUDPrelay         securityTimer       oneWANdialup
addHTTPFilter       delHTTPFilter
```

SYSTEM ADDHOSTMAPPING

This command is used to remap a range of local-LAN IP addresses to a range of public IP addresses on a system-wide basis. These local addresses are mapped one-to-one to the public addresses.

Note: The range of public IP addresses is defined by <first public addr> only. The rest of the range is computed automatically (from <*first public addr*> to <*first public addr*> + number of addresses remapped - 1) inclusive.

system addHostMapping <*first private addr*> <*second private addr*> <*first public addr*>

first private addr	First IP address in the range of IP address to be remapped, in the format of 4 decimals separated by periods.
second private addr	Last address in the range of IP address to be remapped, in the format of 4 decimals separated by periods.
first public addr	Defines the range of public IP addresses, in the format of 4 decimals separated by periods. The rest of the range is computed automatically.
Example:	`system addHostMapping 192.168.207.40 192.168.207.49 10.1.1.7`

SYSTEM ADDHTTPFILTER

This command is used to allow devices within the defined IP address range to use the HTTP protocol (for example, to browse the Web). This command is useful to block devices on the WAN from accessing the Web browser.

system addHTTPFilter <*first ip addr*> [<*last ip addr*>] | LAN

first ip addr	First IP address of the range
last ip addr	Last IP address of the range. May be omitted if the range contains only one IP address.
LAN	Local Ethernet LAN
Example:	`system addHTTPFilter 192.168.1.5 192.168.1.12`

SYSTEM ADDSERVER

This Network Address Translation (NAT) command is used to configure a local IP address as the particular server on the LAN (FTP, SMTP, etc.) for the global configuration.

system addServer <*ipaddr*>|discard|me <*protocolid*> |tcp|udp <*first port*> |ftp|telnet|smtp|snmp|http [<*last port*> [<*first private port*>]]

ipaddr	IP address of the host selected as server in the format in the format of 4 decimals separated by periods.
discard	Used to discard the incoming server request.
me	Used to send the incoming server request to the local router, regardless of its IP address.
protocolid	Protocol used by the selected server; can be **tcp** or **udp,** or a numeric value.
first port	First or only port as seen by the remote end**.** Port used by the selected server; can be as string such as ftp, telnet, smtp, snmp, or http, or a numeric value between 0 and 65,535. A numeric value of 0 will match any port.

last port If specified, is used with <first port> to specify a range of ports as seen by the remote end for the server on the LAN.

first private port If specified, is a port remapping of the incoming request from the remote end.

Example: `system addServer 192.168.1.5 tcp smtp`

SYSTEM ADDSNMPFILTER

This command is used to validate SNMP clients by defining a range of IP addresses that are allowed to access the router via SNMP. This validation feature is **off** by default.

Note 1: This command does not require a reboot and is effective immediately.

Note 2: To list the range of allowed clients, use the command **system list** when logged in with read and write permission (log in with password).

system addSNMPFilter <*first ip addr*> [<*last ip addr*>] | LAN

first ip addr First IP address of the client range

last ip addr Last IP address of the client range. May be omitted if the range contains only one IP address.

LAN Local Ethernet LAN

Example: `system addsnmpfilter 192.168.1.5 192.168.1.12`

SYSTEM ADDTELNETFILTER

This command is used to validate Telnet clients by defining a range of IP addresses that are allowed to access the router via Telnet. This validation feature is **off** by default.

Note 1: This command does not require a reboot and is effective immediately.

Note 2: To list the range of allowed clients, use the command **system list** when logged in with read and write permission (log in with password).

system addTelnetFilter <*first ip addr*> [<*last ip addr*>] | LAN

first ip addr First IP address of the client range

last ip addr Last IP address of the client range. May be omitted if the range contains only one IP address.

LAN Local Ethernet LAN

Example: `system addtelnetfilter 192.168.1.5 192.168.1.12`

SYSTEM ADDUDPRELAY

This command is used to create a UDP port range for packet forwarding. You can specify a port range from 0 to 65535. 137 to 139 are reserved for NetBIOS ports. Overlap of UDP ports is not allowed.

system addUDPrelay <*ipaddr*> <*first port*>|all [<*last port*>]

ipaddr IP address of the server to which the UDP packet will be forwarded.

first port First port in the UDP port range to be created.

all Incorporates all the available UDP ports in the new range

last port Last port in the UDP port range to be created

Example: `system addudprelay 192.168.1.5 all`

SYSTEM ADMIN

Sets the administration password used to control write access to the target router configuration.

system admin <*password*>

password Write-enable login password

Example: `system admin adx1lp`

SYSTEM AUTHEN

Forces the target router authentication protocol used for security negotiation with the remote routers when setting the local side authentication. You should not need to issue this command as the best security possible is provided with the **none** default.

system authen none | pap | chap

none When set to **none** (the default), the authentication protocol is negotiated, with the minimum best security level as defined for each remote router in the database.

pap When set to **pap**, negotiation will begin with PAP (instead of CHAP) for those entries that have PAP in the remote database and only when the call is initiated locally.

chap Overrides all the remote database entries with **chap**; i.e., only CHAP will be performed.

Example: `system authen CHAP`

SYSTEM BOOTPSERVER

Lets the router relay BootP or DHCP requests to a DHCP server on the WAN, when a PC attempts to acquire an IP address using DHCP. This command disables the router's DHCP server.

system bootpServer <*ipaddr*>

ipaddr IP address of the target router in the format of 4 decimals separated by periods.

Example: `system bootpserver 128.1.210.64`

SYSTEM COMMUNITY

This command is used to enhance SNMP security. It allows the user to change the SNMP community name from its default value of “public” to a different value. Refer to *Management Security*, page 92.

Note: The command **system community** (with no value) will display the current community name.

system community [<*SNMP community name*>]

SNMP community name String of up to 40 characters

Example: `system community fred`

Example: `system community`

SYSTEM DELHOSTMAPPING

This command is used to undo an IP address/ host translation (remapping) range that was previously established with the command **remote addHostMapping** on a per-systemwide basis.

system delHostMapping <*first private addr*> <*second private addr*> <*first public addr*>

first private add First IP address in the range of IP address, in the format of 4 decimals separated by periods.

second private addr Last address in the range of IP address, in the format of 4 decimals separated by periods.

first public addr Defines the range of public IP addresses, in the format of 4 decimals separated by periods.

The rest of the range is computed automatically.

Example: `system delHostMapping 192.168.207.40 192.168.207.49 10.1.1.7`

SYSTEM DELHTTPFILTER

This command is used to delete an IP address range created by the **system addHTTPFilter** command.

system delHTTPFilter <*first ip addr*> [<*last ip addr*>] | LAN

first ip addr First IP address of the range

last ip addr Last IP address of the range. May be omitted if the range contains only one IP address.

LAN Local Ethernet LAN

Example: `system delHTTPFilter 192.168.1.5 192.168.1.12`

SYSTEM DELSERVER

This Network Address Translation (NAT) command is used to delete an entry created by the **system addServer** command.

system delServer <*ipaddr***>**| discard|me <*protocolid***>** |tcp|udp <f*irst port*> |ftp|telnet|smtp|snmp|http [<*last port*> [<*first private port*>]] <*remoteName*>

ipaddr	IP address of the host selected as server in the format of 4 decimals separated by periods
discard	Used to discard the incoming server request.
me	Used to send the incoming server request to the local router, regardless of its IP address.
protocolid	Protocol used by the selected server; can be **tcp** or **udp.**
first port	First or only port as seen by the remote end**.** Port used by the selected server; Can be as string such as ftp, telnet, smtp, snmp, or http, or a numeric value between 0 and 65,535. A numeric value of 0 will match any port
last port	If specified, is used with <first port> to specify a range of ports as seen by the remote end for the server on the LAN.
first private port	If specified, is a port remapping of the incoming request from the remote end.

Example: `system delServer 192.168.1.5 tcp smtp`

SYSTEM DELSNMPFILTER

This command deletes the client range previously defined by the command **system addsnmpfilte**r.

Note 1: This command does <u>not</u> require a reboot and is effective immediately.

Note 2: To list the range of allowed clients, use the command system list when logged in with read and write permission (log in with password).

system delSNMPFilter <*first ip addr*> [<*last ip addr*>] | LAN

first ip addr	First IP address of the client range
last ip addr	Last IP address of the client range. May be omitted if the range contains only one IP address
LAN	Local Ethernet LAN

Example: `system delsnmpfilter 192.168.1.5 192.168.1.12`

SYSTEM DELTELNETFILTER

This command deletes the client range previously defined by the command **system addtelnetfilter**.

Note 1: This command does <u>not</u> require a reboot and is effective immediately.

Note 2: To list the range of allowed clients, use the command system list when logged in with read and write permission (log in with password).

system delTelnetFilter <*first ip addr*> [<*last ip addr*>] | LAN

first ip addr First IP address in the client range

last ip addr Last IP address in the client range. May be omitted if the range contains only one IP address.

LAN Local Ethernet LAN

Example: `system deltelnetfilter 192.168.1.5 192.168.1.12`

SYSTEM DELUDPRELAY

This command deletes the port range that was previously enabled by the command: **system addUDPrelay.**

system delUDPrelay <*ipaddr*> <*first port*>| *all [<last port>]*

ipaddr IP address of the server

first port First port in the UDP port range to be deleted

all Deletes all existing UDP ports

last port Last port in the UDP port range to be deleted

Example: `system deludprelay 192.168.1.5 all`

SYSTEM HISTORY

Displays the router's most recent console log.

system history

Example: `system history`

SYSTEM LIST

Lists the target router's system name, security authentication protocol, callerID and 'data as voice' status, and system message.

system list

Example: `system list`

Response:

```
GENERAL INFORMATION FOR <SOHO>

System started on..................... 1/7/1998 at 13:29
```

```
  Authentication override.............. NONE
WAN to WAN Forwarding................ yes
  BOOTP/DHCP Server address............ none
  Telnet Port.......................... default (23)
SNMP Port............................ default (161)
System message: Configured January 1998
```

SYSTEM LOG

Allows logging of the router's activity in a TELNET session.

system log start | stop | status

start — Used to monitor the router activity at all time

Example: `system log start`

stop — Use to discontinue the logging utility at the console

Example: `system log stop`

status — Used to find out if other users (yourself included) are using this utility

Example: `system log status`

SYSTEM MSG

Sets a message that is saved in the target router you are configuring.

system msg <*message*>

message — Message (character string) — Space characters are not allowed within the message; you may use underscore characters instead. If you do not enter a message, the current message is displayed. The message must be no more than 255 characters.

Example: `system msg Configured _on_ 10/21/98`

SYSTEM NAME

Sets the name for the target router that you are configuring. You are required to assign a name to the target router. This name is sent to a remote router during PAP/CHAP authentication.

system name <*name*>

name — Name of the target router (character string). Space characters are not allowed within the name; you may use underscore characters instead. (The system name is a "word" when exchanged with PAP/CHAP.) If you do not enter a name, the current name of the router is displayed. If you type anything after **system name**, the characters will be taken as the new name.

Note: The system name is case sensitive and must be no more than 50 characters.

Example: `system name Router1`

SYSTEM ONEWANDIALUP

This command is useful when security concerns dictate than the router can only have one connection active at one time. For example, a connection to the Internet and to another location such as one's company at the same time can be prevented. The command **system oneWANdialup** on forces the router to have at most ONE connection to a remote entry to be active at one time. (Multiple links to the same remote are allowed).

A connection is only brought up when data is received for forwarding to the remote router (dial on demand); the automatic bringing up of permanent links is disabled.

At system startup time, each remote entry is examined. If only ONE enabled remote is found, the remote is left enabled. If more than one enabled remote entry is found, then every entry which does not have a protocol of PPP or PPPLLC is disabled. The minimum number of active Links (remote minLink) is set to 0 on the enabled entries; otherwise, connections to multiple destinations would not be possible (since the link to the destination with minLink non-zero would be active).

Multiple connections to the SAME location are allowed; PPP Multi-link protocol is supported.

This command complements the system command controlling WAN-to-WAN forwarding. That command allows multiple connections to different locations to be active at the same time, but stops traffic from passing from one WAN connection to another.

system oneWANdialup on|off

on	Enables only one active connection at one time to a remote entry
off	Turns off **system oneWANdialup**

Example: `system oneWANdialup on`

SYSTEM PASSWD

Sets the target router system authentication password used when the router connects to other routers or is challenged by them. This password is a default password used for all remote sites, unless a unique password is explicitly defined for connecting to a remote router with the **remote setOurPasswd** command.

system passwd <*password*>

password	Authentication password of the target router.
	Note: The password is case-sensitive and should be no more than 40 characters.

Example: `system passwd chwgn1`

SYSTEM SECURITYTIMER

A Telnet or console user is automatically logged out of privileged mode when no typing has occurred for 10 minutes. This command allows the user to change the 10-minute default to a different value.

system securityTimer <*time in Minute*>

time in Minute Length of time in minutes
Auto logout can be disabled by setting the <time in minute> to zero.

Example: `system securityTimer 15`

SYSTEM SNMPPORT

This command is used to manage SNMP port access; this includes disabling SNMP, reestablishing SNMP services, or redefining the SNMP port for security reasons. Refer to *Advanced Features - Management Security* in Chapter 4.

Note: This command requires a save and reboot to take effect.

system snmpport default|disabled | <*port*>

default Restores the default values to 161

disabled Disables remote management.

port Used to define a new SNMP port number.

Use this option to redefine the SNMP port to a non well-known value to restrict remote access.

Example:
```
system snmpport default
system snmpport disabled
system snmpport 3333
```

SYSTEM SUPPORTTRACE

This command lets you capture to a file all of the configuration data that Technical Support may need to investigate configuration problems. This exhaustive list command incorporates the following commands:

- system history
- vers
- mem
- system list
- eth list
- dhcp list (if DHCP is enabled)
- remote list
- ifs
- bi (if bridging)
- ipifs
- iproutes

- ipxroutes

system supporttrace

Example: `system supporttrace`

SYSTEM TELNETPORT

The router has a built-in Telnet server. This command is used to specify which router's TCP port is to receive a Telnet connection.

Note: This command requires a save and reboot to take effect.

system telnetport default|disabled|<*port*>

default	The default value is 23.
disabled	The router will not accept any incoming TCP request.
port	Port number of the Ethernet LAN. It is recommended that this number be > 2048 if not 0 (disabled) or 23 (default).

Example:
```
system telnetport default
system telnetport disabled
system telnetport 3333`
```

SYSTEM WAN2WANFORWARDING

This command allows the user to manage WAN-to-WAN forwarding of data from one WAN link to another.

For example, if the router is used at home to access both a company network and the Internet at the same time, and it is desirable that company information not pass to the Internet, then disable WAN-to-WAN forwarding.

system wan2wanforwarding on|off

on	Used to allow data to be forwarded from one WAN link to another link.
off	Used to stop the data from being forwarded from one WAN link to another WAN link.

Example: `system wan2wanforwarding on`

Target Router Ethernet LAN Bridging and Routing (ETH)

The following commands allow you to:

- Set the Ethernet LAN IP address
- List the current contents of the IP routing table
- Enable and disable IP routing
- List or save the current configuration settings

All of these commands will require a reboot.

ETH ?

Lists the supported keywords.

eth ?

Examples:
```
eth ?
eth ip ?
```

Response:
```
Eth commands:
?                       ip          ipx
list

eth ip sub-commands
?                       addr        ripmulticast
options                 enable      disable
firewall                addroute    delroute
defgateway              filter      directbcast
```

ETH IP ADDR

Sets the IP address, subnet mask, and port number for the Ethernet LAN connection. After entering this command, Ethernet LAN IP routing is disabled.

eth ip addr <*ipaddr*> <*ipnetmask*> [<*port#*>]

ipaddr Ethernet LAN IP address, in the format of 4 decimals separated by periods.

ipnetmask IP network mask, in the format of 4 decimals separated by periods.

port# Port number of the Ethernet LAN. This number must be 0 (default)or 1, or may be omitted.

Example: `eth ip 128.1.2.0 255.255.255.0`

ETH IP ADDROUTE

Allows to define IP routes reached via the LAN interface. It is only needed if the system does not support RIP.

Note: This command requires a reboot.

eth ip addRoute <*ipaddr*> <*ipnetmask*> <*gateway*> <*hops*> [<port#>]

ipaddr Ethernet LAN IP address in the format of 4 decimals separated by periods.

ipnetmask IP network mask in the format of 4 decimals separated by periods.

gateway IP address in the format of 4 decimals separated by periods.

hops Number of routers through which the packet must go to get to its destination.

port# Port number of the Ethernet LAN; must be 0, or 1, or omitted.

Example: `eth ip addRoute 128.1.2.0 255.255.255.0 128.1.1.17 1`

ETH IP DEFGATEWAY

Lets you assign an Ethernet default gateway for packets that do not have a destination specified. This setting is most useful when IP routing is not enabled, in which case the system acts as an IP host (i.e. an end system, as opposed to an IP router).

Note: This command requires a reboot; it is also an alternative to:

```
eth ip addRoute 0.0.0.0 255.255.255.0 <gateway> 1
```

eth ip defgateway <*ipaddr*>[<*port#*>]

ipaddr Ethernet LAN IP address in the format of 4 decimals separated by periods.

port# Port number of the Ethernet LAN; must be 0, or 1, or omitted.

Example: `eth ip defgateway 128.1.210.65`

ETH IP DELROUTE

Used to remove IP routes reached via the LAN interface. It is only needed if the system does not support RIP.

Note: This command requires a reboot.

eth ip delRoute <*ipaddr*> <*ipnetmask*> [<port#>]

ipaddr Ethernet LAN IP address in the format of 4 decimals separated by periods.

ipnetmask IP network mask in the format of 4 decimals separated by periods.

port# Port number of the Ethernet LAN; must be 0, or 1, or omitted.

Example: `eth ip delRoute 128.1.2.0 255.255.255.0 128.1.1.17 1`

ETH IP DIRECTEDBCAST

This command is used to enable or disable the forwarding of packets sent to the network-prefix-directed broadcast address of an interface.

A network-prefix-directed broadcast address is the broadcast address for a particular network.
For example, a network's IP address is 192.168.254.0 and its mask is 255.255.255.0. Its network-prefix-directed broadcast address is 192.168.254.255.

eth ip directedbcast on | off

on Enables the forwarding of packets

off Disables the forwarding of packets

Example: `eth ip disable`

ETH IP DISABLE

Disables IP routing across the Ethernet LAN. This acts as a master switch allowing you to disable IP Routing for testing or control purposes.

Note: A reboot is required after this command.

eth ip disable [*port#*]

port# Port number of the Ethernet LAN. This number must be 0 or 1, or may be omitted.

Example: `eth ip disable`

ETH IP ENABLE

Enables IP routing across the Ethernet LAN. This acts as a master switch allowing you to enable IP routing.

eth ip enable [*port#*]

port# Port number of the Ethernet LAN. This number must be 0 or 1, or may be omitted.

Example: `eth ip enable`

ETH IP FILTER

This command is used to define an IP filter on the Ethernet interface of the connection. The filter is used to screen IP packets and operates at the interface level. Each interface is defined by 3 types of filters: Input, Forward, and Output filters. For more information on IP filters and Firewall, please refer *Configuring Special Features, IP Filtering - Chapter 4*.

eth ip filter ***<command> <type> <action> <parameters> [<port#>]***		
command	append <type> <action> <parameters>	Append a filter to the end of this <type>
	insert <type> <action> <parameters>	Insert a filter at the front of this <type>
	delete <type> <action> <parameters>	Delete the first filter matching this filter
	flush <type>	Delete all filters of this <type> from this interface
	check <type> <parameters>	Check the action to take (Accept, Drop, Reject) based on the parameters
	list <type>	List all filters of a <type> on this interface
	watch on | off	Print out a message to the console if a packet to or from this remote is dropped or rejected
type	input output forward	
action	accept drop reject	

parameters Each IP filter can have any combination of the following parameters used for matching against the IP packet. Below are the option/value pairs currently possible:

-p <protocol>|TCP|UDP|ICMP
where <protocol> is an IP protocol number or the string "TCP", "UDP", "ICMP".
If <protocol> is 0 (or the -p option is not specified), this IP filter will match ANY protocol.

-sa <first source ip addr>[:<last source ip addr>]
where <first source ip addr> defines the first or only source IP address and <last source ip addr>, if present, defines the last source IP address in a range. If not specified, <first source ip addr> is assumed to be 0.0.0.0, <last source ip addr> is assumed to be 255.255.255.255.

-sm <source ip mask>
where <source ip mask>, when present, defines a mask to use when comparing the <first source ip addr>...<last source ip addr> with the source IP address in the IP packet. If not specified, the source IP mask is set to 255.255.255.255.

-sp <first source port>[:<last source port>]
where <first source port> defines the first or only source port and <last source port>, if present, defines the last source port in a range. If not specified, the <first source port> is assumed to be 0, the <last source port> is assumed to be 0xffff.

--da <first dest ip addr>[:<last dest ip addr>]
where <first dest ip addr> defines the first or only destination IP address and <last dest ip addr>, if present, defines the last destination IP address in a range. If not specified, <first dest ip addr> is assumed to be 0.0.0.0, <last dest ip addr> is assumed to be 255.255.255.255.

-dm <dest ip mask>
where <dest ip mask>, when present, defines a mask to use when comparing the <first dest ip addr>...<last dest ip addr> with the destination IP address in the IP packet. If not specified, the destination IP mask is set to 255.255.255.255.

-dp <first dest port>[:<last dest port>]
where <first dest port> defines the first or only destination port and <last dest port>, if present, defines the last destination port in a range. If not specified, the <first dest port> is assumed to be 0, the <last dest port> is assumed to be 0xffff.

-b
This option indicates that this filter should be tested twice; a first time with the source filter information matched against the source information in the IP packet and the destination filter information matched against the destination information in the IP packet; and a second time with the source filter information matched against the destination information in the IP packet and the destination filter information matched against the source information in the IP packet.

-c <count of times rule used>
indicates how many IP packets have matched this filter since the router was rebooted.

-tcp syn|ack|noflag
where **syn** is the TCP SYN flag, **ack** is the TCP ACK flag, and **noflag** means there is a TCP packet AND neither the SYN flag or the ACK flag are set. This option is ignored if the IP packet is not a TCP packet. If not specified, the TCP SYN and TCP ACK flags are not checked when matching the IP packet with this filter.

Note: MORE than one **-tcp** option in an IP filter may be specified. For example, to match this IP filter against the initiation of a TCP connection, **-tcp syn** would be used. Only IP packets with the TCP SYN flag AND NOT the TCP ACK flag set will match this IP filter.

To match the response to initiation of a TCP connection, **-tcp syn -tcp ack** would be needed. Only IP packets with BOTH the TCP SYN and TCP ACK flags set would match this IP filter.

port# Ethernet interface number. Can be 0 or 1.

Examples:

```
eth ip filter flush input 0
```
This command deletes all IP filters of type Input on the Ethernet interface 0

```
eth ip filter append forward deny
```
This command will deny the forwarding of all IP traffic. This IP filter could become the "last" IP filter as a default action.

ETH IP FIREWALL

The router supports IP Internet Firewall Filtering to prevent unauthorized access to your system and network resources from the Internet. This filter discards packets received from the WAN that have a source IP address recognized as a local LAN address. This command requires a reboot.
This command sets Ethernet Firewall Filtering ON or OFF and allows you to list the active state.

Note: To perform Firewall Filtering, IP routing must be enabled.

eth ip firewall on|off|list

on	Sets firewall filtering on. IP routing must also be enabled for filtering to be performed.
off	Sets firewall filtering off.
list	Lists the current status of firewall filtering.

Example: `eth ip firewall list`

Response:
```
The Internet firewall filter is currently on.
0 offending packets were filtered out.
```

ETH IP OPTIONS

RIP is a protocol used for exchanging IP routing information among routers. The following RIP options allow you to set IP routing information protocol controls on the local Ethernet LAN.

Note: This command requires a reboot.

eth ip options <*option*> on|off [<*port#*>]

option	Must be one of the following:
rxrip	Receive and process IP RIP-1 compatible and RIP-2 broadcast packets from the Ethernet LAN. Also receive and process RIP-2 packets that are multicast as defined by the **eth ip ripmulticast** command. Set this option if the local router is to discover route information from the Ethernet LAN. This defaults to **ON**.
rxrip1	Receive and process RIP-1 packets only.
rxrip2	Receive and process RIP-2 packets only.
rxdef	Receive the default route address from the Ethernet LAN. This defaults to **ON**. This option is useful if you do not want to configure your router with a default route.
txrip	Transmit RIP-1 compatible broadcast packets and RIP-2 multicast packets over the Ethernet LAN. This defaults to **ON**.
txrip1	Transmit broadcast RIP-1 packets only.
txrip2	Transmit multicast RIP-2 packets only.

txdef/avdfr Advertise this router as the default router over the Ethernet LAN (provided it has a default route!). This default is set to ON. Set this to OFF if another router on the local LAN is the default router.

port# Port number of the Ethernet LAN. This number must be 0 or 1, or may be omitted.

Example: `eth ip options avdfr off`

ETH IP RIPMULTICAST

This commands lets you change the multicast address for RIP-1 compatible and RIP-2 packets. The default address is 224.0.0.9.

eth ip ripmulticast <*ipaddr*> [<*port#*>]

ipaddr IP address of the remote network or station, in the format of 4 decimals separated by periods.

port# Port number of the Ethernet LAN. This number must be 0 or 1, or may be omitted.

Example: `eth ip ripmulticast 128.1.210.64`

ETH IPX ADDR

Sets the IPX network number for the Ethernet LAN connection.

eth ipx addr <*ipxnet*> [*port#*]

ipxnet IPX network number represented by 8 hexadecimal characters.

port# Port number of the Ethernet LAN. This number must be 0 or 1, or may be omitted.

Example: `eth ipx 123`

ETH IPX DISABLE

Disables IPX routing across the Ethernet LAN. This acts as a master switch allowing you to disable IPX Routing for testing or control purposes.

Note: This command requires a reboot.

eth ipx disable [*port#*]

port# Port number of the Ethernet LAN. This number must be 0 or 1, or may be omitted.

Example: `eth ipx disable`

ETH IPX ENABLE

Enables IPX routing across the Ethernet LAN. This acts as a master switch allowing you to enable IPX routing.

Note: This command requires a reboot.

eth ipx enable [*port#*]

port# Port number of the Ethernet LAN. This number must be 0 or 1, or may be omitted.

Example: `eth ipx enable`

ETH IPX FRAME

Sets the frame encapsulation method. The default is 802.2.

eth ipx frame <*type*>

type **802.2** (DEC standard)
802.3 (Intel standard)
dix (Xerox/Ethernet II standard)

Example: `eth ipx frame 802.3`

ETH LIST

Lists the Ethernet LAN port number, status of bridging and routing, IP protocol controls, and IP address and subnet mask.

eth list

Example: `eth list`

Response:

```
ETHERNET INFORMATION FOR <ETHERNET/0>
  Hardware MAC address.................. 00:20:6F:02:98:04
  Bridging enabled...................... no
  IP Routing enabled.................... no
    Firewall filter enabled ............ yes
    Send IP RIP to the LAN.............. rip-1 compatible
      Advertise me as default router... yes
    Process IP RIP packets received.... rip-1 compatible
      Receive default route by RIP..... yes
  RIP Multicast address................. default
  IP address/subnet mask................ 192.168.254.254/255.255.255.0
  IP static default gateway............. none
  IPX Routing enabled................... no
    External network number............. 00000000
    Frame type.......................... 802.2
```

Remote Router Access Configuration (REMOTE)

The following commands allow you to add, delete, and modify remote routers to which the target router can connect. Remote router information that can be configured includes:

- PVC numbers
- Security authentication protocols and passwords
- WAN IP/ IPX addresses
- IP routes
- IPX routes and SAPS
- Remote bridging addresses and bridging control
- Host mapping
- Encryption (option)
- IP Filtering (option)
- L2TP tunneling (option)

REMOTE ?

Lists the supported keywords.

remote ?

Response:

```
Sub-commands for remote:
?                    help                 add
del                  list                 enable
disable              setAuthen            enaAuthen
disAuthen            setPasswd            setOurPasswd
delOurPasswd         setOurSysName        delOurSysName
listPhones           setPVC               setProtocol
addServer            delServer            setIPTranslate
setCompression       stats                statsclear
setSrcIpAddr         setRmtIpAddr         addIproute
delIproute           setIpOptions         listIproutes
setIpxaddr           addIpxroute          delIpxroute
listIpxroutes        addIpxsap            delIpxsap
listIpxsaps          listBridge           enaBridge
disBridge            setBrOptions         setEncryption
delEncryption        addHostMapping       delHostMapping
set l2tpclient       setLNS
```

REMOTE ADD

Adds a remote router entry into the remote router database.

remote add <*remoteName*>

remoteName Name of the remote router (character string). The name is case-sensitive.

Example: `remote add HQ`

REMOTE ADDHOSTMAPPING

This command is used to remap a range of local LAN IP addresses to a range of public IP addresses on a per-remote-router basis. These local addresses are mapped one-to-one to the public addresses.

Note: The range of public IP addresses is defined by <first public addr> only. The rest of the range is computed automatically (from <first public addr> to <first public addr> + number of addresses remapped - 1) inclusive.

remote addHostMapping <*first private addr*><*second private addr*><*first public addr*><*remoteName*>

first private addr First IP address in the range of local IP address to be remapped, in the format of 4 decimals separated by periods.

second private addr Last address in the range of local IP address to be remapped, in the format of 4 decimals separated by periods.

first public addr Defines the range of public IP addresses, in the format of 4 decimals separated by periods.

The rest of the range is computed automatically.

remoteName Name of the remote router (character string).

Example: `remote addHostMapping 192.168.207.40 192.168.207.49 10.0.20.11 HQ`

REMOTE ADDIPROUTE

Adds an IP address route for a network or station on the LAN network connected beyond the remote router. The target router's routing table must be seeded statically to access networks and stations beyond this remote router. After the connection is established, standard RIP update packets will dynamically add to the routing table. Setting this address is not required if a target router never connects to the remote router and the remote router supports RIP.

Note: A **reboot** must be performed on the target router for the addition of a static route to take effect.

remote addIpRoute <*ipnet*> <*ipnetmask*> <*hops*> <*ipgateway*> <*remoteName*>

ipnet IP address of the remote network or station, in the format of 4 decimals separated by periods.

ipnetmask IP network mask of the remote network or station, in the format of 4 decimals separated by periods.

hops Number between 1 and 15 that represents the perceived cost in reaching the remote network or station.

ipgateway Enter a gateway address only if you are configuring RFC 1483MER. The gateway address that you enter is the address of a router on the remote LAN. Check with your system administrator for details.

remoteName Name of the remote router (character string).

Examples:
```
remote addIpRoute 128.1.210.64 255.255.255.192 1 HQ
remote addIpRoute 128.1.210.032 255.255.255.224 1 HQ
remote addIpRoute 128.1.206.0 255.255.255.0 2 HQ
remote addIpRoute 128.1.210.072 255.255.255.255 1 HQ
remote addIpRoute 0.0.0.0 255.255.255.255 1 HQ
remote addIproute 0.0.0.0 255.255.255.255 1 187.12.10.1 HQ
```

The first two addresses in the list represent subnetworks, the third is a class B network, and the fourth is a host. The fifth address is the default route.

REMOTE ADDIPXROUTE

Adds an IPX route for a network or station on the LAN network connected beyond the remote router. The target router's routing information table must be seeded statically to access networks and stations beyond this remote router. After the connection is established, standard RIP update packets will dynamically add to the routing table. (Setting this address is not required if a target router never connects to the remote router and the remote router supports RIP.)

Note: A **reboot** must be performed on the target router for the addition of a static route to take effect.

remote addIpxRoute <*ipxNe#*> <*metric*> <*ticks*> <*remoteName*>

ipxNe# IPX network number represented by 8 hexadecimal characters.

metric Number of routers through which the packet must go to get to the network/station.

ticks Number in 1/8 seconds which is the estimated time delay in reaching the remote network or station.

remoteName Name of the remote router (character string).

Example:
```
remote addIpxRoute 456 1 4 HQ
```

REMOTE ADDIPXSAP

Adds an IPX SAP to the server information table for a service on the LAN network connected beyond the remote router. The target router's SAP table must be seeded statically to access services beyond this remote router. After the connection is established, standard SAP broadcast packets will dynamically add to the table.

Note: A **reboot** must be performed on the target router for the addition of a SAP to take effect.

remote addIpxSap <*servicename*> <*ipxNet* > <*ipxNode*> <*socket*> <*type*> <*hops*> <*remoteName*>

servicename Name of server

ipxNet IPX network number represented by 8 hexadecimal characters.

ipxNode IPX node address represented by 12 hexadecimal characters.

socket Socket address of the destination process within the destination node. The processes include services such as file and print servers.

type Number representing the type of server.

hops Number of routers through which the packet must go to get to the network/station.

remoteName Name of the remote router (character string).

Example: `remote addIpxSap Fileserver 010a020b 0108030a0b0c 451 HQ`

REMOTE ADDSERVER

This Network Address Translation (NAT) command is used to add a server's IP address (on the LAN) associated with this remote router for a particular protocol.

remote addServer <*ipaddr***>**|discard|me <*protocolid***>** |tcp|udp<*first port*> |ftp|telnet|smtp|snmp|http [<*last port*> [<*first private port*>]] <*remoteName*>

ipaddr IP address of the host selected as server in the format of 4 decimals separated by periods

discard Used to discard the incoming server request.

me Used to send the incoming server request to the local router, regardless of its IP address.

protocolid Protocol used by the selected server; can be **tcp** or **udp.**

first port First or only port as seen by the remote end**.** Port used by the selected server;
Can be as string such as ftp, telnet, smtp, snmp, or http, or a numeric value between 0 and 65,535. A numeric value of 0 will match any port

last port If specified, is used with <first port> to specify a range of ports as seen by the remote end for the server on the LAN.

first private port If specified, is a port remapping of the incoming request from the remote end.

remoteName Name of the remote router (character string).

Examples:
```
remote addServer 192.168.1.5 tcp smtp
remote addServer 192.168.1.10 tcp 9000 9000 telnet router2
```

REMOTE DEL

Deletes a remote router entry from the remote router database.

remote del <*remoteName*>

remoteName Name of the remote router (character string).

Example: `remote del HQ`

REMOTE DELENCRYPTION

Deletes encryption files associated with a remote router.

remote delencryption <*remoteName*>

remoteName Name of the remote router (character string).

Example: `remote delEncryption HQ`

REMOTE DELHOSTMAPPING

This command is used to undo an IP address/ host translation (remapping) range that was previously established with the command **remote addhostmapping** on a per-remote-router basis.

remote delHostMapping <*first private addr*> <*second private addr*> <*first public addr*> <*remoteName*>

first private addr First IP address in the range of IP address, in the format of 4 decimals separated by periods.

second private addr Last address in the range of IP address, in the format of 4 decimals separated by periods.

first public addr Defines the range of public IP addresses, in the format of 4 decimals separated by periods.

The rest of the range is computed automatically.

remoteName Name of the remote router (character string).

Example: `remote delHostMapping 192.168.207.40 192.168.207.49 10.0.20.11 HQ`

REMOTE DELIPROUTE

Deletes an IP address for a network or station on the LAN network connected beyond the remote router.
Note: A **reboot** must be performed on the target router for a deletion of a static route to take effect.

remote delIpRoute <*ipnet*> <*remoteName*>

ipnet IP address of the remote network or station, in the format of 4 decimals separated by periods.

remoteName Name of the remote router (character string).

Example: `remote delIpRoute 128.1.210.64 HQ`

REMOTE DELIPXROUTE

Deletes an IPX address for a network on the LAN network connected beyond the remote router.

Note: A **reboot** must be performed on the target router for a deletion of a static route to take effect.

remote delIpxroute <*ipxNet*> <*remoteName*>

ipxNet IPX network number represented by 8 hexadecimal characters.

remoteName Name of the remote router (character string).

Example: `remote delIpxRoute 010a020b HQ`

REMOTE DELIPXSAP

Deletes an IPX service on the LAN network connected beyond the remote router.

Note: A **reboot** must be performed on the target router for a deletion of a service to take effect.

remote delIpxsap <*servicename*> <*remoteName*>

servicename Name of server

remoteName Name of the remote router (character string).

Example: `remote delIpxSap Fileserver HQ`

REMOTE DELOURPASSWD

Removes the unique CHAP or PAP authentication password entries established by the command **remote setOurPasswd.**

remote delOurPasswd <*remoteName*>

remoteName Name of the remote router (character string.).

Example: `remote delOurPasswd HQ`

REMOTE DELOURSYSNAME

Removes the unique CHAP or PAP authentication system name entries established by the command **remote setOurSysName.**

remote delOurSysName <*remoteName*>

remoteName Name of the remote router (character string).

Example: `remote delOurSysName HQ`

REMOTE DELSERVER

This Network Address Translation (NAT) command is used to delete an entry created by the **remote addServer** command. Please refer to the section Server Configuration, page 86, for detailed information.

remote delServer <*ipaddr*>|discard|me <*protocolid*> |tcp|udp <*first port*> |ftp|telnet|smtp|snmp|http [<*last port*> [<*first private port*>]] <*remoteName*>

ipaddr IP address of the host selected as server in the format of 4 decimals separated by periods

discard Used to discard the incoming server request.

me Used to send the incoming server request to the local router, regardless of its IP address.

protocolid Protocol used by the selected server;
Can be **tcp** or **udp.**

first port First or only port as seen by the remote end**.** Port used by the selected server;
Can be as string such as ftp, telnet, smtp, snmp, or http, or a numeric value between 0 and 65,535. A numeric value of 0 will match any port

last port If specified, is used with <first port> to specify a range of ports as seen by the remote end for the server on the LAN.

first private port If specified, is a port remapping of the incoming request from the remote end.

remoteName Name of the remote router (character string).

Example: `remote delServer 192.168.1.5 tcp ftp router1`

REMOTE DISABLE

Disables communications with the remote router. This allows you to enter routers into the remote router database but sets them inactive.

Note: The routing information defined for <routerName> is still in effect when the entry is disabled until you save and reboot. However, no calls will be made to that remote router.

remote disable <*remoteName*>

remoteName Name of the remote router (character string).

Example: `remote disable HQ`

REMOTE DISAUTHEN

This command is intended for situations where third-party routers are not capable of being authenticated: the target router will not attempt to authenticate the remote router.

remote disAuthen <*remoteName*>

remoteName Name of the remote router (character string).

Example: `remote disAuthen HQ`

REMOTE DISBRIDGE

Disables bridging from the target router to the remote router.

Note: This command requires rebooting the target system for the change to take effect.

remote disBridge <*remoteName*>

remoteName Name of the remote router (character string).

Example: `remote disBridge HQ`

REMOTE ENAAUTHEN

With this command the target router will try to negotiate authentication as defined in the remote router's database.

remote enaAuthen <*remoteName*>

remoteName Name of the remote router (character string).

Example: `remote enaauthen HQ`

REMOTE ENABLE

Enables communications with the remote router. This command allows you to activate the entry in the remote router database when you are ready.

remote enable <*remoteName*>

remoteName Name of the remote router (character string).

Example: `remote enable HQ`

REMOTE ENABRIDGE

Enables bridging from the target router to the remote router. This command requires rebooting the target system for the change to take effect.

remote enaBridge <*remoteName*>

remoteName Name of the remote router (character string).

Example: `remote enaBridge HQ`

REMOTE IPFILTER

This command is used to define an IP filter on the remote/WAN interface of the connection to establish a Firewall. The filter is used to screen IP packets and operates at the interface level. Each interface is defined by 3 types of filters: Input, Forward, and Output filters. For more information on IP filters, please refer to the topic IP Filtering, page 98.

remote ipfilter *<command> <type> <action> <parameters> <remoteName>*		
command	append <type> <action> <parameters>	Append a filter to the end of this <type>
	insert <type> <action> <parameters>	Insert a filter at the front of this <type>
	delete <type> <action> <parameters>	Delete the first filter matching this filter
	flush <type>	Delete all filters of this <type> from this interface
	check <type> <parameters>	Check the action to take (Accept, Drop, Reject) based on the parameters
	list <type>	List all filters of a <type> on this interface
	watch on | off	Print out a message to the console if a packet to or from this remote is dropped or rejected
type	input, output, forward	
action	accept, drop, reject	
parameters	Each IP filter can have any combination of the following parameters used for matching against the IP packet. Below are the option/value pairs currently possible:	

-p <protocol>|TCP|UDP|ICMP
where <protocol> is an IP protocol number or the string "TCP", "UDP", "ICMP".
If <protocol> is 0 (or the -p option is not specified), this IP filter will match ANY protocol.

-sa <first source ip addr>[:<last source ip addr>]
where <first source ip addr> defines the first or only source IP address and <last source ip addr>, if present, defines the last source IP address in a range. If not specified, <first source ip addr> is assumed to be 0.0.0.0, <last source ip addr> is assumed to be 255.255.255.255.

-sm <source ip mask>
where <source ip mask>, when present, defines a mask to use when comparing the <first source ip addr>...<last source ip addr> with the source IP address in the IP packet. If not specified, the source IP mask is set to 255.255.255.255.

-sp <first source port>[:<last source port>]
where <first source port> defines the first or only source port and <last source port>, if present, defines the last source port in a range. If not specified, the <first source port> is assumed to be 0, the <last source port> is assumed to be 0xffff.

--da <first dest ip addr>[:<last dest ip addr>]
where <first dest ip addr> defines the first or only destination IP address and <last dest ip addr>,

if present, defines the last destination IP address in a range. If not specified, <first dest ip addr> is assumed to be 0.0.0.0, <last dest ip addr> is assumed to be 255.255.255.255.

-dm <dest ip mask>
where <dest ip mask>, when present, defines a mask to use when comparing the <first dest ip addr>...<last dest ip addr> with the destination IP address in the IP packet. If not specified, the destination IP mask is set to 255.255.255.255.

-dp <first dest port>[:<last dest port>]
where <first dest port> defines the first or only destination port and <last dest port>, if present, defines the last destination port in a range. If not specified, the <first dest port> is assumed to be 0, the <last dest port> is assumed to be 0xffff.

-b
This option indicates that this filter should be tested twice; a first time with the source filter information matched against the source information in the IP packet and the destination filter information matched against the destination information in the IP packet; and a second time with the source filter information matched against the destination information in the IP packet and the destination filter information matched against the source information in the IP packet.

-c <count of times rule used>
indicates how many IP packets have matched this filter since the router was rebooted.

-tcp syn|ack|noflag
where **syn** is the TCP SYN flag, **ack** is the TCP ACK flag, and **noflag** means there is a TCP packet AND neither the SYN flag or the ACK flag are set. This option is ignored if the IP packet is not a TCP packet. If not specified, the TCP SYN and TCP ACK flags are not checked when matching the IP packet with this filter.

Note:MORE than one **-tcp** option in an IP filter may be specified. For example, to match this IP filter against the initiation of a TCP connection, **-tcp syn** would be used. Only IP packets with the TCP SYN flag AND NOT the TCP ACK flag set will match this IP filter.

To match the response to initiation of a TCP connection, **-tcp syn -tcp ack** would be needed. Only IP packets with BOTH the TCP SYN and TCP ACK flags set would match this IP filter.

remoteName Name of the remote router (character string)

Examples:

```
remote ipfilter flush forward internet
```
This command deletes all IP filters of type Forward on the remote interface internet.

```
remote ipfilter append forward drop -da 192.168.0.0 -dm 255.255.0.0 internet
```
This command will deny any IP traffic whose destination address is 192.168.0.0 masked with 255.255.0.0 (i.e., matches IP addresses 192.168.0.0 through 192.168.255.255) to the remote internet.

```
remote ipfilter append forward drop -da 192.168.0.0:192.168.255.255 internet
```
This command has the SAME effect as the previous filter.

```
remote ipfilter list forward internet
```
This command will list all IP filters defined of type Forward on the remote internet.

REMOTE LIST

Lists the remote router entry in the remote router database or all the entries in the database. The result is a complete display of the current configuration settings for the remote router(s), except for the authentication password/secret.

remote list [<*remoteName*>]

remoteName Name of the remote router (character string)

Example: `remote list HQ`

Response:

```
INFORMATION FOR <HQ>
  Status.............................. enabled
  Our Password used when dialing out... no
  Protocol in use..................... RFC1483 (SNAP) - Frame Relay IP
  Connection Identifier (VPI*VCI)..... 0*38
  IP address translation.............. off
  Compression Negotiation............. off
  Source IP address/subnet mask....... 0.0.0.0/0.0.0.0
  Remote IP address/subnet mask....... 0.0.0.0/0.0.0.0
  Send IP RIP to this dest............ no
    Send IP default route if known..... no
  Receive IP RIP from this dest....... no
    Receive IP default route by RIP.... no
  Keep this IP destination private..... yes
  Total IP remote routes.............. 1
           128.1.0.0/255.255.0.0/1
  IPX network number.................. 00000789
  Total IPX remote routes............. 1 00001001/1/4
  Total IPX SAPs...................... 1
    SERV312_FP 00001001 00:00:00:00:00:01 0451 0004 1
  Bridging enabled.................... yes
    Exchange spanning tree with dest... no
```

REMOTE LISTBRIDGE

Lists the bridging capability from the target router to the remote router.

remote listBridge <*remoteName*>

remoteName Name of the remote router (character string).

Example: `remote listBridge HQ`

Response:

```
BRIDGING INFORMATION FOR <HQ>
 Bridging enabled.................. yes
 Exchange spanning tree with dest.... yes
```

REMOTE LISTIPROUTE

Lists all network or station IP addresses defined for the LAN network connected beyond the remote router. If the remote name is not specified, a list of IP Routes is displayed for each remote router in the database.

remote listIproutes [*remoteName*]

remoteName Name of the remote router (character string).

Example: `remote listIpRoute HQ`

Response:

```
IP INFORMATION FOR <HQ>
Send IP RIP to this dest.............   rip-1 compatible
    Send IP default route if known..... no
  Receive IP RIP from this dest........ rip-2
    Receive IP default route by RIP.... yes
  Keep this IP destination private..... no
  Total IP remote routes................ 0
```

REMOTE LISTIPXROUTE

Lists all network IPX route addresses defined for the LAN network connected beyond the remote router. The network number, hop count, and ticks are displayed. If the remote name is not specified, a list of IPX routes is displayed for each remote router in the database.

remote listIpxroutes [*remoteName*]

remoteName Name of the remote router (character string).

Example: `remote listIpxRoute HQ`

Response:

```
IPX ROUTE INFORMATION FOR <HQ>
Total IPX remote routes.............. 1    00001001/1/4
```

REMOTE LISTIPXSAPS

Lists all services defined for the LAN network connected beyond the remote router. Each service includes the server name, network number, node number, socket number, server type, and hop count. If the remote name is not specified, a list of IPX SAPs is displayed for each remote router in the database.

remote listipxsaps [*remoteName*]

remoteName Name of the remote router (character string.)

Example: `remote listIpxSap HQ`

Response:

```
  IPX SAP INFORMATION FOR <HQ>
Total IPX SAPs..................... 1
SERV312_FP 00001001 00:00:00:00:00:01 0451 0004 1
```

```
IPX SAP INFORMATION FOR <ISP>
Total IPX SAPs..................... 0
   SERV312_FP 00001001 00:00:00:00:00:01 0451 0004 1
```

REMOTE LISTPHONES

Lists the PVC numbers available for connecting to the remote router.

remote listPhones <*remoteName*>

remoteName Name of the remote router (character string).

Example: `remote listPhone HQ`

Response:

```
PHONE NUMBER(s) FOR <HQ>
Connection Identifier (VPI*VCI)...... 0*38
```

Note: If the remote name is not specified, a list of phone numbers is displayed for each remote router in the database.

REMOTE SETATMTRAFFIC SCR MBS

This command applies only to ATM routers. Please refer to page 166 for more syntax information.

remote setATMTraffic scr mbs <*remoteName*>

REMOTE SETAUTHEN

Sets the authentication protocol used when communicating with the remote router. The authentication protocol is the minimum security level that the target router must use with the remote router; this level is verified during security negotiation. The router will *always* attempt to negotiate the highest level of security possible (CHAP). The router will not accept a negotiated security level less than this minimum authentication method.

The parameter in the remote router database is used for the local side of the authentication process; the minimum security level used by the target router when challenging or authenticating the remote router.

remote setAuthen <*protocol*> <*remoteName*>

protocol **chap**, **pap,** or **none**. The default is **pap**.

remoteName Name of the remote router (character string).

Example: `remote setAuthen pap HQ`

REMOTE SETBROPTIONS

Sets controls on the bridging process.

Warning: Do not change this setting without approval of your system administrator.

remote setBrOptions <*option*> on|off <*remoteName*>

option stp

Use the spanning tree protocol for bridging. Set this option on only if the bridging peers support the spanning tree protocol and you wish to detect bridging loops. The default is **on**.

Note: This adds a 40-second delay each time the ADSL or ATM link comes up; use only if necessary.

remoteName Name of the remote router (character string).

Example: `remote setbroptions stp on HQ`

REMOTE SETCOMPRESSION

Used to enable or disable compression between the local router and the remote router.

remote setCompression on|off <*remoteName*>

on Compression will be negotiated between the local and the remote router if both routers are set to perform compression and if they both share a common compression protocol.

off Disables compression. The default is OFF.

remoteName Name of the remote router (character string).

Example: `remote setCompression on HQ`

REMOTE SETDLCI

Please refer to page 176 for further information regarding this command.

REMOTE SETENCRYPTION (RFC 1969 Encryption)

This command is used to specify a PPP DES (Data Encryption Standard) 56-bit key with fixed transmit and receive keys.

remote setEncryption dese rx|tx <*key*> <*remoteName*>

rx Receive key

tx Transmit key

key Key in the format of an eight-hexadecimal number

remoteName Name of the remote router (character string).

Example:
```
remote setEncryption dese tx 1111111111111111 HQ
remote setEncryption dese rx 2222222222222222 HQ
```

REMOTE SETENCRYPTION (Diffie-Hellman Encryption)

This command is used to specify encryption based on the Diffie-Hellman key exchange protocol. Each router possesses an internal encryption file that is associated with a public key providing 768-bit security. The predefined keys can be replaced by the user. The configuration file on the router must have a "num" suffix (e.g. dh96.num).

remote setEncryption *DESE_1_KEY*|*DESE_2_KEY* [<*filename*>] <*remoteName*>

DESE_1_KEY Specifies that the same key is used in both directions

DESE_2_KEY Specifies that the keys are different

filename Name of the file containing the Diffie-Hellman values. If not specified, default values built into the router's kernel are automatically selected.

remoteName Name of the remote router (character string).

Example:
```
remote setEncryption DESE_1_KEY dh96.num HQ
```

REMOTE SETIPOPTIONS

RIP is a protocol used for exchanging IP routing information among routers. The following RIP options allow you to set IP routing information protocol controls over a point-to-point WAN.

remote setIpOptions <*option*> on|off <*remoteName*>

option Includes the following choices:

rxrip Receive and process IP RIP-1 compatible packets and RIP-2 broadcast packets from the remote site. Also receive and process RIP-2 multicast packets.

Set this option if the local router is to discover route information from other sites connected to the remote router. This is useful for hierarchical organizations. If connecting to another company or an Internet Service Provider, you may wish to set this option off. The default is **off.**

rxrip1 Receive and process RIP-1 packets only.

rxrip2 Receive and process RIP-2 packets only.

rxdef Receive default IP route address. Set on, the local router will receive the remote site's default IP route. The default is **off.**

txrip Transmit IP RIP-1 compatible broadcast packets and RIP-2 multicast packets to the remote site. Set on, the local router will send routing information packets to the remote site. The default is **off.**

txrip1 Transmit broadcast RIP-1 packets only.

txrip2 Transmit multicast RIP-2 packets only.

txdef Transmit the local router's default IP route. Set on, the local router will send the default route to the remote site. The default is **off.**

private Keep IP routes private. Used to prevent advertisement of this route to other sites by the remote router. Used as a security mechanism when the remote site is outside your company (an Internet Service Provider, for example), or whenever you would prefer to keep the identify of the site private. The default is **yes.**

multicast Allows the remote router to send and receive IP multicast traffic.

remoteName Name of the remote router (character string).

Example: `remote setipoptions private on HQ`

REMOTE SETIPTRANSLATE

This command is used to control Network Address Translation on a per remote router basis. It allows several PCs to share a single IP address to the Internet. The remote router must assign the source WAN IP address to the routers' local WAN port. This command requires that you define a Source WAN IP Address with the command: **remote setSrcIpAddr**

remote setIPTranslate on|off <*remoteName*>

remoteName Name of the remote router (character string).

Example: `remote setIPTranslate on HQ`

REMOTE SETIPXADDR

Sets the IPX network number for the remote WAN connection.

remote setIpxaddr <*ipxNet*> [*port#*]

ixpNet IPX network number represented by 8 hexadecimal characters.

port# Port number of the Ethernet LAN. This number must be 0 or may be omitted.

Example: `remote setipxaddr 789 HQ`

REMOTE SETL2TPCLIENT

This command is specific to L2TP tunnel configuration. Please, refer to the L2TP commands section, page 191, for more usage information.

remote setl2tpclient <*TunnelName*><*remoteName*>

REMOTE SETLNS

This command is specific to L2TP tunnel configuration. Please, refer to the L2TP commands section, page 197, for more usage information.

remote setLNS <*TunnelName*><*remoteName*>

REMOTE SETOURPASSWD

Sets a unique CHAP or PAP authentication password for the local router used for authentication when the local router connects to the specified remote router. This password overrides the password set in the **system passwd** command. A common use would be to set a password assigned to you by Internet Service Providers.

remote setOurPasswd <*password*> <*remoteName*>

password Authentication password of the local router for use in connecting to the remote router.

Note: the password is case-sensitive.

remoteName Name of the remote router (character string).

Example: `remote setOurPasswd s1dpxl7 HQ`

REMOTE SETOURSYSNAME

Sets a unique CHAP or PAP authentication system name for the local router used for authentication when the local router connects to the specified remote router. This system name overrides the system name set in the **system name** command. A common use would be to set a password assigned to you by Internet Service Providers.

remote setOurSysName <*name*> <*remoteName*>

name System name of the target router.

Note: The system name is case-sensitive and must be no more than 50 characters.

remoteName Name of the remote router (character string).

Example: `remote setOurSysName s1dpxl7 HQ`

REMOTE SETPASSWD

Sets the CHAP or PAP authentication password used when the remote router establishes a connection or is challenged by the target router.

remote setPasswd <*password*> <*remoteName*>

password Authentication password of the remote router. Not that the password is case-sensitive.

remoteName Name of the remote router (character string).

Example: `remote setPasswd s2dpxl7 HQ`

REMOTE SETPROTOCOL

Sets the link protocol for the remote router.

remote setProtocol [PPP | PPPLLC | RFC1483 | RFC1483MER | FRF8 | RAWIP] <*remoteName*>

PPP	PPP protocol with no encapsulation
PPLLC	PPP protocol with LLC SNAP encapsulation (used with frame relay internetworking units)
RFC1483	RFC 1483 protocol
RFC1483MER	RFC 1483MER (MAC Encapsulated Routing) protocol
FRF8	This protocol implements ATM to frame relay as defined in the Frame Relay Forum FRF.8 Interworking Agreement.
RAWIP	RawIP protocol
remoteName	Name of the remote router (character string).

Example: `remote setprotocol ppp fp1`

REMOTE SETPVC

Specifies the PVC number to be used when connecting to the remote router.

remote setPVC <*vpi number*>*<*vci number*> <*remoteName*>

vpi number	Virtual Path ID — Number that identifies the link formed by the virtual path.
vci number	Virtual Circuit ID — Number that identifies a channel within a virtual path in a DSL/ATM environment.
remoteName	Name of the remote router (character string).

Example: `remote setPVC 0*38 HQ`

REMOTE SETRMTIPADDR

Sets the WAN IP address for the remote router. This address is required only if the remote router does not support IP address negotiation under PPP (i.e., numbered mode is required and the remote router cannot specify a WAN IP address for use during the negotiation process).

remote setRmtIpAddr <*ipaddr*> <*mask*> <*remoteName*>

ipaddr	IP address of the remote router, in the format of 4 decimals separated by periods.

mask IP network mask of the remote router, in the format of 4 decimals separated by periods.

remoteName Name of the remote router (character string).

Example: `remote setRmtIpAddr 128.1.210.65 255.255.255.192 HQ`

REMOTE SETSRCIPADDR

Sets the IP address for the target WAN connection to the remote router. You may set this address when the remote router requires the target and remote WAN IP addresses to be on the same subnetwork. Another instance is to force numbered mode and to prevent the remote router from changing the target WAN IP Address through IPCP address negotiation. The target WAN IP Address defaults to the Ethernet LAN IP address.

remote setSrcIpAddr <*ipaddr*> <*mask*> <*remoteName*>

ipaddr Target IP addr of the WAN connection to the remote router, in the format of 4 decimals separated by periods.

mask IP network mask, in the format of 4 decimals separated by periods.

remoteName Name of the remote router (character string).

Example: `remote setSrcIpAddr 128.1.210.151 255.255.255.192 HQ`

REMOTE STATS

Shows the current status of the connection to the remote router including the bandwidth and data transfer rate.

remote stats [<*remoteName*>]

remoteName Name of the remote router (character string).

Example: `remote stats HQ`

Response:

```
STATISTICS FOR <HQ>:
   Current state..................... currently connected
  Current output bandwidth.......... 0 bps
  Current input bandwidth........... 0 bps
  Current bandwidth allocated....... 25600000 bps
  On port ATM_VC/1.................. 0+01:02:36 (0%/0% of 25600000 bps)
  Total connect time................ 0+01:11:48
  Total bytes out................... 15896
  Total bytes in.................... 0

STATISTICS FOR <internet>:
  Current state..................... not connected
  Current output bandwidth.......... 0 bps
  Current input bandwidth........... 0 bps
  Current bandwidth allocated....... 0 bps
  Total connect time................ 0+00:00:00
  Total bytes out................... 0
  Total bytes in..................... 0
```

where:

Current state: connected, not connected, currently connecting, currently attempting to connect, currently closing, out of service, or not known

Bandwidth state: idle, increasing, decreasing, decreasing hold, unknown, and idle

REMOTE STATSCLEAR

Allows to reset the statistics counter for a given remote router.

remote statsclear <*remoteName*>

remoteName Name of the remote router (character string).

Example: `remote statsclear HQ`

Asymmetric Digital Subscriber Line Commands (ADSL)

The following ADSL commands are used to manage the ADSL link for an ADSL router.

ADSL ?

Lists the supported keywords.

adsl ?

Response:

```
ADSL commands:
?          restart        stats          speed
```

ADSL RESTART

This command is used to resynchronize the modem with the Central Office equipment.

adsl restart

Response:

```
# 12/02/1997-12:47:46:ADSL: Idle
12/02/1997-12:47:46:ADSL: Startup initiated
12/02/1997-12:47:48:ADSL: Startup training in progress
12/02/1997-12:47:54:ADSL: Modem started successfully
12/02/1997-12:47:54:ADSL: Near Avg SQ #: 44 dB [   3]
12/02/1997-12:47:54:ADSL: Far  Avg SQ #: 44 dB [   3]
12/02/1997-12:47:54:ADSL: Downstream rate: 6272 Kb/s, Upstream rate: 1088 Kb/s
12/02/1997-12:47:54:DOD: connecting to internet @ 0*38 over ATM_VC/1
12/02/1997-12:47:56:ADSL: Data Mode
DUM: BR CHG ATM_VC/1 - to internet now forwarding
```

ADSL SPEED

This command is used to display the current downstream and upstream rates.

adsl speed

Example: `adsl speed`

Response: `downstream rate: 6272 Kb/s, upstream rate: 1088 Kb/s`

ADSL STATS

Shows the current error status for the ADSL connection.

adsl stats [clear]

clear Option used to reset the counters

Example: `adsl stats`

Response:

```
ASDL Statistics:
  Out of frame errors....              0
  HEC errors received...               0
  CRC errors received....              0
  FEBE errors received...              0
Remote Out-of-frame....                0
Remote HEC errors......                0
```

Asynchronous Transfer Mode Commands (ATM)

The following ATM commands are used to manage the ATM link for an ATM router.

ATM ?

Lists the supported keywords.

atm ?

Example: `atm?`

Response:

```
ATM commands:
?            save         speed
reset        pcr
```

Note: Other ATM-specific commands are also included in this section:

atom dumpUnknownCells

atom findPVC

remote setatmtraffic

ATM PCR

This command sets the speed of the ATM link in cells per second. This command is similar to atm speed (speed in kilobytes). Please refer to the command **atm speed**.

Note: This command requires privileged access (login password).

atm pcr <*cells/seconds*>

cells/second number of cells per second

Example: `atm pcr 471`

ATM RESET

This command is used to perform traffic shaping. It causes the ATM link to re-initialize.

atm reset

Example: `atm reset`

ATM SAVE

This command is used to save the ATM configuration settings.

atm save

Example: `atm save`

ATM SPEED

This command sets the speed of the ATM link in kilobits per second. This command is similar to atm pcr (speed in cells per second). Please refer to the command **atm pcr**.
The upstream speed default is 326 Kb/s. Use this command if the upstream speed exceeds 326 Kb/s. The speed value is generally obtained from your Network Service Provider.

Note: This command requires privileged access (login password).

atm speed [*upstream speed in Kb/S*]

upstream speed in Kb/S Number provided by the Network Service Provider. 326 Kb/s is the default value for the upstream speed.

Example: `atm speed 326`

Response:
```
ATM Upstream Rate:   326 Kb/S
```

ATOM DUMPUNKNOWNCELLS

This command is used to look at the content of an ATM cell. It will not affect normal operation performance.

atom dumpunknowncells [on|off]

Example: `atom findPVC on`

ATOM FINDPVC

This command is normally used to try to find the ATM VPI*VCI number to be used to configure a remote when the Service Provider either has supplied the wrong value or simply is not able to supply one. Additionally, this command should only be used when there are NO remotes defined or when the remote entries are disabled.
The command output is directed to the console. If Telnet is used to log into the router, then issue the **system log start** command to direct the console output to the Telnet session.

atom findPVC [on|off]

Example: `atom findPVC on`

Response:
```
No remote entry found with PVC (VPI*VCI) 1*2
```

In this case, an ATM VPI*VCI is detected for which there is no remote defined.

1 is the number of the VPI as found in the ATM stream.

2 is the number of the VCI as found in the ATM stream.

The discovered number may be used as the VPI*VCI value in the remote, to determine if communications are possible.

REMOTE SETATMTRAFFIC

This command sets ATM traffic shaping on a remote router. ATM traffic shaping allows the user to set the average rate at which cells are sent (SCR, Sustained Cell Rate) to a value lower than the ATM link speed (PCR, Peak Cell Rate).

Note 1: This command can only apply to one remote router. ATM traffic shaping must be used if more than one remote router is defined.

Note 2: ATM traffic shaping can be disabled with **remote setATMTraffic 0 0** <*remoteName*>

remote setATMTraffic SCR MBS <*remoteName*>

SCR — Sustained Cell Rate (cells per second).

MBS — Maximum Burst Size (cells).

remoteName — Name of the remote router (character string).

Examples: Assume that the ATM link speed (upstream) is 200 Kb/second or 471 cells/second and an average upstream data rate of 20 Kb/second (47 cells/second) is desired, then the following command would be used:

```
remote setatmtraffic 47 31 HQ
```

To disable ATM traffic shaping on HQ, use:

```
remote setatmtraffic 0 0 HQ
```

Dual Ethernet Router Commands (ETH)

The following Ethernet commands are used to manage the Ethernet interfaces for the Dual Ethernet (Ethernet-to-Ethernet) router and thus are specific to this type of router only.

Note: For non-specific Ethernet commands, refer to the Ethernet Commands section of this chapter, page 136.

General information

This device may be configured via the Web Browser GUI or from the Command Line Interface (CLI). You will need to use the CLI to set up any DHCP options as well as configuring optional features like IP filtering.

The Dual Ethernet router has two interfaces:

ETH/0 = Hub with four 10Base-T connectors

ETH/1 = Single 10Base-T connector

Note 1: For configuration information, please refer to the Customer Release Notes provided with the Dual Ethernet router.

Note 2: When using the **Boot from Network** option from the boot menu to perform a boot code update or install a software key option, the boot request is sent out of the ETH/0 interface only.

ETH BR ENABLE

Enables bridging in a Dual Ethernet environment. This command requires rebooting the router for the change to take effect.

eth br enable

Example: `eth br enable`

ETH BR DISABLE

Disables bridging in a Dual Ethernet environment.

Note: This command requires rebooting the router for the change to take effect.

eth br disable

Example: `eth br disable`

ETH IP ADDHOSTMAPPING

This command is used to remap a range of local LAN IP addresses to a range of public IP addresses on a per-interface basis. These local addresses are mapped one-to-one to the public addresses.

Note: The range of public IP addresses is defined by <first public addr> only. The rest of the range is computed automatically (from <first public addr> to <first public addr> + number of addresses remapped - 1) inclusive.

eth ip addHostMapping <*first private addr*><*second private addr*><*first public addr*><*port#*>>

first private addr	First IP address in the range of local IP address to be remapped, in the format of 4 decimals separated by periods.
second private addr	Last address in the range of local IP address to be remapped, in the format of 4 decimals separated by periods.
first public addr	Defines the range of public IP addresses, in the format of 4 decimals separated by periods. The rest of the range is computed automatically.
port#	Ethernet interface number. Can be 0 or 1.

Example: `eth ip addHostMapping 192.168.207.40 192.168.207.49 10.0.20.11 1`

ETH IP ADDSERVER

This Network Address Translation (NAT) command is used to add a server's IP address (on the LAN) associated with this interface for a particular protocol.

eth ip addServer <*ipaddr*>|discard|me <*protocolid*> |tcp|udp<*first port*> |ftp|telnet|smtp|snmp|http [<*last port*> [<*first private port*>]] <*port#*>

ipaddr	IP address of the host selected as server in the format of 4 decimals separated by periods
discard	Used to discard the incoming server request.
me	Used to send the incoming server request to the local router, regardless of its IP address.
protocolid	Protocol used by the selected server; can be **tcp** or **udp.**
first port	First or only port as seen by the Ethernet interface**.** Port used by the selected server; Can be as string such as ftp, telnet, smtp, snmp, or http, or a numeric value between 0 and 65,535. A numeric value of 0 will match any port
last port	If specified, is used with <first port> to specify a range of ports as seen by Ethernet interface end for the server on the LAN.
first private port	If specified, is a port remapping of the incoming request from the Ethernet interface.
port#	Ethernet interface number. Can be 0 or 1.

Examples:
```
eth ip addServer 192.168.1.5 tcp smtp 1
eth ip addServer 192.168.1.10 tcp 9000 9000 telnet 0
```

ETH IP DELHOSTMAPPING

Note: This command is used to undo an IP address/ host translation (remapping) range that was previously established with the command **eth ip addhostmapping** on a per-interface basis.

eth ip delHostMapping <*first private addr*> <*second private addr*> <*first public addr*> <*port#*>

first private addr	First IP address in the range of IP address, in the format of 4 decimals separated by periods.
second private addr	Last address in the range of IP address, in the format of 4 decimals separated by periods.
first public addr	Defines the range of public IP addresses, in the format of 4 decimals separated by periods.The rest of the range is computed automatically.
port#	Ethernet interface number. Can be 0 or 1.

Example: `eth ip delHostMapping 192.168.207.40 192.168.207.49 10.0.20.11 1`

ETH IP DELSERVER

This Network Address Translation (NAT) command is used to delete an entry created by the **eth ip addServer** command.

eth ip delServer <*ipaddr*>|discard|me <*protocolid*> |tcp|udp <*first port*> |ftp|telnet|smtp|snmp|http [<*last port*> [<*first private port*>]] <*port#*>

ipaddr	IP address of the host selected as server in the format of 4 decimals separated by periods
discard	Used to discard the incoming server request.
me	Used to send the incoming server request to the local router, regardless of its IP address.
protocolid	Protocol used by the selected server; Can be **tcp** or **udp.**
first port	First or only port as seen by the Ethernet interface**.** Port used by the selected server; Can be as string such as ftp, telnet, smtp, snmp, or http, or a numeric value between 0 and 65,535. A numeric value of 0 will match any port
last port	If specified, is used with <first port> to specify a range of ports as seen by the Ethernet interface for the server on the LAN.
first private port	If specified, is a port remapping of the incoming request from the Ethernet interface.
port#	Ethernet interface number. Can be 0 or 1.

Example: `eth ip delServer 192.168.1.5 tcp ftp 0`

ETH IP TRANSLATE

Note: This command is used to control Network Address Translation on a per-interface basis. It allows several PCs to share a single IP address to the Internet

eth ip translate on|off <*port#*>

port# Ethernet interface number. Can be 0 or 1.

Example: `eth ip translate on 0`

High-Speed Digital Subscriber Line Commands (HDSL)

The following HDSL commands are used to manage the HDSL link for an HDSL router.

General information about HDSL

♦ Line activation

Line activation is independent of network settings. During activation, the Link light (on the front panel of the router) is yellow and then turns green when the link becomes active.

The router at the CPE end will try auto-speed detection starting at 384 and try to the next higher speed (for about 30 seconds per speed). The WAN light should turn yellow, then green when the link has activated.

Auto-speed detection can be turned off with the command **hdsl speed noauto**.

If the line was previously set to “no auto-speed” (noauto), the Link light will be amber instead, when the line tries to activate.

The **ifs** command displays the Link as either Off or Opened when successfully activated. Following is a sample output.

Sample:

```
ifs
Interface   Speed        In %     Out % Protocol     State
Connection
ETHERNET/0 10.0mb       0%/0%     0%/0% (Ethernet)   OPENED
HDSL/0      384kb       0%/0%     0%/0% ((HDSL)      OPENED
CONSOLE/0  9600 b       0%/0%     0%/0% (TTY)        OPENED
```

♦ Auto-speed sequence

Auto-speed starts with the lower speed (384) and tries to activate for 30 seconds. If no activation takes place, the next higher speed is attempted. The time intervals between activation may change if the modems don’t activate as expected. Following is a correct activation output.

```
03/09/1998-17:11:59:HDSL: Deactivated
03/09/1998-17:12:22:HDSL: CPE is Activating at 384 Kb/s
03/09/1998-17:13:00:HDSL: Deactivated
03/09/1998-17:13:01:HDSL: CPE is Activating at 1168 Kb/s
03/09/1998-17:13:32:HDSL: Deactivated
03/09/1998-17:13:32:HDSL: CPE is Activating at 1168 Kb/s
03/09/1998-17:14:11:HDSL: Deactivated
03/09/1998-17:14:12:HDSL: CPE is Activating at 384 Kb/s
03/09/1998-17:14:51:HDSL: Activated
03/09/1998-17:14:53:FRAMER: The framer is synchronized
```

HDSL ?

Lists the supported keywords.

hdsl ?

Example: `hdsl?`

Response:

```
HDSL commands:
?            help         terminal
save         speed
```

HDSL SPEED

CO end: This command is used to set the speed manually on the CO end only.

CPE end: The router on the CPE end is always in auto-speed mode: it uses an auto-speed algorithm to attempt to match the CO speed. The command **hdsl speed noauto** is used to override auto-speed.

Note 1: The command **hdsl speed** (with no option) displays the current speed if the modem has activated successfully.

Note 2: **hdsl speed noauto** should be followed by the command **hdsl save** to become persistent across reboots.

Note 3: During auto-speed search, use the command **hdsl speed** <*speed*> to stop the search and restart it at the speed you just entered.

hdsl speed [384 | 1168 | noauto]	
384	Default speed for the CO
1168	Authorized non-default speeds for the CO in Mbps
noauto	Used to override auto-speed on the CPE end

Example:

```
hdsl speed 1168
hdsl speed noauto
hdsl speed
```

HDSL SAVE

Used to save the HDSL-related changes across reboots.

hdsl save

Example: `hdsl save`

HDSL TERMINAL

The router is by default configured as the CPE. Use this command if you intend to configure the router as a Central Office equipment (CO).

hdsl terminal cpe is used to define the CPE (customer premise) end (default configuration)

hdsl terminal co is used to define the CO (central office) end.

hdsl terminal displays the current settings.

hdsl terminal [cpe|co]

co This option lets you define the router as the central office (CO).

Example: `hdsl terminal`

Response: `Customer Premises`

Example: `hdsl terminal co`

ISDN Digital Subscriber Line (IDSL)

General information about IDSL

- ### DLCI (Data Link Connection Identifier)

  The IDSL router can support several DLCI virtual circuits over a Frame Relay IDSL link. However, a typical connection to the Internet will require only one DLCI.

  The DLCI number must match the DLCI of the remote end.

  An activated router should have the LINE, CH1, CH2, and NT1 LEDs all lit green.

  The following IDSL commands are used to manage the IDSL link for an IDSL router.

ISDN ?

Lists the supported keywords.

isdn ?

Example: `isdn?`

Response:

```
ISDN commands:
?           help
save        set          list
```

ISDN LIST

Lists the current switch type information.

isdn list

Example: `isdn list`

Response: `Switch type is Frame Relay IDSL 144k`

ISDN SAVE

Used to save the IDSL-related changes across reboots.

isdn save all | dod | sys | eth | filter | isdn | dhcp

Example: `isdn save`

ISDN SET SWITCH

This command is used to specify link speeds of 64, 128, or 144 Kbps for the IDSL connection.

isdn set switch [FR64 | FR128 | FR144]

FR64	Link speed of 64 Kbps
FR128	Link speed of 128 Kbps
FR144	Link speed of 144 Kbps

Example: `isdn set switch fr144`

REMOTE SETDLCI

This command allows the user to set the Data Link Connection Identifier – an address identifying a logical connection – in a Frame Relay environment. This number is generally provided by the Network Service Provider.

remote setDLCI <*dlcinumber*> <*remoteName*>

dlcinumber	Frame Relay number identifying the data link connection
remoteName	Name of the remote router (character string)

Example: `remote setDLCI 16 HQ`

REMOTE SETPROTOCOL

This IDSL-specific command is used to select the appropriate link protocol for your IDSL connection. Your Network Service Provider will tell you which link protocol to use.

remote setprotocol [PPP | FR | MER] <*remoteName*>

PPP	PPP protocol with no encapsulation.
FR	RFC 1490 protocol (Multiprotocol encapsulation over Frame Relay)
MER	RFC 1490 protocol with MAC Encapsulated Routing
remoteName	Name of the remote router (character string)

Example: `remote setProtocol FR HQ`

Symmetric Digital Subscriber Line Commands (SDSL)

The following SDSL commands are used to manage the SDSL link for an SDSL router.

General information about SDSL

♦ Line activation

Line activation is independent of network settings. During activation, the Link light (on the front panel of the router) is yellow and then turns green when the link becomes active.

The router at the CPE end will try auto-speed detection starting at 384 and try to the next higher speed (for about 30 seconds per speed). The WAN light should turn yellow, then green when the link has activated.

Auto-speed detection can be turned off with the **command sdsl speed noauto**. If the line was previously set to “no auto-speed” (noauto), the Link light will be amber instead, when the line tries to activate.

The IFS command displays the Link as either Off or Opened when successfully activated. Following is a sample output.

Sample:

```
ifs
Interface   Speed        In %     Out % Protocol     State
Connection
ETHERNET/0 10.0mb       0%/0%     0%/0% (Ethernet)   OPENED
SDSL/0      384kb       0%/0%     0%/0% (ATM)        OPENED
CONSOLE/0  9600 b       0%/0%     0%/0% (TTY)        OPENED
```

♦ Auto-speed sequence

Auto-speed start with the lower speed (384) and tries to activate for 30 seconds. If no activation takes place, the next higher speed is attempted. The time intervals between activation may change if the modems don’t activate as expected. Following is a correct activation output.

```
03/09/1998-17:11:59:SDSL: Deactivated
03/09/1998-17:12:22:SDSL: CPE is Activating at 768 Kb/s
03/09/1998-17:13:00:SDSL: Deactivated
03/09/1998-17:13:01:SDSL: CPE is Activating at 1152 Kb/s
03/09/1998-17:13:32:SDSL: Deactivated
03/09/1998-17:13:32:SDSL: CPE is Activating at 1152 Kb/s
03/09/1998-17:14:11:SDSL: Deactivated
03/09/1998-17:14:12:SDSL: CPE is Activating at 384 Kb/s
03/09/1998-17:14:51:SDSL: Activated
03/09/1998-17:14:53:FRAMER: The framer is synchronized
03/09/1998-17:15:19:DOD: connecting to co @ 0*38 over ATM-VC/1
03/09/1998-17:15:35:DOD: link to co over ATM-VC/1 is now up
03/09/1998-17:15:57:SDSL: Line Rate at last activation saved
```

SDSL ?

Lists the supported keywords.

sdsl ?

Example: `sdsl?`

Response:

```
SDSL commands:
?             help        terminal
save          speed
```

SDSL SPEED

CO end: This command is used to set the speed manually on the CO end only.

CPE end: The router on the CPE end is always in auto-speed mode: it uses an auto-speed algorithm to attempt to match the CO speed. The command **sdsl speed noauto** is used to override auto-speed.

Note 1: The command **sdsl speed** (with no option) displays the current speed if the modem has activated successfully.

Note 2: **sdsl speed noauto** should be followed by the command sdsl save to become persistent across reboots.

Note 3: During auto-speed search, use the command **sdsl speed** <*speed*> to stop the search and restart it at the speed you just entered.

sdsl speed [192 | 384 | 768 | 1152 | noauto]

384	Default speed for the CO
192, 768, 1152	Authorized non-default speeds for the CO in Mbps
noauto	Used to override auto-speed on the CPE end

Example:

```
sdsl speed 1152
sdsl speed noauto
sdsl speed
```

SDSL SAVE

Used to save the SDSL-related changes across reboots.

sdsl save

Example: `sdsl save`

SDSL TERMINAL

The router is by default configured as the CPE. Use this command if you intend to configure the router as a Central Office equipment (CO).

sdsl terminal cpe is used to define the CPE (customer premise) end (default configuration).

sdsl terminal co is used to define the CO (central office) end.

sdsl terminal displays the current settings.

sdsl terminal [cpe | co]

co This option lets you define the router as the central office (CO)

Example: `sdsl terminal`

Response: `Customer Premises`

Example: `sdsl terminal co`

Dynamic Host Configuration Protocol Commands (DHCP)

The following DHCP commands allow you to:

- Enable and disable subnetworks and client leases
- Add subnetworks and client leases
- Set the lease time
- Change client leases manually
- Set option values globally, for a subnetwork, or for a client lease
- Enable/disable BootP
- Use BootP to specify the boot server
- Define option types

DHCP ?

Lists the supported keywords.

dhcp ?

Response:

```
Sub-commands for dhcp
?           help        set
list        bootp       clear
enable      add         del
disable     relay
```

DHCP ADD

This command is used to add a subnetwork, a client lease, or an option type.

dhcp add [<*net*> <*mask*>]| <*ipaddr*>	<*code*><*min*><*max*><*type*>

net	IP address of the subnetwork lease in the format of 4 decimals separated by periods.
mask	IP network mask, in the format of 4 decimals separated by periods.
ipaddr	IP address of the client lease in the format of 4 decimals separated by periods.
code	The code is user-defined can be a number between 128 to 254 or a keyword.
min	Minimum number of value(s).
max	Maximum number of value(s).
type	Byte | word | long | longint | binary | ipaddress | string

Examples: `dhcp add 192.168.254.0.255.255.255.0`
`(adds this subnetwork)`

`dhcp add 192.168.254.31`
(adds this client lease)

`dhcp add 128 1 4 ipAddress`
(adds this option type).

Note: In the above example, 128 allows IP addresses, the server has a minimum of one IP address, the server can have up to four IP addresses, and the type is “ipaddress”).

DHCP BOOTP ALLOW

This command allows a BootP request to be processed for a particular client or subnet.

dhcp bootp allow <*net*>|<*ipaddr*>

net IP address of the subnetwork lease in the format of 4 decimals separated by periods.

ipaddr IP address of the client lease in the format of 4 decimals separated by periods.

Example: `dhcp bootp allow 192.168.254.0`

DHCP BOOTP DISALLOW

This command is used to disallow a BootP request to be processed for a particular client or subnet.

dhcp bootp disallow <*net*>|<*ipaddr*>

net IP address of the subnetwork lease in the format of 4 decimals separated by periods.

ipaddr IP address of the client lease in the format of 4 decimals separated by periods.

Example: `dhcp bootp disallow 192.168.254.0`

DHCP BOOTP FILE

This command lets you specify the boot file name (kernel).

Note: The TFTP server IP address must also be set when the file is specified.

dhcp bootp file [<*net*>|<*ipaddr*>]<*name*>

net IP address of the subnetwork lease in the format of 4 decimals separated by periods.

ipaddr IP address of the client lease in the format of 4 decimals separated by periods.

name Name of the file to boot from; the default name for this file is KERNEL.F2K

Example: `dhcp boot file 192.168.254.0 Kernel.f2k`

DHCP BOOTP TFTPSERVER

This command lets you specify the TFTP server (boot server).

dhcp bootp tftpserver [<*net*>|<*ipaddr*>]<*tftpserver ipaddr*>

net IP address of the subnetwork lease in the format of 4 decimals separated by periods.

ipaddr IP address of the client lease in the format of 4 decimals separated by periods.

tftpserver ipaddr IP address of the TFTP server in the format of 4 decimals separated by periods
0.0.0.0 is used to clear the IP address of the server.

Example:
```
dhcp bootp tftpserver 192.168.254.7
dhcp bootp tftpserver 192.168.254.0 192.168.254.8
dhcp bootp tftpserver 192.168.254.21 192.168.254.9
dhcp bootp tftpserver 0.0.0.0
```

DHCP CLEAR ADDRESSES

This command is used to clear the values from a pool of addresses.

dhcp clear addresses <*net*>

net IP address of the subnetwork lease in the format of 4 decimals separated by periods.

Example:
```
dhcp clear addresses 192.168.254.0
```

DHCP CLEAR EXPIRE

This command is used to release the client lease. It then becomes available for other assignments.

dhcp clear expire <*ipaddr*>

ipaddr IP address of the client lease in the format of 4 decimals separated by periods.

Example:
```
dhcp clear expire 192.168.254.12
```

Note: The client does not get updated. It will still have the old value.

DHCP CLEAR VALUEOPTION

This command is used to clear the value for a global option, for an option associated with a subnetwork, or with a specific client.

dhcp clear valueoption [<*net*>|<*ipaddr*>] <*code*>

net IP address of the subnetwork lease in the format of 4 decimals separated by periods.

ipaddr IP address of the client lease in the format of 4 decimals separated by periods.

code Code can be a number between 1 and 61 or a keyword. Use the command **dhcp list definedoptions** to list the codes and keywords.

Examples:
```
dhcp clear valueoption 4
dhcp clear valueoption 192.168.254.0 7
dhcp clear valueoption 192.168.254.2 gateway
```

DHCP DEL

This command is used to delete a subnetwork lease, a specific client lease, or a code.

dhcp del <*net* |<*ipaddr*>|<*code*>

net IP address of the subnetwork lease in the format of 4 decimals separated by periods.

ipaddr IP address of the client lease in the format of 4 decimals separated by periods.

code The code is user-defined and can be a number between 128 to 254 or a keyword.

Examples:
`dhcp del 192.168.254.0`
(deletes this subnetwork)

`dhcp del 192.168.254.31`
(deletes this client lease)

`dhcp del 128`
(deletes this option with code 128)

DHCP DISABLE

This command is used to disable a subnetwork or a client lease.

dhcp disable all	<*net*>	<*ipaddr*>

all Disables all subnets.

net IP address of the subnetwork lease in the format of 4 decimals separated by periods.

ipaddr IP address of the client lease in the format of 4 decimals separated by periods.

Examples:
```
dhcp disable 192.168.254.0
dhcp disable 192.168.254.17
```

DHCP ENABLE

This command is used to enable a subnetwork or a client lease.

dhcp enable all	<*net*>|<*ipaddr*>

all Enables all subnets.

net IP address of the subnetwork lease in the format of 4 decimals separated by periods.

ipaddr IP address of the client lease in the format of 4 decimals separated by periods.

Examples:
```
dhcp enable 192.168.254.0
dhcp enable 192.168.254.17
```

DHCP LIST

This command lists global, subnetwork, and client lease information.

dhcp list | <*net*>|<*ipaddr*>

net IP address of the subnetwork lease in the format of 4 decimals separated by periods.

ipaddr IP address of the client lease in the format of 4 decimals separated by periods.

Examples:

To list global information, use:

```
dhcp list
```

Response:
```
       bootp server ................. none
       bootp file ...................
       DOMAINNAMESERVER (6) ......... 192.168.210.20 192.84.210.21
       DOMAINNAME (15) .............. flowpoint.com
       WINSSERVER (44) .............. 192.168.254.73
Subnet 192.168.254.0, Enabled
       Mask ......................... 255.255.255.0
       first ip address ............. 192.168.254.2
       last ip address .............. 192.168.254.253
       lease ........................ Default
       bootp ........................ not allowed
       bootp server ................. none
       bootp file ...................
    GATEWAY (3) .................. 192.168.254.254
   client 192.168.254.2, Ena, jo-computer, Expired
   client 192.168.254.3, Ena, Jo, 1998/5/16 11:31:33
```

To list information for client 192.168.254.3, use:

```
dhcp list 192.168.254.3
```

Response:
```
Client 192.168.254.3, Enabled
       lease ........................ Default
       expires ...................... 1998/5/16 11:31:33
       bootp ........................ not allowed
       bootp server ................. none
       bootp file ...................
       HOSTNAME (12) ................ JO
       CLIENTIDENTIFIER (61) ........ 1 2 96 140 76 149 180
```

To list information for the subnetwork 192.168.254.0, use:

```
dhcp list 192.168.254.0
```

Response:

```
Subnet 192.168.254.0, Enabled
       Mask ................. 255.255.255.0
       first ip address ...... 192.168.254.2
       last ip address ....... 192.168.254.253
       lease .................Default
       bootp .................not allowed
       bootp server ...........none
       bootp file .............
       GATEWAY (3) ............192.168.254.254
   client 192.168.254.2, Ena, Jo-computer, Expired
   client 192.168.254.3, Ena, Jo,  1998/5/16 11:31:33
```

DHCP LIST DEFINEDOPTIONS

This command lists all available predefined and user-defined options.

Note: For description of the predefined options listed below, please refer to RFC 1533. A predefined code can be a number between 1 and 61 or a keyword. A user-defined code can be a number between 128 and 254 or a keyword.

dhcp list definedoptions | <*code*> | <*string*>

code Predefined or user-defined number or keyword.

string Character string.

Examples:

To list all available options (they may be predefined as in the list below, and/or user-defined), use:

```
dhcp list definedoptions
```

Response:

```
code TIMEOFFSET (2), 1 occurrence, type LONG
code GATEWAY (3), 1 to 63 occurrences, type IPADDRESS
code TIMESERVER (4), 1 to 63 occurrences, type IPADDRESS
code NAMESERVER (5), 1 to 63 occurrences, type IPADDRESS
code DOMAINNAMESERVER code SUBNETMASK (1), 1 occurrence, type IPADDRESS-RESERVED
 (6), 1 to 63 occurrences, type IPADDRESS
code LOGSERVER (7), 1 to 63 occurrences, type IPADDRESS
code COOKIESERVER (8), 1 to 63 occurrences, type IPADDRESS
code LPRSERVER (9), 1 to 63 occurrences, type IPADDRESS
code IMPRESSSERVER (10), 1 to 63 occurrences, type IPADDRESS
code RESOURCELOCATION (11), 1 to 63 occurrences, type IPADDRESS
code HOSTNAME (12), 1 to 255 characters, type STRING
code BOOTFILESIZE (13), 1 occurrence, type WORD
code MERITDUMPFILE (14), 1 to 255 characters, type STRING
code DOMAINNAME (15), 1 to 255 characters, type STRING
code SWAPSERVER (16), 1 occurrence, type IPADDRESS
code ROOTPATH (17), 1 to 255 characters, type STRING
code EXTENSIONSPATH (18), 1 to 255 characters, type STRING
code IPFORWARDING (19), 1 occurrence, type BINARY
code NONCALSOURCERTE (20), 1 occurrence, type BINARY
```

```
code POLICYFILTER (21), 1 to 31 occurrences, type IPADDRESS
code MAXDGMREASSEMBLY (22), 1 occurrence, type WORD
code DEFAULTIPTTL (23), 1 occurrence, type BYTE
code PATHMTUAGETMOUT (24), 1 occurrence, type LONGINT
code PATHMTUPLATEAUTBL (25), 1 to 127 occurrences, type WORD
code INTERFACEMTU (26), 1 occurrence, type WORD
code ALLSUBNETSLOCAL (27), 1 occurrence, type BINARY
code BROADCASTADDRESS (28), 1 occurrence, type IPADDRESScode PERFORMMASKDSCVR (29), 1
occurrence, type BINARY
code MASKSUPPLIER (30), 1 occurrence, type BINARY
code PERFORMRTRDSCVR (31), 1 occurrence, type BINARY
code RTRSOLICITADDR (32), 1 occurrence, type IPADDRESS
code STATICROUTE (33), 1 to 31 occurrences, type IPADDRESS
code TRAILERENCAP (34), 1 occurrence, type BINARY
code ARPCACHETIMEOUT (35), 1 occurrence, type LONGINT
code ETHERNETENCAP (36), 1 occurrence, type BINARY
code TCPDEFAULTTTL (37), 1 occurrence, type BYTE
code TCPKEEPALIVEINTVL (38), 1 occurrence, type LONGINT
code TCPKEEPALIVEGARBG (39), 1 occurrence, type BINARY
code NETINFOSVCDOMAIN (40), 1 to 255 characters, type STRING
code NETINFOSERVERS (41), 1 occurrence, type IPADDRESS
code NETTIMEPROTOSRVRS (42), 1 occurrence, type IPADDRESS
code VENDORSPECIFIC (43), 1 to 255 occurrences, type BYTE
code WINSSERVER (44), 1 to 63 occurrences, type IPADDRESS
code NETBIOSTCPDGMDIST (45), 1 to 63 occurrences, type IPADDRESS
code NETBIOSTCPNODETYP (46), 1 occurrence, type BYTE
code NETBIOSTCPSCOPE (47), 1 to 255 characters, type STRING
code XWSFONTSERVER (48), 1 to 63 occurrences, type IPADDRESS
code XWSDISPLAYMANAGER (49), 1 to 63 occurrences, type IPADDRESS
code REQUESTEDIPADDR (50), 1 occurrence, type IPADDRESS-RESERVED
code IPADDRLEASETIME (51), 1 occurrence, type LONGINT-RESERVED
code OPTIONOVERLOAD (52), 1 occurrence, type BYTE-RESERVED
code MESSAGETYPE (53), 1 occurrence, type BYTE-RESERVED
code SERVERIDENTIFIER (54), 1 occurrence, type IPADDRESS-RESERVED
code PARAMREQUESTLIST (55), 1 to 255 occurrences, type BYTE-RESERVED
code MESSAGE (56), 1 to 255 characters, type STRING-RESERVED
code MAXDHCPMSGSIZE (57), 1 occurrence, type WORD-RESERVED
code RENEWALTIME (58), 1 occurrence, type LONGINT
code REBINDTIME (59), 1 occurrence, type LONGINT
code CLASSIDENTIFIER (60), 1 to 255 occurrences, type BYTE
code CLIENTIDENTIFIER (61), 2 to 255 occurrences, type BYTE
code NOTDEFINED62 (62), 1 to 255 occurrences, type BYTE
code NOTDEFINED63 (63), 1 to 255 occurrences, type BYTE
code NISDOMAIN (64), 1 to 255 characters, type STRING
code NISSERVERS (65), 1 to 63 occurrences, type IPADDRESS
code TFTPSERVERNAME (66), 4 to 255 characters, type STRING
code BOOTFILENAME (67), 1 to 255 characters, type STRING
code MOBILEIPHOMEAGNT (68), 0 to 63 occurrences, type IPADDRESS
code SMTPSERVERS (69), 1 to 63 occurrences, type IPADDRESS
code POP3SERVERS (70), 1 to 63 occurrences, type IPADDRESS
code NNTPSERVERS (71), 1 to 63 occurrences, type IPADDRESS
code WWWSERVERS (72), 1 to 63 occurrences, type IPADDRESS
code FINGERSERVERS (73), 1 to 63 occurrences, type IPADDRESS
code IRCSERVERS (74), 1 to 63 occurrences, type IPADDRESS
code STREETTALKSERVERS (75), 1 to 63 occurrences, type IPADDRESS
code STREETTALKDASRVRS (76), 1 to 63 occurrences, type IPADDRESS
```

To list options starting with the string “ga”, use:

```
dhcp list definedoptions ga
```

Response:

```
code,           number of values,      type of value
code GATEWAY (3), occurrence 1, type IPADDRESS
```

DHCP LIST LEASE

This command lists the lease time.

dhcp list lease

Example: `dhcp list lease`

Response:

```
Default lease time ......... 168 hours
```

DHCP RELAY

Lets the router relay DHCP or BootP requests to a DHCP server on the WAN, when a PC attempts to acquire an IP address using DHCP. This command disables the router’s DHCP server.

dhcp relay <*ipaddr*>

ipaddr IP address of the target router in the format of 4 decimals separated by periods.

Example: `dhcp relay 128.1.210.64`

DHCP SET ADDRESSES

This command is used to create or change a pool of IP addresses that are associated with a subnetwork.

dhcp set addresses <*first ipaddr*> <*last ipaddr*>

first ipaddr First address in a pool of addresses for a particular subnetwork.

last ipaddr Last address in a pool of addresses for a particular subnetwork.

Example: `dhcp set addresses 192.168.254.1 192.168.254.250`

DHCP SET EXPIRE

This command is used to manually change a client lease expiration time to a certain value.

Note 1: Changing a client lease time manually is rarely required.

Note 2: The client information does not get updated. It will still have the old value.

dhcp set expire <*ipaddr*><*hours*>|default|infinite

ipaddr IP address of the client lease in the format of 4 decimals separated by periods.

hours Lease time; minimum is 1 hour; 168 hours is the global default.

default Lease time that has been specified at the subnetwork or global level.

infinite No lease time limit; the lease becomes permanent.

Example: `dhcp set expire 192.168.254.18 8`

DHCP SET LEASE

This command is used to control lease time.

dhcp set lease [<*net*>|<*ipaddr*>]<*hours*>|default|infinite

net IP address of the subnetwork lease in the format of 4 decimals separated by periods.

ipaddr IP address of the client lease in the format of 4 decimals separated by periods.

hours Lease time; minimum is 1 hour; 168 hours is the global default

default Lease time that has been specified at the subnetwork or global level.

infinite No lease time limit; the lease becomes permanent.

Examples:

```
dhcp set lease 192.168.254.17 default
(sets client lease time to default)
dhcp set lease 192.168.254.0 infinite
(sets lease time to infinite for this subnet)
dhcp set lease 2
(sets global lease time to 2 hours)
```

DHCP SET OTHERSERVER

This command instructs the router's DHCP server to either continue or stop sending DHCP requests when another DHCP server is detected on the LAN. The default is **stop**.

dhcp set otherserver <*net*> continue|stop

net IP address of the subnetwork lease in the format of 4 decimals separated by periods.

continue The router's DHCP server continues sending DHCP requests, even if another DHCP server is detected on the LAN.

stop The router's DHCP server stops sending DHCP requests when another DHCP server is detected on the LAN.

Example: `dhcp set otherserver 192.168.254.17 stop`

DHCP SET MASK

Used to conveniently change the mask of a DHCP subnet without deleting and recreating the subnet and all of its entries.

dhcp set mask <*net*> <*mask*>

net IP address of the subnetwork lease in the format of 4 decimals separated by periods.

mask IP network mask, in the format of 4 decimals separated by periods.

Example: `dhcp set mask 192.168.254.0 255.255.255.0`

DHCP SET VALUEOPTION

This command is used to set values for global options, options specific to a subnetwork, or options specific to a client lease.

dhcp set valueoption [<*ipaddr*>|<*net*> <*code*> <*value*>....

ipaddr IP address of the client lease in the format of 4 decimals separated by periods.

net IP address of the subnetwork lease in the format of 4 decimals separated by periods.

code Code can be a number between 1 and 61 or a keyword. Use the command **dhcp list definedoptions** to list the codes and keywords.

value Can be a byte, word, signed long, unsigned long, binary, IP address, or string depending on the type of option.

Examples: `dhcp set value option 192.168.254.0 gateway 192.168.254.254`
(sets the value for an option associated with a subnetwork).

`dhcp set valueoption domainnameserver 192.168.254.2 192.168.254.3`
(sets a global value for the domain name server option)

`dhcp set valueoption 192.168.254.251 winserver 192.168.254.7`
(sets a value for an option associated with a specific client)

L2TP — Virtual Dial-Up Configuration (L2TP)

The following L2TP commands allow you to add, delete, and modify tunnels. L2TP router information that can be configured includes:

- Names
- Security authentication protocols and passwords
- Addresses
- Management of traffic performance

Note: Two **remote** commands specific to L2TP are also included in this section.

L2TP ?

Lists the supported keywords.

l2tp ?

Response:

```
L2tp Sub-commands:
?                       add                     del
forward                 list                    set
call                    close
```

L2TP ADD

This command creates a tunnel entry.

l2tp add <*TunnelName*>

TunnelName Name of the tunnel (character string). The name is case sensitive.

Example: `l2tp add PacingAtWork`

L2TP SET ADDRESS

Used to define the IP address of the other end of the tunnel, either the remote LAC or remote LNS.

CAUTION: If the IP address of the remote tunnel is part of a subnet that is also reached through the tunnel, a routing table entry for this address must be explicitly added. Normally, this routing entry will be added to remote entry, which has the default route.

Note 1: When a remote router tries to create a tunnel, the remote router's IP address is NOT authenticated .

Note 2: If this command is not used, then <*ipaddr*> defaults to 0.0.0.0 and this end cannot initiate the tunnel.

l2tp set address <*ipaddr*> <*TunnelName*>

ipaddr IP address of the remote LAC or LNS

TunnelName Name of the tunnel (character string). The name is case sensitive.

Example: `l2tp set address 192.168.100.1 PacingAtWork`

L2TP SET AUTHEN

Used to enable or disable authentication of the remote router during tunnel establishment using the CHAP secret, if it exists. If the remote router tries to authenticate the local end during tunnel authentication, the local router will always attempt to respond, provided a CHAP secret has been configured.

l2tp set authen on|off <*TunnelName*>

on Enables authentication

off Disables authentication

TunnelName Name of the tunnel (character string). The name is case sensitive.

Example: `l2tp set authen PacingAtWork`

L2TP CALL

This command is primarily used for “debugging” purposes and has the effect to establish a tunnel without creating a session.

l2tp call <*TunnelName*>

TunnelName Name of the tunnel (character string). The name is case sensitive.

Example: `l2tp call PacingAtWork`

L2TP SET CHAPSECRET

Creates a CHAP secret. This CHAP secret is used to authenticate the creation of the tunnel and is used for hiding certain control packet information. The LAC and the LNS can share a SINGLE CHAP secret for a given tunnel.

l2tp set CHAPSecret <*secret*> <*TunnelName*>

secret CHAP secret (character string) used to authenticate the creation of the tunnel

TunnelName Name of the tunnel (character string). The name is case sensitive.

Example: `l2tp set CHAPSecret PacingAtWork`

L2TP CLOSE

Used to close an L2TP tunnel and/or session.

l2tp close <*L2TP unit number*>|-*n*<*TunnelName*>|-*t*<*tunnelid*>|-*s*<*serialnum*>|-*c*<*callid*>

L2TP unit number

-*n TunnelName* Name of the tunnel (character string). The name is case sensitive.

-*t tunnelid* Local tunnel id

-*s serialnum* Serial number of the call within the tunnel

-*c callid* ID of the local call for the session

Note: Either <*TunnelName*> or <*tunnelid*> must be specified.

Example: `l2tp close -n PacingAtWork`

L2TP DEL

Used to delete a tunnel entry.

l2tp del <*TunnelName*>

TunnelName Name of the tunnel (character string). The name is case sensitive.

Example: `l2tp del PacingAtWork`

L2TP FORWARD

The router can be configured to forward all incoming calls to an LNS without answering the incoming call. This feature is normally used when the router is acting as a LAC or both a LAC/LNS.

Note: Only ONE tunnel entry can have this option set.

l2tp forward all|none <*TunnelName*>

all Forward all incoming call through the tunnel to an LNS

none No incoming calls are allowed to be forwarded through the tunnel to an LNS

TunnelName Name of the tunnel (character string). The name is case sensitive.

Example: `l2tp forward PacingAtWork`

L2TP LIST

The result of this command provides a complete display of the current configuration settings for tunnel(s), except for the authentication password/secret.

l2tp list |<*TunnelName*>|

TunnelName Name of the tunnel (character string). The name is case sensitive.

Example: `l2tp list PacingAtWork`

```
# l2tp list
INFORMATION FOR <pacingAtWork>
  type.............................. L2TPClient (LAC-will not dial)/LNS
All Incoming Calls Tunneled here..... no
  CHAP challenge issued............... yes
  hidden AVPs used.................... yes
  sequencing/pacing................... window pacing
    sequencing/pacing is.............. required
    window size for sequencing/pacing.. 10
  ip address.......................... 10.0.0.1
  Our host name....................... pacingAtHome

  ACTIVE TUNNEL....................... UNKNOWN
    current state..................... CLOSED
    LOCAL TUNNEL ID................... 1
    REMOTE TUNNEL ID.................. 0
    remote firmware................... 0
    remote ip address................. 10.0.0.1
    LAC SESSION serial number......... 0
      current state................... CLOSED
      LOCAL CALL ID................... 1
        local window size.............. 10
        sequencing/pacing.............. WINDOW PACING
          sequencing/pacing is......... required
      REMOTE CALL ID.................. 0
        remote window size............. 0
```

L2TP SET HIDDENAVP

Used to configure the router to protect some L2TP control information (such as names and passwords for a PPP session) using hidden AVPs. This command is often used to turn off hidden AVPs (no option), in cases where the other end of the tunnel does not support hidden AVPs.

l2tp set hiddenAVP yes|no <*TunnelName*>

yes This option lets the router hide AVPs. Yes is the default.

no This option disables hidden AVPs.

TunnelName Name of the tunnel (character string). The name is case sensitive.

Example: `l2tp set hiddenAVP yes PacingAtWork`

L2TP SET DIALOUT

Used to let LNS instruct the L2TP client to use an ISDN phone line to place a call on its behalf.

l2tp set dialout yes|no <*TunnelName*>

yes This option lets the router place outgoing calls.

no This option prevents the router from placing outgoing calls. No is the default.

TunnelName Name of the tunnel (character string). The name is case sensitive.

Example: l2tp set dialout yes PacingAtWork

L2TP SET OURPASSWORD

This command is used to specify the router's secret/password for PPP authentication on a per-tunnel basis.

l2tp set ourpassword <*password*> <*TunnelName*>

password Router's secret/password used for authentication when challenged by another router

TunnelName Name of the tunnel (character string). The name is case sensitive.

Example: `l2tp set ourpassword PacingAtWork`

L2TP SET OURSYSNAME

This command is used to specify the router's name for PPP authentication on a per-tunnel basis.

l2tp set oursysname <*name*> <*TunnelName*>

name Name of the router that is used for authentication when challenged by another router

TunnelName Name of the tunnel (character string). The name is case sensitive.

Example: `l2tp set oursysname myName PacingAtWork`

L2TP SET OURTUNNELNAME

This command creates local router's host name.

Note: If this command is not used, then, if specified, the <*name*> from the **l2tp set ourSysName** command or the <*name*> from the command **system name** <*name*> command is used.

l2tp set ourTunnelName <*name*> <*TunnelName*>

name Host name of the local router. This is the fully qualified domain name of the local router.

The name is case sensitive

TunnelName Name of the tunnel (character string). The name is case sensitive.

Example: `l2tp set ourTunnelName isp PacingAtWork`

L2TP SET REMOTENAME

This command creates the host name of the remote tunnel.

Note: If this command is not used, then <*TunnelName*> of the tunnel entry is used.

l2tp set remoteName <*name*> <*TunnelName*>

name Host name of the remote tunnel. This is the fully qualified domain name of the remote host.

TunnelName Name of the tunnel (character string). The name is case sensitive.

Example: `l2tp set remoteName isp PacingAtWork`

L2TP SET TYPE

Used to define the type of L2TP support for the tunnel. The router's role is defined on a per-tunnel basis.

l2tp set type all|lac|lns|l2tpclient|disabled <*TunnelName*>

all The router is configured to act as both a LAC/L2TP client and an LNS server.

lac The router is configured to act as a LAC for this tunnel.

lns The router is configured to act as a LNS for this tunnel.

l2tpclient The router is configured to act as an L2TP client for this tunnel

disabled The tunnel entry is disabled.

TunnelName Name of the tunnel (character string). The name is case sensitive.

Example: `l2tp set type l2tpclient PacingAtWork`

L2TP SET WINDOW

This command is used to enhance traffic performance in a tunneling environment. The command's options are used to affect the way incoming payload packets are processed. The router is configured with the following default options: sequencing, required, and size 10..

l2tp set window sequencing|pacing|nosequencing|optional|required|size <*TunnelName*>

sequencing Sequence numbers are placed in theL2TP payload packets. With this option, one end instructs the other end to send sequence packets. No acknowlegments are issued for received packets.

pacing Sequence numbers are placed in the L2TP payload packets. When a session is created, the router specifies a window size. Acknowledgements for received packets are issued.

nosequencing No sequence numbers are placed in the L2TP payload packets carrying the PPP packets. If the remote end carries out sequencing or pacing, the router can still send and receive sequenced packets.

optional Used to allow dynamic switching of a session from pacing or sequencing to nosequencing.

required Used to disable dynamic switching from pacing or sequencing to nosequencing.

size Used to control the window size of the receive window for receiving packets for sequencing or pacing, when a session is created. Size can be 0 if packet sequencing is being carried out. Must be a non-zero value for window pacing. Size must be less than or equal to 30.

TunnelName Name of the tunnel (character string). The name is case sensitive.

Example: `l2tp set window sequencing PacingAtWork`

REMOTE SETL2TPCLIENT

With this command, this remote is the path to the L2TP client and accepts tunnel calls. Use this command if you router acts as an LNS. You must also specify PPP authentication and IP routes for this remote.

remote setl2tpclient <*TunnelName*><*remoteName*>

TunnelName Name of the tunnel (character string) associated with the remote LAC. The name is case sensitive.

remoteName Name of the remote entry (character string). The name is case sensitive.

Example: `remote setl2tpclient PacingAtWork Router2`

REMOTE SETLNS

With this command, this remote is the path to the LNS and will forward the incoming call (which matches this remote entry) through the tunnel named <*TunnelName*>, if your router is the client.

Note: The remote entry must also have appropriate information such as PPP authentication, IP routing, IPX routing, bridging, or Caller ID.

remote setLNS <*TunnelName*><*remoteName*>

TunnelName Name of the tunnel (character string). The name is case sensitive.

RemoteName Name of the remote entry (character string).

Example: `remote setLNS PacingAtWork lnsServer`

Bridging Filtering Commands (FILTER BR)

Bridging filtering allows you to control the packets transferred across the router. This feature can be used to enhance security or improve performance. Filtering occurs based on matched patterns within the packet at a specified offset. Two filtering modes are available:

- Deny mode will discard any packet matched to the deny filter database and let all other packets pass.
- Allow mode will only pass the packets that match the allow filter database and discard all others.

Up to 40 deny and 40 allow filters can be activated from the filter database.

FILTER BR ?

Lists the supported keywords.

filter br ?

Response:

```
Bridge filter commands:
?                       add                     del
use                     list
```

FILTER BR ADD

Adds a bridging filter to the filtering database.

filter br add [*pos*] [*data*] allow | deny

pos Byte offset within a packet; number from 0-127

data Hex number up to 6 bytes

Example: `filter br add 12 8035 deny`
(This filter prevents forwarding of RARP packets across the bridge)

FILTER BR DEL

Deletes a bridging filter from the filtering database.

filter br del [*pos*] [*data*] allow | deny

pos Byte offset within a packet; number from 0-127

data Hex number up to 6 bytes

Example: `filter br del 12 8035 deny`

FILTER BR LIST

Lists the bridging filters in the filtering database.

filter br list

Example: `filter br list`

Response:
```
Allow Filter:

Deny Filter:
pos:12, len=2, <80><35>
```

FILTER BR USE

Sets the mode of filtering to either deny, allow, or none.

filter br use none | deny | allow

Example: `filter br use allow`

Save Configuration Commands (SAVE)

These commands can be used to save the entire configuration of parts of the router's configuration to FLASH memory. The parts of the configuration you can save include:

- System
- Ethernet LAN
- DHCP settings
- Remote Router Database settings
- Filters

SAVE ALL

Saves the configuration settings for the system, Ethernet LAN, DSL line, and remote router database into FLASH memory. Note that there is a time lag between the response issued by a save command and the time the data is actually stored in FLASH memory. Issue a **sync** command after a **save** command prior to powering off the router. This commits the changes to FLASH memory.

save all

Example: `save all`

SAVE ATM25

Saves the ATM configuration settings. All new entries and changed entries are erased from FLASH memory.

save atm25

Example: `save atm25`

SAVE DHCP

Saves the DHCP configuration settings into FLASH memory.

save dhcp

Example: `save dhcp`

SAVE DOD

Saves the current state of the remote router database. All new entries and changed entries are saved into FLASH memory.

save dod

Example: `save dod`

SAVE ETH

Saves the configuration settings for the Ethernet LAN into FLASH memory.

save eth

Example: `save eth`

SAVE FILTER

Saves the bridging filtering database to FLASH memory. A reboot must be executed to load the database for active use.

save filter

Example: `save filter`

SAVE SYS

Saves the name, message, and authentication password system settings into FLASH memory.

save sys

Example: `save sys`

Erase Configuration Commands (ERASE)

These commands can be used to erase the entire configuration or parts of the router's configuration from FLASH memory. The parts of the configuration you can erase include:

- System
- Ethernet LAN
- DSL and Remote Router Database settings
- DHCP settings
- Filters

Once you erase part of the configuration, you will need to reconfigure that part of the configuration entirely.

Important: All of the following **erase** commands require a ***reboot*** without a **save** command to take effect.

ERASE ALL

Erases the configuration settings for the system, Ethernet LAN, DSL line, DHCP, and remote router database from FLASH memory.

Note: There is a time lag between the response issued by the **erase** command and the time the data is actually deleted from FLASH memory. Issue a **sync** command after an **erase** command prior to powering off the router. This commits the changes to FLASH memory.

erase all

Example: `erase all`

ERASE ATM25

Erases the ATM configuration settings. All new entries and changed entries are erased from FLASH memory.

erase atm25

Example: `erase atm25`

ERASE DHCP

Erases the dhcp configuration settings. All new entries and changed entries are erased from FLASH memory.

erase dhcp

Example: `erase dhcp`

ERASE DOD

Erases the current state of the remote router database. All new entries and changed entries are erased from FLASH memory.

erase dod

Example: `erase dod`

ERASE ETH

Erases the configuration settings for the Ethernet LAN from FLASH memory.

erase eth

Example: `erase eth`

ERASE FILTER

Erases the current bridging filtering database from FLASH memory. This command requires a **reboot** (without a **save**).

erase filter

Example: `erase filter`

ERASE SYS

Erases the name, message, and authentication password system settings from FLASH memory.

erase sys

Example: `erase sys`

File System Commands

The file system commands allow you to perform maintenance and recovery on the router. These commands allow you to:

- Format the file system
- List the contents of the file system
- Copy, rename, and delete files

The router file system is DOS-compatible and the file system commands are similar to the DOS commands of the same name.

COPY

Copies a file from the source to the destination. This command allows you to update the router software level or to write configuration files to a TFTP server.

copy <*srcfile*> <*dstfile*>

srcfile Filename of the source file to be copied.

dstfile Destination filename from where the file is to be copied.

Example:
```
copy tftp@128.1.210.66:kernelnw kernel.f2k
```

Response:
```
Copying...

421888 bytes copied
```

A filename is either the name of a local file or a file accessed remotely via a TFTP server:

A local filename is in the format:

`yyyyyyyy.yyy.`
A remotely accessed filename is specified as:

`TFTP@xxx.xxx.xxx.xxx:yyyyyyyy.yyy`
where xxx.xxx.xxx.xxx is the (optional) TFTP server address and yyyyyyyy.yyy is the name of the file to be copied. If the TFTP server address is not specified, the address used is the one from which the router booted or the one permanently configured in the boot system. Issue a **sync** command after a **copy** to commit the changes to FLASH memory.

Caution: No warning message is issued if you copy over an existing file.

DELETE

Removes a file from the file system.

delete <*filename*>

filename Name of the file to be deleted. The filename is in the format xxxxxxxx.xxx.

Example: `delete kernel.f2k`

Response: `kernel.f2k deleted.`

DIR

Displays the directory of the file system. The size of each file is listed (bytes).

dir

Example: `dir`

Response:

```
SYSTEM   CNF  2304
ATM25    DAT  20
DHCP     DAT  1536
KERNEL   F2K  257014
IDL_7    AIC  14828
ASIC     AIC  14828
FILTER   DAT  1284
```

EXECUTE

This command is used to load batch files of configuration commands into the router. This allows for customization and simpler installation of the router. A script file can contain commands, comments (lines introduced by the # or; characters) and blank lines.

There are two kinds of script files:

- A one-time script that is executed on startup (only once).
- A group of commands that can be executed at any time from the Command Line Interface with the **execute** <*filename*> command.

One-time scripts are useful to execute the complete configuration process from a default (unconfigured) state.

execute <*filename*>

filename Name of the file to be executed.

Example: `execute script1`

FORMAT DISK

Erases and reformats the router file system. This command should **only** be used when the file system is unusable. If the router does not execute the POST test and software boot successfully, and the result of the **dir** command indicates the file system is corrupted, you may wish to reformat the disk, reboot the router, and recopy the router software.

format disk

Example: `format disk`

Response:

```
NEWFS: erasing disk...
NEWFS: fs is 381k and will have 762 sectors
NEWFS: 128 directory slots in 8 sectors
NEWFS: 747 fat entries in 3 sectors
NEWFS: writing boot block...done.
NEWFS: writing fat tables...done.
NEWFS: writing directory...done.
Filesystem formatted!
```

MSFS

Checks the file structure of the file system. This command performs a function similar to the DOS **chkdsk** command. The router analyzes the File Allocation Table (FAT) and produces a file system status report.

WARNING: When specifying **fix**, make sure that no other operation is being performed on the configuration files at the same time (by the Configuration Manager).

msfs [fix]

fix — If fix is specified, errors are corrected in the FAT. This option should *only* be used when an **msfs** command results in a recommendation to apply the **fix** option.

Example: `msfs`

Response:

```
Filesystem 0, size=825k:
Checking filesystem...
Checking file entries...
  SYSTEM   CNF ...   2304 bytes .. ok.
  ATM25    DAT ...     20 bytes .. ok.
  DHCP     DAT ...   1536 bytes .. ok.
  KERNEL   F2K ... 257014 bytes .. ok.
  IDL_7    AIC ...  14828 bytes .. ok.
  ASIC     AIC ...  14828 bytes .. ok.
  FILTER   DAT ...   1284 bytes .. ok.
  1097 fat(s) used, 0 fat(s) unused, 0 fat(s) unref, 534 fat(s) free
  561664 bytes used by files, 9728 bytes by tables, 273408 bytes free
```

RENAME

Renames a file in the file system to a new name.

rename <*oldName*> <*newName*>

oldName Existing name of the file to be renamed. The filename is in the format xxxxxxxx.xxx.

newName New name of the file. The filename is in the format xxxxxxxx.xxx.

Example: `rename ether.dat oldeth.dat`

Response: `'ether.dat' renamed to 'oldeth.dat'`

SYNC

Commits the changes to the file system to FLASH memory.

sync

Example: `sync`

Response: `Syncing  file systems...done.`

Warning: Syncing is not complete until you see 'done'.

Chapter 6. Managing the Router

This chapter describes the options available for booting software, how to upgrade the router with new releases of software, and explains the process for maintaining copies of configuration files.

Simple Network Management Protocol (SNMP)

SNMP, a member of the TCP/IP protocol suite, was designed to provide network management interoperability among different vendors' management applications and equipment. SNMP provides for the exchange of messages between a management client and a management agent. The messages contain requests to get or set variables that exist in network nodes, thus allowing a management client to obtain statistics, set configuration parameters and monitor events. These variables (or objects) are defined in Management Information Bases (MIBs), some of which are general or standard SNMP-defined bases. Other bases, Enterprise Specific MIBs are defined by the different vendors for specific hardware.

The router provides SNMP agent support and support for standard as well as Enterprise Specific MIBs. SNMP is also used internally for configuration of the router. The active SNMP agent within the router accepts SNMP requests for status, statistics and configuration updates. Communication with the SNMP agent occurs over the LAN or WAN connection.

The supported MIB and a description of their contents are listed in the following table:

MIB II	Internet-standard MIB contains only essential elements such as system, interface, addressing, protocol (IP, etc.) and SNMP objects
Bridge MIB	States/statistics (including spanning tree states) within bridging system
Ethernet MIB	State/statistics of Ethernet port (collisions, etc.)
IP Forwarding MIB	State of routing tables (updates MIB II)
PPP MIB For LCP	State/Statistics for each PPP link
Enterprise MIB for configuration	Router-specific objects for configuration purposes

Any management application using SNMP over UDP/IP has access to the local SNMP agent. SNMP network management tools vary but often have features to display network maps of SNMP nodes, poll nodes at intervals, trigger alarms on thresholds, graph or list node statistic counters, view and edit individual MIB variables and print reports.

An example of useful information that can be obtained from a remote SNMP client would be the current status of the router’s WAN link and Ethernet interfaces including: protocol (PPP, CSMA-CD), line speed, maximum frame (transmission unit) size, physical address, operating status, or packet traffic rates.

TELNET Remote Access

TELNET access to the router is supported. TELNET allows you to log in to the router as if you are directly connected through the Console port. In this manner you can issue commands, using the command line interface, to configure the router and perform status monitoring from any remote location. You can use one of the available TCP/IP packages containing the TELNET application. To access the router using TELNET, issue the appropriate command syntax and assign the IP address of the router. You are then directly connected to the router and can issue commands. When you wish to end the TELNET session, exit the application by entering 'logoff' or another appropriate command.

A system security timer will log off a Telnet session after 10 minutes of inactivity. For more information, refer to the **system securitytimer** command, page 133.

Use the command **system telnetport** to enable or disable TELNET access.

Client TFTP Facility

A client Trivial File Transfer Protocol (TFTP) facility is built into the router and is capable of reading from and writing to the network. A TFTP server must be properly configured to communicate with the router for file transfers to be successful. The client TFTP facility is employed to boot software from a TFTP server, perform software upgrades and copy configuration files to a TFTP server. A TFTP server is integrated into the Windows' Configuration Manager and can also be used as a standalone application.

TFTP Server

The TFTPD (Trivial File Transfer Protocol Daemon) program is installed on your PC as part of the DSL Tools software. TFTPD waits for incoming TFTP requests from TFTP clients. It will put or get a file to or from your computer's hard disk.

There is no security built into TFTPD, so it is important to specify a root directory where all the files that may be accessed are located. When a file is requested, it must be at or below this root directory on your directory tree or the request will be denied. If a TFTP client wants to put a file on your PC, then the file must already exist for writing.

The **Options** menu of the TFTPD program allows the user to configure additional parameters such as the number of retries and the time between retries. The root directory can also be specified from the **Options** menu.

The DOS command line usage for TFTPD is:

TFTPD rootdirectory

The TFTPD operational parameters are kept in the file ROUTER.INI in the form:

rootdir=rootdirectory

retries=maxtries

timeout=timeout

TFTPD is automatically called by BOOTP and Configuration Manager.

BootP Server

BootP is the Bootstrap Protocol server and is installed on your PC with the DSL Tools software.

The BootP Server waits for incoming BootP broadcasts from BootP clients. The server looks up the MAC addresses of the incoming BootP request in its database. If the Mac Address is found, the server normally responds to the requestor with an IP address, the IP address of a TFTP server and the name of a file to use for booting.

Boot Code

The router provides a number of maintenance options for booting router software. You can boot from the router's FLASH memory, the most common option. Or, you can boot across the LAN network from a TFTP server, perhaps to test a new level of router software before downloading to FLASH memory. You can also boot through a gateway to a WAN. The router allows you to set permanent network boot parameters used during network booting and enables you to temporarily override those parameters. Finally, the router lets you define the order in which the router boot procedures are performed. You can make changes to the boot procedures and specify network boot parameters by entering manual boot mode.

Note: This section provides Boot Mode information for models with configuration switches and models with a reset button (model 2210).

Manual Boot Menu

This information applies to most routers with Configuration (DIP) switches.

Note: For routers with a reset button, see Recovering Kernels for Routers with a Reset Button (models 2210), page 218.

The router, as received when shipped, is set for automatic boot from FLASH memory. If you wish to change the boot options to allow for network booting, change the order of boot procedures, or perform a manual boot, you must enter manual boot mode. Automatic and manual boot are controlled by Configuration Switches (on the back panel of the router). The options menu will be displayed if the router's kernel is missing.

Access Manual Boot Mode

1. Set Switch 6 **DOWN** for Manual Boot mode
2. Reboot the router by issuing the **reboot** command or powering up the router.

The router then displays this menu of options:

```
1. Retry start-up
2. Boot from Flash memory
3. Boot from network
4. Boot from specific file
5. Configure boot system
6. Set date and time
7. Set console baud rate
8. Start extended diagnostics
```

To Return to Automatic Boot Mode

1. When you are ready to return to automatic boot mode, set switch 6 **UP**

2. Reboot by selecting **1**, **2**, **3,** or **4**. Rebooting with switch 6 in the **UP** position will cause the router to boot router software automatically in the order and manner you have specified.

Option 1: Retry Start-up

When in Manual Boot mode, you can reboot the router in the boot procedure order by selecting option **1**, "Retry start-up". The boot procedure order is either one you have specified or the default order. The default order is to boot from FLASH memory and then the network (if defined). If you wish to boot from the network and/or alter the boot procedure order, refer to *Option 3: Boot from Network.*

Option 2: Boot from FLASH Memory

If you wish to perform a manual boot from FLASH memory, select **2** from the main boot procedure menu. The router will attempt to boot from FLASH memory. If unsuccessful, the router will return to manual boot mode. (When you first receive the router, the router defaults to booting from FLASH during power-up or automatic reboot.)

Option 3: Boot from Network

First, you have to define permanent network boot parameters using selection **5**. Then, select **3** from the main boot procedure menu to perform a manual boot from the network. The router will attempt to boot from the network using the permanent network boot parameters you have specified.

If you have not defined network boot parameters, the router attempts to locate a BOOTP or RARP server on the network.

BOOTP can be used to supply an IP address, a TFTP Server IP address, and a filename.

RARP is used to obtain an IP address, given the MAC address. The router assumes that the RARP server is also capable of performing the duties of a TFTP Server and will request the filename KERNEL.F2K or the filename assigned when setting permanent network boot parameters.)

If a BOOTP or RARP server exists and is properly configured with the router's MAC address, the router will boot from the network. If unsuccessful, the router will return to manual boot mode.

Option 4: Boot from Specific File

You can temporarily override permanent network boot parameters when performing a network boot. When the router is in Manual Boot mode, select option **4**, "boot from specific file", from the main boot procedure menu. Set the network boot parameters; the current default (permanent) parameters are as shown. After setting the parameters, hit the **return** key and the router will boot from the network using the temporary boot parameters. If unsuccessful, the router will return to manual boot mode.

Once you have installed router software on a network TFTP server, you can have the router boot across the LAN. Network booting requires three parameters:

- the boot IP address

- the TFTP boot server address
- the router software filename on the server

The boot IP address is the router LAN IP address used *during* the boot procedure. This address may differ from the LAN IP address that the router is ultimately assigned. This address is different so that a system can be booted from one subnetwork and then moved to its operational network, if necessary.

The boot IP address is of the form:
zzz.zzz.zzz.zzz.

The TFTP boot server address is specified as:
xxx.xxx.xxx.xxx (where xxx.xxx.xxx.xxx is the LAN IP address of the boot server)

The filename must be in the format:
yyyyyyyy.yyy (similar to the DOS filename format).

Note that once you have set a TFTP server address, it will be assigned to the router software TFTP facility. This server address will then be used whenever a server address is not explicitly specified, including when the **copy** command is in the form:

```
copy tftp:filename kernel.f2k
```

Option 5: Configure Boot System

1. If you wish to specify permanent network boot parameters, boot the router in Manual Boot mode.
2. Then select **5**, "Configure boot system", from the main boot procedure menu to set permanent values.
3. Select **2**[1], **3**, and **4** to set the three boot parameters described above. After setting permanent network boot parameters, you can change the boot procedure order and/or perform a manual boot from the network.
4. Select **4** to `"Boot through the IP gateway";` In this procedure, the router on the local LAN can boot from a boot server not connected directly. Instead, the path to the boot server can include other networks (including WAN, if adequate routers exist). The gateway must be located on the local LAN and reachable by the local router.

You can specify whether the router boots from FLASH first, a network TFTP server first, or never automatically reboots.

1. To set the order, select **1** under Configure boot system option **5**.
2. To boot from FLASH first, enter **1**; to boot from the network first, enter **2**. If you enter **3**, the router will always go into manual boot mode; i.e., you must select the boot procedure to be performed.

Option 6: Set Time and Date

To set the current time and date, boot the router in Manual Boot mode, and select **6** from the main boot procedure menu. Set the new date in the format **MM[/DD[/YY (or YYYY)]]**. Set the new time in military

1. To reset any parameter, press **`enter`** when prompted.

format **HH[:MM[:SS]])**. You are shown the current date and time. If you set the date to 0/0/0, the real-time clock will be disabled.

Note: This router is Y2K compliant. If you choose to only enter 2 digits for the year, values greater than 93 translate to 19xx. Values less or equal to 93 translate to 20xx. The router has a one-hundred-year date range (from 1994 to 2093).

If the date is set to 0, the real-time clock is disabled for long-term storage.

The time and date fields are overwritten by the GUI, when the router is configured by a PC. The time and date values are then read from the PC.

Option 7: Set Console Baud Rate

Select **7** to alter the baud rate that is used by the router to communicate over the Console port with the terminal emulation program. You can override the default rate of 9600. Remember to set the identical baud rate in your terminal emulation program.

Option 8: Start Extended Diagnostics

Manual boot mode allows you to run extended diagnostics. You may want to run extended diagnostics if you suspect a hardware problem. If you select **8** from the main boot procedure menu, you will see the following display:

```
[1] DRAM test
[2] Parity test
[3] POST firmware CRC test
[4] Real-Time Clock chip test
[5] Timers and Interrupts test
[6] Multi-port UART (internal loopback) test
[7] Multi-port HDLC (internal loopback) test
[8] SCC2 External Loopback test
[9] SCC3 External Loopback test
[a] SCC4 External Loopback test
[b] Ethernet Transceiver (internal loopback) test
[-] Deselect all tests
[+] Select all tests
[.] Run selected tests
[#] Enter debugger
[/] Exit extended diagnostics (reboot)
```

Enter the number of each test that you would like to run or select all tests. Then enter "**.**" to begin diagnostic testing. (All of the tests are automatically run when you power up or reboot the router.) A debugging mode is available for use primarily when you have encountered a serious problem, in consultation with customer support services.

Note: Boot diagnostics are only available on routers with the MC68EN360 processor.

Identifying Fatal Boot Failures

Fatal boot failures can be identified by the LEDs light patterns displayed on the front panel of the router.

Note: Normal LED states are described in the in the *Hardware Reference* section of the *Quick Start Guide*.

TEST, LNK, WAN, and LANT are used to display these fatal errors according to the following LED patterns:

0-0-0-G	CPM fail
0-0-G-0	Timer fail
0-0-G-G	Bad FCS
0-G-0-0	DRAM fail
0-G-0-G	Interrupt fail
0-G-G-0	SCC fail
Y-0-0-0	CPU step fail
Y-0-0-G	Ethernet loop fail
FG-0-0-*	Wait stuck in the boot menu. Kernel file could be missing (* green LED blinking very rapidly)
G-0-0-*	Green occasionally blinks off (at 10-second intervals). Issuing BootP requests

Where 0 = LED light is off

G = LED light is on, blinking green

FG = LED blinking fast

Y = LED blinking yellow

* = LED could be on, off, or blinking

Any other combinations of the 4LEDs flashing in a regular pattern will indicate an internal error. The router should be returned to the factory for repair or replacement.

Note: Non-fatal errors are not displayed by the LEDs, but prompt the system to print explanatory messages on the console.

Software Kernel Upgrades

Booting and Upgrading from the LAN

You can download a new version of the router software kernel using a TFTP server existing on the LAN. The following steps show you how to boot the router software from the network and copy the image from the network into the router's FLASH memory. When first connecting to the router, the GUI backs up all the files to a directory called Sxxxxx where x is the router's serial number.

Note: It is strongly suggested that you use the Configuration Manager's **Upgrade/Backup** tool to upgrade or backup the kernel. The Configuration Manager's tool is more convenient to use than the Command Line Interface.

Upgrade Instructions

*Read the following steps very **carefully***!

1. **WARNING:** Before performing this procedure, make sure that you can successfully boot from the network using the manual boot procedure option 3 or 4. Refer to the section *Option 3: Boot from Network*.

2. Copy the router software file KERNEL.F2K to a directory where it can be accessed by a TFTP server. The TFTP server must be on the same LAN as the target router; i.e., there must not be a router or gateway between the target system and the TFTP server. If the TFTP sever is not on the same network as the target router, enter the gateway in the boot menu as described in the previous section.

3. Log into the Command Line Interface.

4. Enter **reboot** using the Command Line Interface to synchronize the file system and reboot the router. Since the kernel is no longer stored in FLASH memory, the router will try to boot from the network. If you have never set permanent boot parameters, the router attempts to locate a BOOTP or RARP server. If the router successfully reboots from the server, go to step 7.

5. Select **4** to boot router software from the TFTP server using temporary network boot parameters. You are prompted for: the router's boot LAN IP address, the TFTP server's IP address, the load address and the filename of the router's kernel saved on the server. Note that the LAN IP address is the address to be used during the network boot and this may differ from the IP address ultimately assigned to the router. Enter the temporary network boot parameters (hit the **return** key for the load address). If all entered information is valid, the router will boot from the network. An example follows:

```
Enter selection: 4
  Enter my IP address:
128.1.210.65
  Enter server IP address:
128.1.210.70
  Enter load address [80100]:
  Enter file name: kernel.f2k
```

   Alternatively, select **5** to set permanent network boot parameters and then boot from the network with selection **3**. You would use this option if you wish to boot from the network for a period of time before copying the software to FLASH memory.

6. After the boot is complete, verify that the kernel is running successfully.

7. When you are satisfied that the new kernel is performing as expected, copy the kernel into FLASH memory in the router typing the following commands:

   copy tftp@xxx.xxx.xxx.xxx:sfilename kernel.f2k
   sync

   where xxx.xxx.xxx.xxx is the TFTP server IP address, SFILENAME is the server filename of the kernel and KERNEL.F2K is the name of the file loaded from FLASH memory by the boot procedure. If you do

not specify the server address, a permanent or more recent override TFTP server address will be used, if defined. Enter the **sync** command to commit the changes to FLASH memory.

WARNING: After the kernel is copied, DO NOT power down the router until you have either issued a **sync** or **reboot** command to reboot the router. Otherwise the file is not written to FLASH memory.

8. After successfully copying the kernel to the router, set Configuration switch 2 or 6 **UP** (if you have set it down)**,** and reboot the router from FLASH memory via the **reboot** command. If you have altered the boot procedure order in any way, reset to boot from FLASH memory first. Verify the software revision number by issuing the **vers** command.

   The system is now ready to be re-configured if necessary. The configuration files are unchanged by the upgrade process.

Upgrading from the WAN Line

You can download a new version of the router software kernel using a TFTP server over the WAN line. The following steps show you how to copy the software across the WAN line into the router's FLASH memory.

WARNING: Before performing this procedure, make sure that you can successfully access the software across the WAN line via a TFTP server.

1. Copy router software KERNEL.F2K to a directory where it can be accessed by a TFTP server.
2. Log in to the Command Line Interface.
3. Copy the kernel into FLASH memory in the router typing the following commands:

   copy tftp@xxx.xxx.xxx.xxx:sfilename kernel.f2k
   sync

   where xxx.xxx.xxx.xxx is the TFTP server IP address, `sfilename` is the server filename of the kernel and KERNEL.F2K is the name of the file. If you do not specify the server address, a permanent or more recent override TFTP server address will be used, if defined.

WARNING: After the kernel is copied, DO NOT power down the router until you have either issued a **sync** command or rebooted the router. Otherwise the file is not written to FLASH memory**.**

4. After successfully copying the kernel to the router, reboot the router from FLASH memory via the **reboot** command. If a problem occurs during the upgrading, try the command again (do not reboot until you have successfully copied the kernel.) If you have altered the boot procedure order in any way, be sure to reset to boot from FLASH memory first. Verify the software revision number by issuing the **vers** command.

   The system is now ready to be re-configured if necessary. The configuration files are unchanged by the upgrade process.

Backup and Restore Configuration Files

To successfully save configuration files to the server, the files to be saved to the server must already exist, be writeable by everyone. This restriction is part of the TFTP protocol. Also, all the files accessed by the TFTP server must be under a single "root" directory. Multiple sub-directories can exist below this root, but they must be created manually at the server. Neither the sub-directories nor the files can be created remotely.

Note: Don't forget to start the TFTP server from the **DSL Tools** menu.

The **copy** command is used to upload configuration files to the TFTP server where the destination is in the form:

tftp@xxx.xxx.xxx.xxx:filename.ext

Backup Configuration Files (Recommended Procedure)

1. Create a directory under the TFTP root directory corresponding to the system name you want to back up.
2. Create files called SYSTEM.CNF, DHCP.DAT, and FILTER.DAT in this subdirectory. The files can be empty or not, but should be writeable by everyone.

   To backup a copy of configuration files, enter

   copy system.cnf tftp@xxx.xxx.xxx.xxx:myname/system.cnf

   copy filter.dat tftp@xxx.xxx.xxx.xxx:myname/filter.dat

   copy dhcp.dat tftp@xxx.xxx.xxx.xxx:myname/dhcp.dat

   where **xxx.xxx.xxx.xxx** is the IP address of the TFTP server and **myname** the router name.

   Note: SYSTEM.CNF, FILTER.DAT, and DHCP.DAT are three key files that should be backed up. To see other files that you may also want to save, type the command **dir**.

Restore Configuration Files

To restore the configuration files, enter:

copy tftp@xxx.xxx.xxx.xxx:myname/system.cnf system.cnf

copy tftp@xxx.xxx.xxx.xxx:myname/filter.dat filter.dat

copy tftp@xxx.xxx.xxx.xxx:myname/dhcp.dat dhcp.dat

sync

FLASH Memory Recovery Procedures

Recovering Kernels for Routers with Configuration Switches

In the unlikely event that the FLASH file system becomes corrupted, you can take a number of steps to attempt to recover. Perform the following procedures in the order listed:

1. Try to repair the file system by issuing the **msfs** command. While logged in, issue a **sync** command followed by an **msfs** command. If the display shows that the file system is corrupted, verify that no other console (via TELNET) is currently modifying the file system with the **ps** command. Then attempt to repair the file system typing the following commands:

   msfs fix
   sync

2. If the file system is still corrupted; i.e., you cannot write a file, you will have to reformat the file system. First, attempt to save your configuration files as explained in the section Backup and Restore Configuration Files, page 216. Then, while logged in, enter the following commands:

format disk
save
copy tftp@xxx.xxx.xxx.xxx:kernel.f2k kernel.f2k
sync

The above assumes that the software presently running from RAM is correctly configured and still functional. The **save** command re-creates all the configuration files (except the FILTER.DAT file, which you may re-create manually by typing **save filter**). The **copy** command reinstalls the operational software on the FLASH file system and **sync** commits all this information to disk.

3. In the event that the software running from RAM is not functional enough to perform those steps, you will have to boot from the network using a TFTP server, as explained in the section *Software Kernel Upgrades, page 214*.

   If you cannot issue the **format** command as explained in the previous tip, you will have to erase the FLASH file system from the boot code.

   a. Flip configuration switch 6 to the **DOWN** position and reboot the router (by powering down and up again, for example).
   b. At the manual boot menu, enter **5** to select 5.`"Configure boot system"`, and enter the "magical" number 98. Then, move switch 6 back to its **UP** position.
   c. Reboot from the network following the steps described in the Software Upgrade Procedure. You will notice error messages indicating that the file system is not formatted. Then log in and enter:

      format disk

   d. Recreate the configuration files either by re-entering the information or by restoring them from a TFTP server. Re-install the operational software entering the command:

      copy tftp@xxx.xxx.xxx.xxx:kernel.f2k kernel.f2k

      This assumes that TCP/IP routing is enabled and that an IP address has been assigned to the Ethernet interface.

Recovering Kernels for Routers with a Reset Button (models 2210)

A router that fails to boot may be an indication that the kernel is corrupted.

In order to use the following recovery steps, you need to have a kernel for the particular router model. If you installed the DSL Tools and successfully connected to the router, an automatic backup process was started and saved a copy of the kernel and other files to the PC in a subdirectory under DSL Tools called Sxxxxxx, where xxxxxx is the serial number of the unit. The file needed for this recovery is called KERNEL.F2K.

Before proceeding with the recovery steps described below, make sure that the router has a good Ethernet connection to the PC. If a console cable is available, you may want to connect it and start a terminal emulator session to see the router's console messages.

Additionally, you may also want to check the LEDs' blinking patterns (on the front panel of the router). They may help identify the state of the router.

Recovery Steps Using BootP

If available, you may want to connect a console cable and start a terminal emulator session to see the router's console messages.

1. Make sure that the PC path and directory information to a valid kernel are correct.
2. Start **Configuration Manager** or **Quick Start** (refer to your Quick Start Guide).
3. Select Tools and BootP.
4. In the BootP dialog box, enter the following information:
   - The path to the kernel file
   - The serial number of the router
   - The IP address to be used for the boot.

   Note: This IP address needs to belong to the same subnet as your PC and not be used by another device. For a simple configuration, the IP address 192.168.254.254 will work if your PC has already received an IP address from the router when it was still functioning.
5. In the **BootP Setting** dialog box, click **OK**. Configuration Manager writes the above settings to a file called BOOTDBASE.TXT and calls the Bootp server.
6. Power off the router.
7. Insert a a small pen or pointed object into the small reset switch (unlabeled hole) on the back panel of the router (right of the Ethernet hub connector). With the object still inserted in the reset switch, power up the router. Wait until all the LED lights flash (about 10 seconds).
8. Once this is accomplished, the BootP server should see a BootP request and start the TFTPD server. The TFTPD server will send the kernel to the router.
9. Restart Configuration Manager and try to connect to the router. With the following instructions, you will attempt to write a new kernel to the flash system.
10. From Configuration Manager's Main Menu, select **Tools** and **Upgrade/ Backup**.
11. Click **Firmware** and the **Upgrade** button.
12. Select a kernel file and click **OK**.
13. Wait until the file is copied, and click **Yes** to reboot the rooter.

Recovering Passwords and IP Addresses

Routers with Configuration Switches

Recover a password: Set both switches 5 and 6 in the down position after the router has booted. With this step, the system password is overridden, thus allowing a forgotten password to be re-entered**.**

Recover an IP address: Connect to the console terminal and type the **eth list** command to find out what the router's IP address is**.**

Routers with a Reset Button (models 2210)

The following step will assist you in recovering the router's administrative password or IP address, should you forget them.

Push in the reset button and hold it for 3 second while the router is running. With this step, the following features are enabled for a length of 10 minutes:

- The system password can be overridden by using the router's serial number as a password.
- A DHCP client address is enabled or created, so that a connected PC can get an IP address from the router.

Batch File Command Execution

This feature is used to load batch files of configuration commands into the router. This allows for customization and simpler installation of the router. A script file can contain commands, comments (lines introduced by the # or; characters) and blank lines.

There are two kinds of script files:

- A one-time script that is executed on startup (only once).
- A group of commands that can be executed at any time from the Command Line Interface with the **execute** <*filename*> command.

One-time scripts are useful to execute the complete configuration process from a default (unconfigured) state.

The following steps describe how to proceed in order to create and execute a one-time script from the Quick Start application.

- Create the script on your PC using Notepad or other text editor. The command syntax can be found in the Command Line Reference manual or enter **?** on the router command line (assuming you have access to the Command Line with the console or with Telnet).
- Select the **Tools | Execute Script** menu item and choose the script file you just prepared. When you click **OK**, the script file is loaded to the router (under the name AUTOEXEC.BAT) and the router is restarted, thus executing the script.

Alternatively, you can manually transfer the script file from your PC to the router using the following method:

- Start the TFTP server on your PC and set the root directory where the script file is located
- Use the following command to copy the script file to the router file system:
  copy tftp@ <*PC_IP_address*>:<*PC_file*> <*router_file*>
- To process the commands in the script file, you can either reboot your router (if the script file was copied under the name AUTOEXEC.BAT onto the router) or use the command **execute <***file***>**.

NOTES: If present, the file AUTOEXEC.BAT is renamed AUTOEXEC.OLD before it is executed, so that it is only run once. If you clear the router configuration with the **Reset Defaults** button of the **Upgrade/Backup** tool or the **reboot default** command, the AUTOEXEC.OLD is renamed back to AUTOEXEC.BAT and re-run after the boot up, thus restoring your configuration.

You can include the commands **rename** <*autoexec.old*> <*autoexec.bat*> or **reboot** in a script file: there is no limitation on the commands that you might define in your scripts. The **rename** command is useful if you need the

script to execute on every startup, whereas the **reboot** command is useful to apply changes and have them take effect (almost) immediately. However, be aware of the following caution note.

Caution: If you create a one-time script file (copied to the router under the name AUTOEXEC.BAT), do not include both the following commands: **rename** <*autoexec.old*> < *autoexec.bat*> and **reboot.** This would result in an endless loop of starting the router, executing the script, restarting the router, re-executing the script.

Chapter 7. Troubleshooting

Software problems usually occur when the router's software configuration contains incomplete or incorrect information. This chapter discusses:

- Diagnostic tools that are available to help identify and solve problems that may occur with your router
- Symptoms of software configuration problems
- Actions for you to take and also lists system messages

Diagnostic Tools

Using LEDs

Most hardware problems can be diagnosed and solved by checking the LEDs on front panel of your router. The following table summarizes the normal LED sequence in the left column (5 consecutive states) from **Power On** to **Ready State**. The right column lists suggestions to problems reflected by an "abnormal" LED state (no progression to the next state).

Note that this normal progression involves:

PWR LED (Power LED)

TEST LED (Self-test indicator LED)

LINK LED (modem link)

If the Power (PWR) light is off:

- Check that the power cord is firmly plugged into the back panel of the router and the other end into an active AC wall or power strip outlet.
- Check that the power switch is turned on.

Normal LED Sequence	State Length	Problem If the LED sequence stops at this stage:
State 1 **Power ON** PWR - green TEST - amber LINK - off	5 sec	Hardware problem has been detected. Contact Technical Support.
State 2 All lights flash	1 sec	
State 3 PWR - green TEST - green LINK - off	5 sec	1. Check that the DIP switches are all up. 2. Check that the correct software was loaded.
State 4 PWR - green TEST - green LINK - amber	5 to 10 sec	1. Check your DSL cable. 2. Check the physical connection from your router to the DSLAM (Central Office). 3. Possible problem with DSLAM card.
State 5 PWR - green TEST - green LINK - green	Ready State	

Once the router is in **Ready State**, the other LEDs may indicate transmitting and receiving activity as follows:

The WAN LED indicates that the WAN is transmitting activity.

The LANT LED indicates that the Ethernet LAN is transmitting activity.

The LANR LED indicates that the Ethernet LAN is receiving activity.

History Log

The **History Log** utility is a troubleshooting tool which displays the router's activity. It can be accessed from a terminal emulation session (including Configuration Manager) or from TELNET.

Accessing History Log through TELNET

1. Click **Connect** from the menu, **Remote System,** and enter the router's IP address.
2. Click **Connect.**

Accessing History Log through the Configuration Manager

1. Select **Tools** and **Terminal Window** (the console cable is required).
2. Log in with your administration password into the router (e.g. **"admin"**).
3. Use the command **system history** to view the buffer contents.

Other Logging Commands

- If you wish to monitor your router activity at all times, use the command **system log start** to view a continuous log, using TELNET. (This command will not work in a Terminal Window session, but only from TELNET.)
- The command **system log status** is used to find out if other users, including yourself, are using this utility.
- To discontinue the log at the console, use the command **system log stop.**

When you exit TELNET, you automatically stop any logging programs running in that session.

Note: History Log is preserved across reboots, but not across power outages or power down.

Ping Command

You can verify IP connectivity to the router by running a **ping** command. You will probably find a ping utility bundled with your TCP/IP stack. In Windows 95 and Microsoft's TCP/IP 32-bit stack for Windows for Workgroups, the command is called `PING.EXE` and can be found in your Windows directory.

Note: Before using the **ping** command to troubleshoot, make sure that the PWR, TEST, and LINK lights and green.

Instructions

♦ 1. Start a DOS Window:

a. Select **Start** from the Windows 95 taskbar.

b. Select **Programs**.

c. Select **MS-DOS Prompt**.

♦ 2. Issue the PING Command:

In the DOS window, type the command:

`ping <``IP address`**`>`**

Example: ping 192.168.254.254

Interpretation and Troubleshooting

To isolate a problem with the TCP/IP protocol, perform the following three tests:

1. Try to **ping** the IP address of your PC. If you get a response back, proceed directly with step 2. If you don't get a response back, check that:

- The network adapter card is installed.
- The TCP/IP protocol is installed.
- The TCP/IP protocol is bound to the network adapter.

2. Try to **ping** the IP address of your router. If you get a response back, proceed directly with step 3. If you don't get a response back, the problem lies between your PC and router:

- Check the cables.
- Check the hub.
- Make sure that your PC and the local router are in the same IP subnetwork.

3. Try to **ping** the DNS server. Write the results down and call your Network Service Provider.

Investigating Hardware Installation Problems

Check the LEDs to Solve Common Hardware Problems

Please refer to this chapter's section entitled *Diagnostic Tools, Using LEDs,* for more information.

Problems with the Terminal Window Display

- Ensure your console is plugged in and turned on.
- Verify that you are on the right communications port (Com1, Com2).
- Check the configuration parameters for speed, parity, etc. Make sure the console is not in an XOFF state. Try entering a **`"ctrl q"`**.
- Verify that the RS232 device attached to the console is configured as a 'DTE'. If not, a crossover or null modem adapter is required.

Problems with the Factory Configuration

- Compare the router configuration with your router order.
- Verify that the model number is correct (displayed during the boot procedure). The model number (and serial number) is also displayed on the main window of Configuration Manager.

Investigating Software Configuration Problems

Problems Connecting to the Router

If you cannot connect your PC to the target router for configuration:

- For a LAN connection, verify that the router's IP address matches the IP address previously stored into the router's configuration.

  You must have previously set the router's Ethernet LAN IP address and subnet mask, saved the Ethernet configuration changes, and rebooted the router for the new IP address to take effect.

- Check that your LAN cable is pinned correctly and each pin end is securely plugged in.
  Note: If using a straight-through cable, the colors for pins 1, 2, 3, and 6 should match on both connectors. If using a crossover cable, the colors for pins 1, 2, 3, and 6 on one connector should match respectively 3, 6, 1, and 2 on the other connector.

- Make sure the PC and target router are on the same IP subnetwork or the target router is reachable through a router on your LAN. They can, however, be on different networks if IP routing is off.

- Check Network TCP/IP properties under Windows 95 and the control panel of the TCP/IP driver installed under Windows 3.1.

- Check if the LAN LED on the router's front panel blinks when "pinged".

- Check your Ethernet board IRQ settings: the PC's table may become confused. If so, reboot your PC.

Problems with the Login Password

You have been prompted for the login password and received the following message: "Login Password is invalid".

- Re-enter the correct password and press **enter**. Remember that the password is case-sensitive. Check that you are entering **admin** in lowercase and that the Caps key is not active.

- If you have forgotten the password, you must reset the login password. Refer to the User Guide, Appendix E, *Changing Configuration Switches*, and perform the following procedure:

  1. Move switches **5** and **6** down.
  2. Type `login <`***newpasswd***`>`**.** Password checking is overridden.
  3. Move switches **5** and **6** up**.**
  4. Complete any configuration update that caused the prompt for login.
  5. Change your login password to a new password.
  6. Store the configuration and reboot the router.

Note: If you have not reset switches **5** and **6** up and have rebooted, you will place the router in maintenance mode. Set switches **5** and **6** up and turn the power off and then on.

Problems Accessing the Remote Network

If Bridging

- Make sure to reboot if you have made any bridging destination or control changes.
- All IP addresses must be in the same IP subnetwork (IP is being bridged).
- Check that a bridging default destination has been configured and is enabled.
- Be sure to reboot if the bridging destination or status has been changed.
- Check that bridging is enabled locally (use the **remote listBridge** command).
- Verify that bridging is enabled by the remote router (use the **remote list** command).
- Verify that the authentication passwords are correct.
- Reboot your PC if you have Windows for Work Groups.
- In Windows 95, do not forget to declare shared disk directories. Check the sharing properties on your **C: drive .**
- In the Terminal Window, check that calls are answered from the remote router.
- Check also for any PAP/CHAP errors for the remote router.

If TCP/IP Routing

- Check that Ethernet LAN TCP/IP Routing has been enabled (**eth list** command).
- The IP addresses of the local and remote networks belong to different IP subnetworks.
- Make sure that there is an existing route to the remote network.
- Make sure that there is a route back from the remote network.
- There must be a Source WAN IP Address defined when using NAT.
- Check that, if required, the source and remote WAN IP addresses are on the same subnetwork
- Reboot if you have made any IP address, control or protocol option changes.
- Check that the IP address of the station/network connected to the LAN beyond the remote router is correct, as well as the associated subnet mask.
- If the remote router WAN IP address and subnet mask are required, check that they have been specified correctly.
- Check that a default route has been specified, if needed.
- Be sure to reboot if IP addresses, control or protocol option changes have been made.
- Check that you are using an Ethernet cable.
- Check that IP routing is enabled at both ends.
- The IP address must be within the valid range for the subnet.

- Verify that the IP and gateway addresses are correct on the PC.
- Windows 95 may remember MAC addresses: if you have changed MAC addresses, reboot the router and the PC.
- In Windows 3.1., check that the TCP driver is installed correctly. Ping (**ping** command) your PC's IP address from the PC.
- Successful "pinging" results let you know that the TCP driver is working properly.
- If you have changed an IP address to map to a different MAC device, and **ping** or IP fails, reboot your PC.
- Use the **iproutes** command, to verify which router's name is the default gateway (this cannot be 0.0.0.0).

If IPX Routing

- Check that IPX Routing has been enabled and the remote end is enabled for IPX routing
- Validate that the IPX WAN network number matches the remote router's WAN network number.
- Check that IPX SAPs correctly identify the servers and applications on the remote network and have valid network numbers, node numbers, etc.
- Check that every SAP has a router to its internal network.
- Check that the IPX Routes (network numbers, hops, and ticks) seeded into the routing table for network segments and servers beyond the remote router are correct.
- Validate that the IPX WAN network number matches the remote router's WAN network number.
- Check that the IPX Routes (network numbers, hops and ticks) seeded into the routing table for network segments and servers beyond the remote router are correct.
- Check that IPX SAPs correctly identify the servers and applications on the remote network and have valid network numbers, node numbers, etc.
- Be sure to reboot if IPX addresses, routes, SAPs or control has been changed.
- If the router fails to negotiate IPX:
- Make sure that at least one WAN number is not equal to zero at one end of the link.
- The server must have an IPX route to the remote LAN.
- The Novell server needs to have **burst mode** turned **on**.
- Large Internet packets have to be turned **on**:
- Novell 3.12 and later:
- Client needs VLM.EXE, net.cfg: large Internet packets=ON, Pburst=5
- If you can't see the server SAPs:
- Check the frame types using the **eth list** command and that they are the same on both routers.
- Check that the Ethernet cable is correctly plugged in.
- Make sure that the Novell server is up.

Incorrect VPI/VCI (ATM Routers)

If you are given an incorrect or no VCI/VPI number to use for the remote and need to determine what the possible value might be, refer to described page 167 under the command ***atom findpvc,*** for more information.

Problems Accessing the Router via TELNET

- Ensure that the router has a valid IP address.
- Check that the Ethernet cable is plugged in.

Problems Downloading Software

- Ensure that a TFTP server is properly set up to locate the router software
- Verify that the router is loading from the network and not from FLASH memory.

System Messages

System messages are displayed on the terminal and sent to a log file (if you have opened one). The messages listed in this section are time-stamped informational and error messages. The messages are in the following format:

dd+*hh*:*mm*:*ss*:*nn sysfunc*: *message*
where:

dd	date in xx/xx/xx format as specified during router initialization
hh	number of hours (military format)
mm	number of minutes
ss	number of seconds
nn	hundredths of seconds
sysfunc	software function
message	message

The following are some examples of the messages:

```
12/05/1997-16:31:17:ADSL: Startup initiated
12/05/1997-16:36:26:ADSL: Startup handshake in progress
```

Time-Stamped Messages

<router/user> didn't negotiate our IP address correctly
Explanation: The remote router did not negotiate the IP address options as was expected by the local router.

<router/user> terminated IPCP prematurely
Explanation: IP failed to negotiate. Try to change the remote or the source WAN IP address.

Far Avg SQ #: <2-digit number> dB [4-digit number]
Explanation: Message about the average signal quality for the remote router. This information appears during modem startup and should be ignored unless requested by Technical Support.

Authorization failed
Explanation: PAP cannot be negotiated.

Can't agree with <router/user> on what their IP address should be
Explanation: The IP address entry for the remote router in the remote router database does not match with what the local router expects.

Can't obtain an IP address from <router/user>: one is needed in single user mode
Informative message.

Can't supply an IP address to <router/user>
Explanation: The remote end requests an IP address from the local end, which cannot supply it.

Cannot remove SYSTEM.CNF
Informative message.

Connecting to <router/user> @ <number> over <link/number>
Explanation: The local router is trying to connect to the specified remote destination.

Data Mode
Explanation: The connection is established and operational.

Duplicate IPX route to <router/user>
Explanation: There exists two routes to the same IPX destination. One route needs to be removed.

Duplicate IPX SAP <SAP number> to <router/user>
Explanation: There exists two IPX SAPs for the same IPX destination. One SAP needs to be removed.

Duplicate route <IP route> found on remote <router/user>
Explanation: There exists two IP routes to the same IP destination. One route needs to be removed.

Idle
Explanation: data is not being transmitted.

IP is configured for numbered mode with <router/user>, but no address for it
Explanation: On one end of the connection, remote entries have been configured for numbered mode. On the other end, remote entries have been configured for unnumbered mode. Both ends cannot communicate with each other.

No Signal Detected -- Check WAN Cable!
Explanation: (SDSL-specific error message) Your SDSL router cannot establish connectivity. Check your physical line.

No system name known - using defaults
Explanation: The router does not have a system name. For PAP/CHAP negotiation, the router will use a default name and password.

Note: IPX is misconfigured for <router/user> - no IPX WAN network
Explanation: IPX WAN address is wrong or missing.

Note: There is no IPX route statically defined for <router/user>
Informational message.

PPP: Peer not negotiating <IP | BNCP | IPX | CCP> right now
Explanation: One end of the network is not negotiating the same protocol as the other end.

Remote <router/user> didn't accept our CHAP password

Informational message.

Remote <router/user> does not respond to LPC echo. Link closed
The connection was terminated.

Remote <router/user> on <channel> didn't authenticate in time
Explanation: PPP authentication protocol did not succeed.

Remote <router/user> refuses to authenticate
Informational message.

Remote <router/user> tried to use PAP when CHAP was expected
Explanation: The remote end negotiated PAP while its minimum security level in the remote database was set to CHAP.

Remote <router/user> used wrong password <CHAP | PAP>
Explanation: The remote end has used an invalid password during CHAP or PAP security authentication.

Remote didn't accept our CHAP password
Explanation: The router attempted CHAP security authentication but the remote end rejected the password.

Remote on <interface> didn't authenticate in time
Informational message.

Remote on <interface> rejected our password with PAP
Informational message.

Remote on <interface> refuses to authenticate with us
Explanation: The remote destination refused to participate in the PAP/CHAP authentication process.

Startup failed
Explanation: The ATM modem could not synchronize with the remote end. The user should call Technical Support.

Startup failed: failure code = <number>, Status [code]
Explanation: The ATM modem could not synchronize with the remote end. The user should call Technical Support

TelnetD
Explanation: Connection accepted. A remote configuration session has been established.

User <router/user> is disabled in remote database
Informative message.

User <router/user> not found in remote database <PAP | CHAP>
Explanation: The authentication is coming from an unknown remote router.

History Log

The **History Log** utility is a troubleshooting tool which displays the router's activity. It can be accessed from a terminal emulation session (including the Configuration Manager) or from Telnet. Follow the steps described below:

1. If accessing the logging utility through Telnet, enter the router's IP address and connect.
   If accessing the logging utility through the Configuration Manager, select **Tools** and **Terminal Window** (the console cable is required).

2. Login with your administration password into the router (e.g.`admin`).

3. Enter the command **system history** to view the buffer contents.

Other logging commands:

- If you wish to monitor your router activity at all times, use the command **system log start** to view a continuous log**.** This command will not work in a Terminal Window session, but only from Telnet.
- **system log status** is used to find out if other users, including yourself, are using this utility.
- To discontinue the log at the console, use the command **system log stop**.

When you exit Telnet, you automatically stop any logging programs running in that session.

Note: The log is preserved across reboots but not across power outages or power down.

How to Obtain Technical Support

Before you contact Technical Support, please have the following information ready:

- Router model number
- Router software version
- Date of purchase
- Type of Operating System (Windows 95, 98, NT, or Windows for Workgroups)
- Description of the problem

List of other equipment such as personal computers, modems, etc. and third party software you are using, including revision levels.Technical support, repair services, and spare parts are available through your FlowPoint

How to contact Technical Support in the U.S.	Addresses / Numbers
Telephone	1-408-364-8300
E-Mail	Support@flowpoint.com
Fax	1-408-364-8301
Address	FlowPoint Corporation 180 Knowles Drive, Suite 100 Los Gatos, CA 95030
Web Site	http://www.flowpoint.com

Distributor. Otherwise, FlowPoint can provide assistance in the U.S. FlowPoint Distributors are available to provide those services in many countries outside the U.S. Warranty repairs must be accompanied by dated proof of purchase.

Appendix A. Network Information Worksheets

To configure the target (local) router, you need to fill out one of the following blank worksheet(s) that applies to your Link Protocol/Network Protocol situation:

- PPP with IP Routing configuration
- PPP with IPX Routing configuration
- PPP with Bridging configuration
- RFC 1483/RFC 1490 with IP Routing configuration
- RFC 1483/RFC 1490 with IPX Routing configuration
- RFC 1483/RFC 1490 with Bridging configuration
- RFC 1483MER/ RFC 1490MER with IP Routing configuration
- FRF8 with IP Routing configuration
- A blank worksheet is also available to enter information needed to configure a Dual Ethernet Router with IP Routing enabled.

If you are connecting to more than one remote router:

You need to fill out one set of information for each remote router in the *Remote Routers* section of the worksheet.

If you are setting up both ends of the network:

You will need a mirror image of the information listed in your target router worksheet for configuring the router on the other end of the WAN link.

Note: You may want to review one of the sample configurations, refer to the *Sample Configuration* section in Chapter 5.

Configuring PPP with IP Routing

PPP with IP Routing		
STEPS	**COMMANDS**	**YOUR SETTINGS**
System Settings		
System Name	**system name** <*name*>	.....
System Message	**system msg** <*message*>	.....
Authentication Password	**system passwd** <*password*>	.....
Ethernet IP Address	**eth ip addr** <*ipaddr*> <*ipnetmask*> [<*port#*>]	.....
DHCP Settings	**dhcp set valueoption domainname** <*domainname*> **dhcp set valueoption domainnameserver**<*ipaddr*>	.....
Change Login	**system admin** <*password*>	.....
Remote Routers		
New Entry	**remote add** <*remoteName*>	.....
Link Protocol	**remote setProtocol PPP** <*remoteName*>	.....
PVC or DLCI	**remote setPVC** <*vpi number*>*****<*vci number*> <*remoteName*> **remote setDLCI** <*number*><*remoteName*>	.....
Security	**remote setAuthen** <*protocol*> <*remoteName*>	.....
Remote's Password	**remote setOurPasswd** <*passwd*> <*remoteName*>	.....
Bridging On/Off	**remote disBridge** <*remoteName*>	.....
TCP/IP Route Address	**remote addIproute** <*ipnet*> <*ipnetmask*> <*hops*> <*remoteName*>	.....
If NAT is enabled: To enable NAT -and- You may need to enter a Source WAN Port Address	**remote setIpTranslate** on <*remoteName*> **remote setSrcIpAddr** <*ipaddr*> <*ipnetmask*> <*remoteName*>	.....
If NAT is OFF: You may need to enter a Source WAN Port Address	**remote setSrcIpAddr** <*ipaddr*> <*ipnetmask*> <*remoteName*>	.....
IP and IPX Routing		
TCP/IP Routing (Internet Firewall)	**eth ip enable** **eth ip firewall** <on | off>	.....
IPX Routing	**eth ipx disable**	.....
Store **Reboot**	**save** **reboot**	

Configuring PPP with IPX Routing

PPP with IPX Routing		
STEPS	**COMMANDS**	**YOUR SETTINGS**
System Settings		
System Name	**system name** <*name*>	.....
System Message	**system msg** <*message*>	.....
Authentication Passwd	**system passwd** <*password*>	.....
Ethernet IP Address	**eth ip addr** <*ipnet*> <*ipnetmask*> [<*port#*>]	.....
DHCP Settings	**dhcp set valueoption domainname** <*domainname*>	....
Change Login	**dhcp set valueoption domainnameserver** <*ipaddr*>	.....
Ethernet IPX Network #	**system admin** <*password*>	.....
	eth ipx addr <*ipxnet*> [<*port#*>] **eth ipx frame** <*type*>	
Remote Routers		
New Entry	**remote add** <*remoteName*>	.....
Link Protocol	**remote setProtocol PPP** <*remoteName*>	.....
PVC or DLCI	**remote setPVC** <*vpi number*>*****<*vci number*> <*remoteName*> **remote setDLCI** <*number*> <*remoteName*>	.....
Security	**remote setAuthen** <*protocol*> <*remoteName*>	.....
Remote's Password	**remote setPasswd** <*password*> <*remoteName*>	.....
Bridging On/Off	**remote disBridge** <*remoteName*>	.....
IPX Routes Add	**remote addIpxroute** <*ipxNet*> <*metric*> <*ticks*> <*remoteName*>	.....
IPX SAPs Add	**remote addIpxsap** <*servicename*> <*ipxNet*> <*ipxNode*> <*socket*> <*type*> <*hops*> <*remoteName*>	....
	remote setIpxaddr <*ipxNet*> <*remoteName*>	
IP and IPX Routing		
TCP/IP Routing	**eth ip disable**	.....
IPX Routing	**eth ipx enable**	....
Store **Reboot**	**save** **reboot**	

Configuring PPP with Bridging

PPP with Bridging		
STEPS	**COMMANDS**	**YOUR SETTINGS**
System Settings		
System Name	**system name** <*name*>	.....
System Message	**system msg** <*message*>	.....
Authorization Password	**system passwd** <*password*>	.....
DHCP Settings	**dhcp set valueoption domainname** <*domainname*>	.....
	dhcp set valueoptiondomainnameserver <*ipaddr*>	.....
Change Login	**system admin** <*password*>	.....
Remote Routers		
New Entry	**remote add** <*remoteName*>	.....
Link Protocol	**remote setProtocol PPP** <*remoteName*>	.....
PVC or DLCI	**remote setPVC** <*vpi* number>*****<*vci number*> <*remoteName*> **remote setDLCI** <*number*> <*remoteName*>	.....
Security	**remote setAuthen** <*protocol*> <*remoteName*>	.....
Remote's Password	**remote setOurPasswd** <*password*> <*remoteName*>	....
Bridging On/Off	**remote enaBridge** <*remoteName*>	
IP and IPX Routing		
IP Routing	**eth ip disable**	.....
IPX Routing	**eth ipx disable**	.....
Store **Reboot**	**save** **reboot**	

Configuring RFC 1483 / RFC 1490 with IP Routing

RFC 1483 / RFC 1490 with IP Routing		
STEPS	**COMMANDS**	**YOUR SETTINGS**
System Settings		
System Message	**system msg** <*message*>	.....
Ethernet IP Address	**eth ip addr** <*ipnet*> <*ipnetmask*> [*port#*>]	.....
DHCP Settings	**dhcp set valueoption domainname** <*domainname*>	.....
Change Login	**dhcp set valueoption domainnameserver** < *ipaddr*>	.....
	system admin < *password*>	.....
Remote Routers		
New Entry	**remote add** <*remoteName*>	.....
Link Protocol/PVC[a] (for ATM routers)	**remote setProtocol RFC1483** <*remoteName*>	.....
	remote setPVC <*vpi number*> ***** <*vci number*> <*remoteName*>	
Link Protocol /DLCI[b] (for Frame Relay Routers)	**remote setProtocol FR** <*remoteName*>	.....
	remote setDLCI <*number*><*remoteName*>	.....
Bridging On/Off	**remote disBridge** <*remoteName*>	.....
TCP/IP Route Address	**remote addiproute** <*ipnet*> <*ipnetmask*> <*hops*> <*remoteName*>	.....
If NAT is enabled: To enable NAT -and- You must enter a Source WAN Port Address	**remote setIpTranslate on** <*remoteName*>	.....
	remote setSrcIpAddr <*ipaddr*> <*ipnetmask*> <*remoteName*>	.....
If NAT is OFF: You may need to enter a Source WAN Port Address	**remote setSrcIpAddr** <*ipaddr*> <*ipnetmask*> <*remoteName*>	.....
IP and IPX Routing		
TCP/IP Routing (Internet Firewall)	**eth ip enable** **eth ip firewall** <on | off>	.....
IPX Routing	**eth ipx disable**	.....
Store **Reboot**	**save** **reboot**	

a Enter this information if you are using RFC 1483 in an ATM environment.
b Enter this information if you are using RFC 1490 in a Frame Relay environment.

Configuring RFC 1483 / RFC 1490 with IPX Routing

RFC 1483 / RFC 1490 with IPX Routing		
STEPS	**COMMANDS**	**YOUR SETTINGS**
System Settings		
System Message	**system msg** <*message*>	.....
Ethernet IP Address	**eth ip addr** <*ipaddr*> <*ipnetmask*> [*port#*>]	.....
DHCP Settings	**dhcp set valueoption domainname** <*domainname*>	.....
	dhcp set valueoption domainnameserver <*ipaddr*>	.....
Ethernet IPX Network #	**eth ipx addr** <*ipxnet*> [>*port#*>]	.....
	eth ipx frame <*type*>	.....
Change Login	**system admin** <*password*>	.....
Remote Routers		
New Entry	**remote add** <*remoteName*>	.....
Link Protocol/PVC[a] (for ATM routers)	**remote setProtocol RFC1483** <*remoteName*>	.....
	remote setPVC <*vpi number*>*****<*vci number*> <*remoteName*>	.....
Link Protocol/DLCI[b] (for Frame Relay Routers)	**remote setProtocol FR** <*remoteName*>	.....
	remote setDLCI <*number*><*remoteName*>	.....
Bridging On/Off	**remote disBridge** <*remoteName*>	.....
IPX Routes Add	**remote addIpxroute** <*ipxNet*> <*metric*> <*ticks*> <*remoteName*>	.....
IPX SAPs Add	**remote addIpxsap** <*servicename*> <*ipxNet*> <*ipxNode*> <*socket*> <*type*> <*hops*> <*remoteName*>	.....
	remote setIpxaddr <*ipxNet*> <*remoteName*>	.....
IP and IPX Routing		
TCP/IP Routing	**eth ip disable**	.....
(Internet Firewall)	**eth ip firewall** <on | off >	.....
IPX Routing	**eth ipx enable**	.....
Store	**save**	
Reboot	**reboot**	

a Enter this information if you are using RFC 1483 in an ATM environment.
b Enter this information if you are using RFC 1490 in a Frame Relay environment.

Configuring RFC 1483 / RFC 1490 with Bridging

RFC 1483 / RFC 1490 with Bridging		
STEPS	**COMMANDS**	**YOUR SETTINGS**
System Settings		
System Message	**system msg** <*message*>	.....
DHCP Settings	**dhcp set valueoption domainname** <*domainname*>	.....
	dhcp set valueoption domainnameserver <*ipaddr*>	.....
Change Login	**system admin** <*password*>	.....
Remote Routers		
New Entry	**remote add** <*remoteName*>	.....
Link Protocol/PVC[a] (for ATM routers)	**remote setProtocol RFC1483** <*remoteName*> **remote setPVC** <vpi *number*>*****<*vci number*> <*remoteName*>	.....
Link Protocol /DLCI[b] (for Frame Relay Routers)	**remote setProtocol FR** <*remoteName*>	.....
	remote setDLCI <*number*><*remoteName*>	.....
Bridging On/Off	**remote enaBridge** <*remoteName*>	.....
IP and IPX Routing		
IP Routing	**eth ip disable**	.....
IPX Routing	**eth ipx disable**	.....
Store **Reboot**	**save** **reboot**	

a Enter this information if you are using RFC 1483 in an ATM environment.
b Enter this information if you are using RFC 1490 in a Frame Relay environment.

Configuring RFC 1483MER / RFC 1490MER with IP Routing

RFC 1483MER/RFC 1490MER with IP Routing		
STEPS	**COMMANDS**	**YOUR SETTINGS**
System Settings		
System Message	**system msg** <*message*>	.....
Ethernet IP Address	**eth ip addr** <*ipaddr*> <*ipnetmask*>[<*port#*>]	.....
DHCP Settings	**dhcp set valueoption domainname** < *domainname*>	.....
	dhcp set valueoption domainnameserver <*ipaddr* >	.....
Change Login	**system admin** <*password*>	.....
Remote Routers		
New Entry	**remote add** <*remoteName*>	.....
Link Protocol/PVC[a] (for ATM routers)	**remote setProtocol RFC1483MER** <*remoteName*>	.....
	remote setPVC <*vpi number*>*****<*vci number*> <*remoteName*>	.....
Link Protocol /DLCI[b] (for Frame Relay Routers)	**remote setProtocol MER** <*remoteName*>	.....
	remote setDLCI <*number*><*remoteName*>	.....
Bridging On/Off	**remote disBridge** < *remoteName*>	.....
TCP/IP Route Address	**remote addIproute** <*ipnet*> <*ipnetmask*><*ipgateway*> <*hops*> <*remoteName*>	.....
If NAT is enabled: To enable NAT,enter: -and- Enter a Source WAN Port Addr	**remote setIpTranslate on** <*remoteName*>	.....
	remote setSrcIpAddr <*ipaddr*> <*ipnetmask*> <*remoteName*> [c]	.....
If NAT is not enabled: You may need to enter a Source WAN Port Addr	**remote setSrcIpAddr** <*ipaddr*> <*ipnetmask*> <*remoteName*> [d]	.....
IP and IPX Routing		
TCP/IP Routing (Internet Firewall)	**eth ip enable** **eth ip firewall** <on | off>	.....
IPX Routing	**eth ipx disable**	.....
Store **Reboot**	**save** **reboot**	

a Enter this information if you are using RFC 1483 in an ATM environment.
b Enter this information if you are using RFC 1490 in a Frame Relay environment.
c The mask is the mask of the remote network.
d The mask is the mask of the remote network.

Configuring FRF8 with IP Routing

RFC 1483FR with IP Routing		
STEPS	**COMMANDS**	**YOUR SETTINGS**
System Settings		
System Message	**system msg** <*message*>	.....
Ethernet IP Address	**eth ip addr** <*ipaddr*> <*ipnetmask*> [<*port#*>]	.....
DHCP Settings	**dhcp set valueoption domainname** < *domainname*>	.....
	dhcp set valueoption domainnameserver <*ipaddr*>	.....
Change Login	**system admin** <*password*>	.....
Remote Routers		
New Entry	**remote add** <*remoteName*>	.....
Link Protocol/PVC	**remote setProtocol FRF8** <*remoteName*>	.....
	remote setPVC <*vpi number*>*****<*vci number*> <*remoteName*>	.....
Bridging On/Off	**remote disBridge** <*remoteName*>	.....
TCP/IP Route Address	**remoteaddIproute** <*ipnet*> <*ipnetmask*> <*hops*><*remoteName*>	.....
If NAT is enabled: To enable NAT -AND- You must enter a Source WAN Port Addr	**remote setIpTranslate on** <*remoteName*>	.....
	remote setSrcIpAddr <*ipaddr*> <*mask*> <*remoteName*> [a]	.....
If NAT is not enabled: You may need to enter a Source WAN Port Addr	**remote setSrcIpAddr** <*ipaddr*> <*mask*> <*remoteName*> [b]	.....
IP and IPX Routing		
TCP/IP Routing (Internet Firewall)	**eth ip enable** **eth ip firewall** <on | off>	.....
IPX Routing	**eth ipx disable**	.....
Store **Reboot**	**save** **reboot**	

a The mask is the mask of the remote network
b The mask is the mask of the remote network

Configuring a Dual Ethernet Router for IP Routing

This table outlines commands used to configure a Dual Ethernet router for IP Routing.

Dual Ethernet Router - IP Routing		
Steps	**Commands**	**Your Settings**
System Settings System Name	**system name** <*name*>	.....
System Settings Message	**system msg** <*message*>	.....
Ethernet Settings Routing/ Bridging Controls	**eth ip enable** **eth br disable**	....
Ethernet Settings ETH/0 IP Address ETH/1 IP Address TCP/IP default route address	**eth ip addr** <*ipaddr*> <*ipnetmask*> [<*port#*>] **eth ip addr** <*ipaddr*> <*ipnetmask*> [<*port#*>] **eth ip addroute** <*ipaddr*> <*ipnetmask*> <*gateway*> <*hops*> [<*port#*>]	....
DHCP Settings Define DHCP network for ETH/1 Create an address pool for ETH/1 DNS Domain Name DNS Server WINS Server Address	**dhcp add [<***net*> <*mask*>	<*ipaddr*>	<*code*> <*min*> <*max*> <*type*> **dhcp set addresses** <*first ipaddr*> <*last ipaddr*> **dhcp set valueoption domainname <***domainname***>** **dhcp set valueoption domainnameserver** <*ipaddr*> **dhcp set valueoption winsserver** <*ipaddr*>	....
Store **Reboot**	**save** **reboot**	

Appendix B. Configuring IPX Routing

IPX Routing Concepts

IPX Routing is established by entering all remote routers in the remote router database to which this router will connect.

1. For each remote router, enter network addresses and services that may be accessed beyond the remote router.
2. Also enter a network number for the WAN link.
3. After specifying the route addressing and services, you then enable IPX routing across the Ethernet LAN.

Static Seeding:

When IPX traffic is for network segments and servers beyond the remote router, the target router's routing information table must be statically seeded. Static seeding ensures that the target router connects to the appropriate remote router. After the link is established, RIP broadcast packets will dynamically add to the target router's routing table. Seeding the routing table is not necessary when a target router never connects; it will discover remote networks beyond the calling router as soon as RIP updates arrive (provided the remote router supports RIP). However, for watchdog spoofing to work, the remote IPX routes for network segments and servers should be defined.

Configure IPX Routing

Configuring your router for IPX routing can be rather complex. The following section will guide you through the configuration process. Remember that PPP Authentication configuration must be completed *before* attempting IPX routing configuration. The full router configuration for simple IPX routing includes the following:

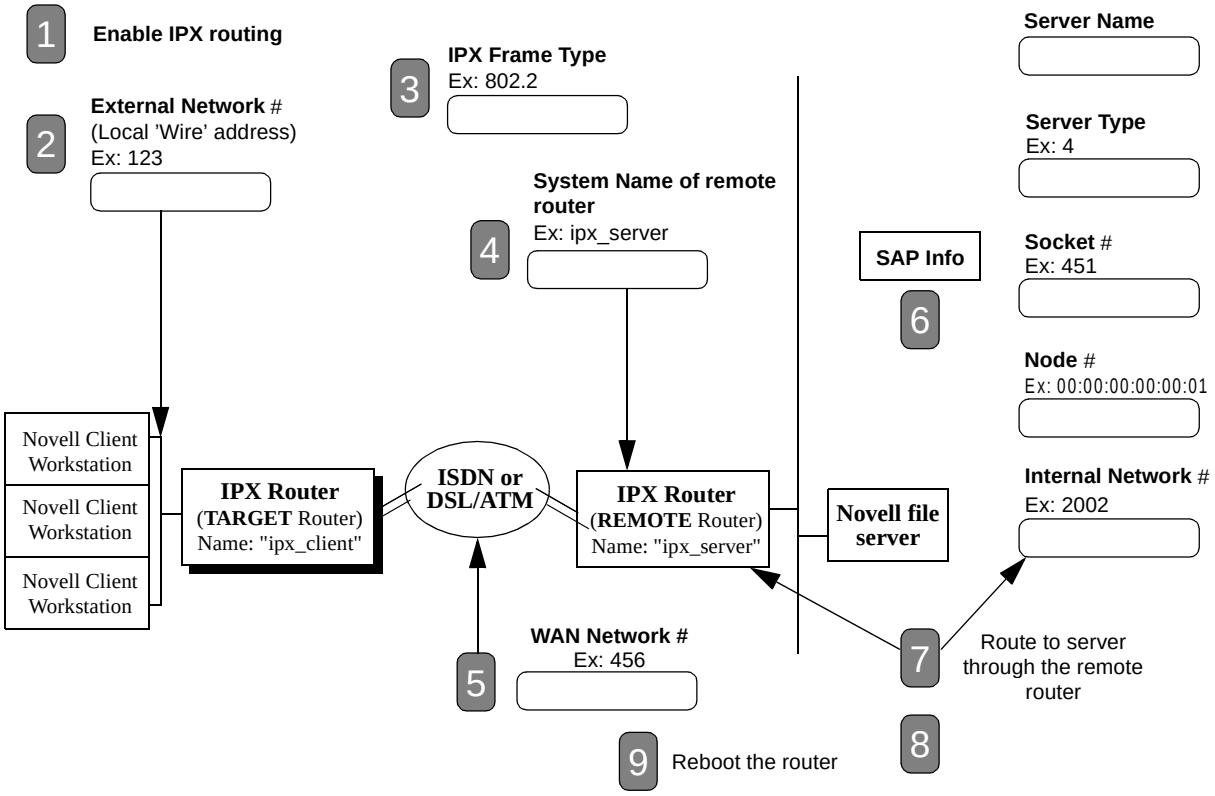
- PPP Authentication
- IPX routing (this section)

The following section, *Step 1: Collect your Network Information for the Target (Local) Router,* provides a configuration diagram and a command table to assist you with the configuration of the target router.

Step 2: Review your Settings lists the commands used to review the IPX configuration and provides a configuration example.

Step 1: Collect your Network Information for the Target (Local) Router

The remote side of the WAN link has all of the file and print services.
Enter the needed network information in the blank boxes of the diagram. Then match the boxes' numbers with the numbers in the Command Table below to configure the target router for IPX.



Command Table

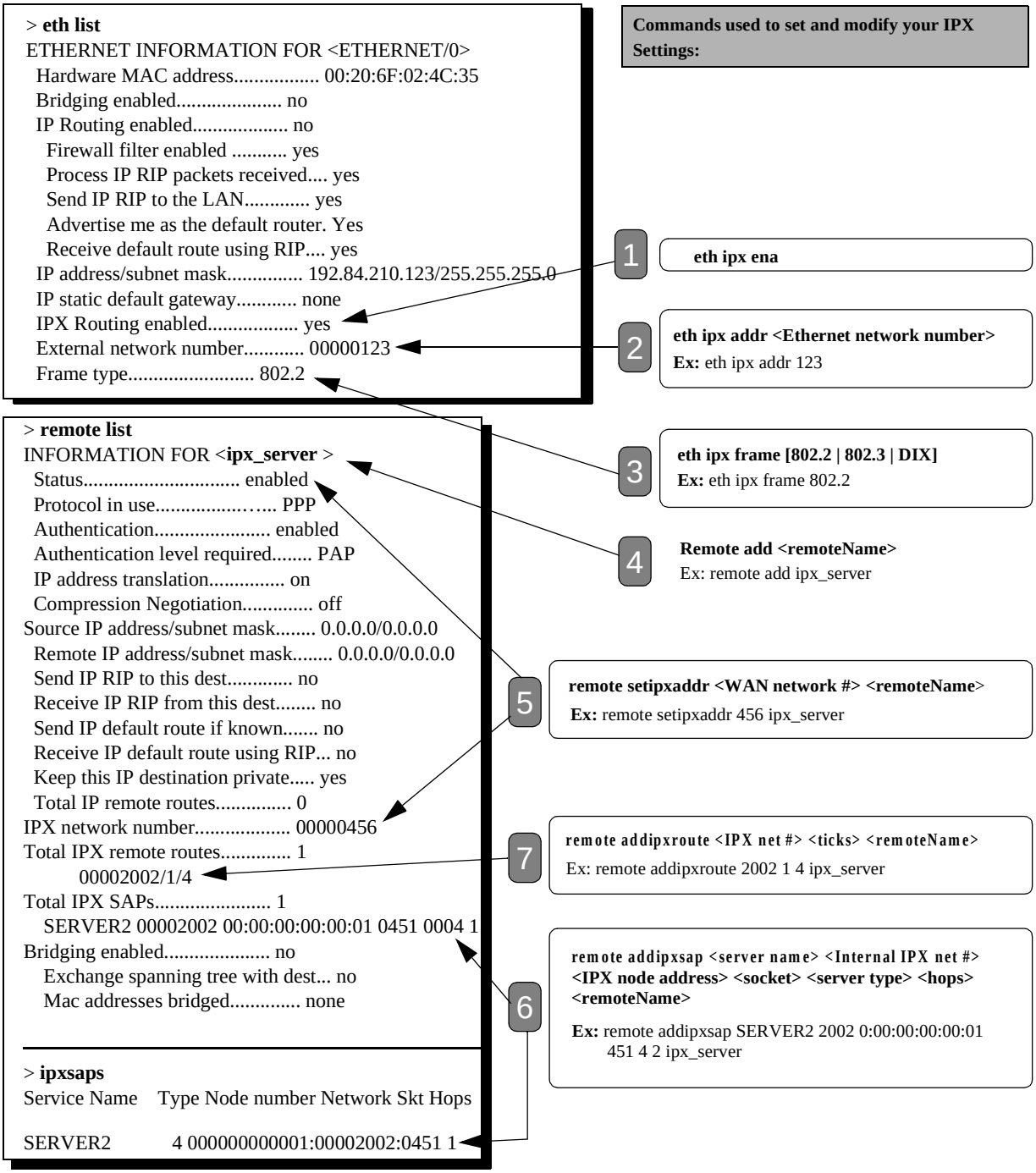
These commands are used to configure the Target (client-side) router (**ipx_client**).Log in with the password **admin.**

IPX Commands with examples	Ref #	Comments
eth ipx enable	**1**	Enable IPX Routing
eth ipx addr 123	**2**	Set the local 'wire' address
eth ipx frame 802.2	**3**	Set the Frame Type
remote add ipx_server	**4**	Add a connection name
remote setIpxaddr 456 ipx_server	**5**	Set the WAN network # (common to both sides)
remote addIpxsap SERVER2 2002 00:00:00:00:00:01 0451 4 1 ipx_server	**6**	Add a file server (SAP)
remote addIpxroute 2002 1 4 ipx_server	**7**	Add a route to the server
save	**8**	Save your settings
reboot	**9**	Reboot for changes to take effect

Step 2: Review your Settings

Commands used to review your IPX configuration:

- **eth list**
- **remote list**
- **ipxsaps**



Index