FlowPoint™

FlowPoint™ ISDN Router Family

Command Line Interface

February 1999

Copyright

Changes are periodically made to the information in this book. They will be incorporated in subsequent editions. FlowPoint may make improvements and/or changes in the product described in this publication at any time.

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 222-00550-01

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.

Part 68 Statement

This equipment complies with Part 68 of the FCC rules. On the back of this equipment is a label that contains, among other information, the FCC registration number for this equipment. If requested, this information must be provided to the telephone company.

This equipment has the FCC Digital Interface Code of 02IS5. The FCC Service Order Code is 6.OY.

The USOC jack for this equipment is RJ49C.

An FCC compliant telephone cord and modular plug is provided with this equipment. This equipment is designed to be connected to the telephone network or premises wiring using a compatible modular jack which is Part 68 compliant.

This equipment cannot be used on telephone company-provided coin service. Connection to Party Line Service is subject to state tariffs.

If this equipment causes harm to the telephone network, the telephone company will notify you in advance that temporary discontinuance of service may be required. If advanced notice is not practical, the telephone company will notify the customer as soon as possible. Also, you will be advised of your right to file a complaint with the FCC if you believe it is necessary.

The telephone company may make changes in its facilities, equipment, operations, or procedures that could affect the operation of the equipment. If this happens, the telephone company will provide advanced notice in order for you to make the necessary modifications in order to maintain uninterrupted service.

If trouble is experienced with this equipment, please contact Systems for warranty information. If the trouble is causing harm to the telephone network, the telephone company may request you remove the equipment from the network until the problem is resolved.

No repairs can be done by the customer.

It is recommended that the customer install an AC surge arrestor in the AC outlet to which this device is connected. This is to avoid damaging the equipment caused by local lightning strikes and other electrical surges.

Industry Canada

CS03 Statement

NOTICE: The Industry Canada label identifies certified equipment. This certification means that the equipment meets certain telecommunications network protective, operational and safety requirements as prescribed in the appropriate Terminal Equipment Technical Requirements document (s). The Department does not guarantee the equipment will operate to the user's satisfaction.

Before installing this equipment, users should ensure that it is permissible to be connected to the facilities of the local telecommunications company. The equipment must also be installed using an acceptable method of connection. The customer should be aware that compliance with the above conditions may not prevent degradation of service in some situations.

Repairs to certified equipment should be coordinated by a representative designated by the supplier. Any repairs or alterations made by the user to this equipment, or equipment malfunctions, may give the telecommunications company cause to request the user to disconnect the equipment.

User should ensure for their own protection that the electrical ground connections of the power utility, telephone lines and in ternal metallic water pipe system, if present, are connected together. This precaution may be particularly important in rural areas.

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, VCCI Class 2
Telecommunications: FCC Part 68, IC CS-03

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. It provides the steps and basic 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 described. The book also provides more detailed information about the system's bridging, routing, addressing, and security operation.

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 user's 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, installed the hardware using the User Guide or the Quick Start Guide, and reviewed the planning section in the User Guide. This document is divided into the following 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.

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.

Internet Quick Start Guide
Describes the configuration process involved in setting up a one-user Internet account.

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

The command Line Interface covers the following basic configuration topics:

- Setting of names, passwords, telephone numbers, and link parameters
- Management of bandwidth
- 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 ISDN subaddressing
- Configure analog services
- 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

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, ISDN BRI, and RS232 asynchronous modem hardware interfaces (depending on the router model).

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

The Router supports 802.3 Media Access Control layer for CSMA/CD Ethernet and ISDN Q.921 LAPD for ISDN.

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 synchronous Point-to-Point Protocol (PPP) for WANs and 802.2 for LANs.

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*.

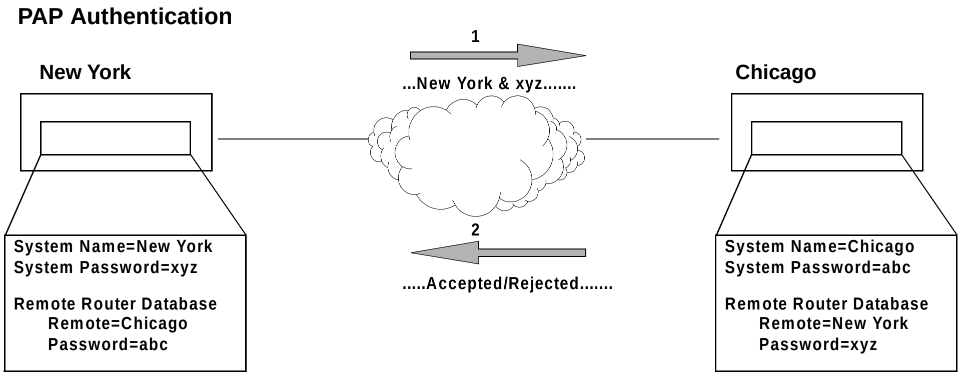
PAP/CHAP Security Authentication

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.

General Security Authentication Information

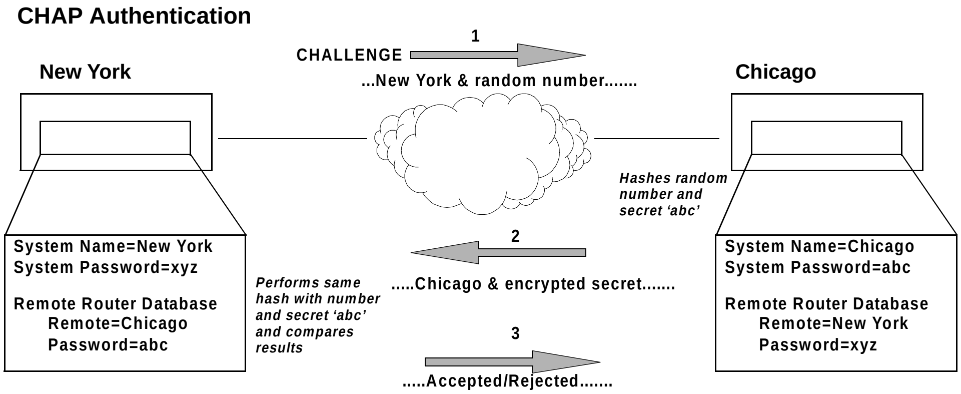
Security authentication may be required by the remote end. The following information describes how authentication occurs.

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.

Bandwidth-On-Demand

Bandwidth-on-Demand enables bandwidth management of up to two ISDN B-channels as the traffic load increases or decreases. This feature optimizes the use of dial-up WAN resources ensuring that a channel is used only when needed and released as soon as it is no longer required.

The Multi-Link Protocol for PPP (MLP) is used to implement this feature. MLP allows two B-channels to be bundled together to provide 128 Kb of data transmission capacity.

Bandwidth-on-Demand Configuration Settings

This feature is controlled by five configuration settings: Maximum and Minimum Links, Bandwidth Threshold, Fallback Interval[1] and Bandwidth Management Direction. These settings are defined for each remote site.

When traffic is sent or received, one or two channels can be used for the data transmission. The configuration setting, maximum links, determines whether a maximum of one or two B-channels are available for remote transmission. Minimum links determines whether one B-channel is permanently allocated for the remote site connection or a channel is only allocated when needed.

Initially a call is activated on one B-channel. When bandwidth utilization reaches the bandwidth threshold, the second B-channel is activated (if maximum links has been set to 2). Both channels are utilized until the bandwidth utilization drops below the threshold after a fallback interval. The fallback interval, in seconds, ensures that channels are not disconnected if traffic drops off for a small interval while overall traffic continues to be heavy.

When two channels are utilized and traffic decreases to the point that one channel can be released, the first channel acquired is released. Releasing this channel rather than the more recently acquired channel may result in some cost savings since the first interval of ISDN access time tends to be the most costly.

The technique used to calculate bandwidth utilization is a sliding window or moving average. Traffic volume is sampled once per second and a moving average is computed by assigning a weight of 20% to the last sample and a weight of 80% to the last average. After five seconds, no dependency is left on previous traffic. Using a moving average technique, the bandwidth utilization average does not drop off or spike upwards steeply if traffic decreases or increases during a few second interval (bursty traffic, for example). This ensures an efficient management of link resources.

Bandwidth management can be applied to incoming, outgoing or both directions of traffic between the router and the remote site.

[1] This configuration setting is fixed at five seconds.

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 1552 Novell IPX Control Protocol (IPXCP)
- RFC 1661 Point-to-Point Protocol (PPP)
- RFC 1723 RIP Version 2
- RFC 1962 PPP Compression Control Protocol (CCP)
- RFC 1969 ECP (option)
- 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.

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
POTS Configuration data

- **ISDN.DAT**
  ISDN Settings files containing:
  SPIDs
  DNs
  Switch type

- **DHCP.DAT**
  DHCP files

- **FILTER.DAT**
  Bridge filters

- **KERNEL.FP1**
  Router system software

- **ETH.DEF and ISDN.DEF**
  These two files are used by manufacturing to set default Ethernet address or switch types.

- **UK.FAC**
  For POTS routers: used to configure different ring codes

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 5. Managing the Router* or the*User Guide* for details on software upgrades, booting router software, copying configuration files and restoring router software to FLASH.

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. Configuring 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 section assumes that you have successfully followed the hardware installation instructions in the *Quick Start Guide* or the *User Guide.*

If you intend to use the Command Line Interface through the Configuration Manager application, it is assumed that you have installed the Configuration Manager software and can access the terminal window.

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.

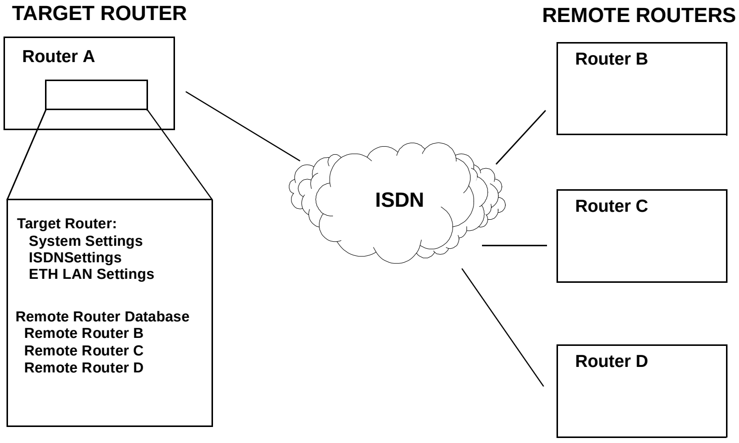
Remote router database
Database which resides in the target router and contains information about the remote routers to which the target router may connect.

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



Configuration Overview

You can configure all of the basic features (steps 1 through 15), save the entire router configuration directly into FLASH memory, reboot the router and then verify the configuration.

You can then configure the optional special features, save those settings, reboot and then test each function. Using the Command Line Interface, you will execute the following steps:

Basic Configuration

1. Log into the target router
2. Set target system settings
3. Set target system ISDN settings
4. Set target system Ethernet LAN addressing and DHCP (to access the Internet)
5. Add remote router(s) to the Remote Router Database
6. Configure dial-up link information
7. Configure bandwidth management
8. Set up security
9. Set up IP routing
10. Set up IPX routing
11. Set up bridging
12. Save the configuration and reboot the router
13. Verify the router's configuration
14. Logout

Optional Special Features

The following features are discussed in Chapter 3. *Configuring Special Features*.

- Ethernet Firewall and/or bridging filtering
- ISDN subaddressing
- CallerID security
- Call management
- Analog settings
- DHCP
- Network Address Translation (and host mapping)
- Management security
- Encryption
- IP filtering
- L2TP tunneling
- Software Options keys

Note: Each setting that you specify results in a dynamic update of the router's configuration, but some changes will not alter the active configuration until you save and reboot the router.

If you change any of the following settings, you must reboot the router for the changes to take effect:

- **Ethernet LAN**: Ethernet IP or IPX Address, TCP/IP Routing, IPX Routing
- **Bridging**: Bridging default destination, Filters
- **Remote Router**: TCP/IP Route Addresses, IPX Routes, IPX SAPs and Bridging control, enable, disable or add remote routers

Refer to Chapter 4. *Command Line Interface Reference* for usage conventions and a complete description of the commands mentioned in this chapter.

Network Information Tables

The following tables list the items you need to define or obtain to complete a basic configuration of the router (see Note 1). This information was described and illustrated on network information diagrams in the *User Guide*.

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

Note 1: For configuring special/advanced features, consult *chapter 3* of this manual.

Note 2: 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 ISDN link.

TARGET ROUTER (SOHO)

Target Router Settings	Item	Description
System Settings	System (Router) Name	Name used to identify this router; sent to other routers during PAP/CHAP security authentication
	System Message	Message saved in the router to be read by a system administrator (optional)
	DHCP Settings	DNS, Domain Name, Address Pool
	Dial Authentication Password/ Secret	This router's password used for authentication when the router dials out to other routers or is challenged
ISDN Settings	ISDN Line Numbers (supplied by the service provider)	SPIDs and Directory Numbers for one or two ISDN B-Channels on this router
	ISDN switch type	Type of switch
If running IP: Ethernet IP Settings	Ethernet IP Address and Subnet Mask	Address and Subnet Mask for Ethernet Port Connection
	LAN gateway address	Ethernet default gateway for packets that do not have a destination specified
	Ethernet LAN IP Routing On/Off	TCP/IP routing to all destinations On or Off
	Ethernet LAN IP Internet Firewall	Internet Firewall On or Off
	Ethernet LAN IP Options	Transmit/Receive RIP packets/routes and advertise as default route
If running IPX: Ethernet IPX Settings	Ethernet IPX Address and frame type	Network Number for Ethernet LAN connection
	Ethernet LAN IPX Routing ON/ OFF	IPX routing to all destinations On or Off

REMOTE ROUTER (HQ) – This information is defined in the Remote Router Database

Remote Router Settings	Item	Description
Dial Up Settings	ISDN Phone Numbers Disconnect Timer	ISDN Phone Numbers for one or two B-Channel(s) Disconnect Line on Inactivity
Bandwidth Management	Maximum Links Minimum Links Threshold Bandwidth Direction	Maximum links (up to 2 ISDN lines) Minimum links (up to max links) % threshold to access second channel Management on IN|OUT|BOTH
Security	Minimum Authentication Protocol Password/Secret Unique system override password	PAP|CHAP|NONE minimum protocol required for remote router Remote router's password used for authentication of target router Password used by remote router for authentication of target router
If running IP: **TCP/IP Routing**	IP Address, Subnet Mask, and Metric Remote Router WAN IP Addr/Subnet Mask[a] Target WAN IP Address and Subnet Mask[b] IP RIP options	IP Address, Subnet Mask of remote network/station beyond the remote router and route efficiency metric IP Address and Subnet Mask of the Remote Router IP Address and Subnet Mask of the local end of the WAN link Transmit/receive RIP, default routes
If running IPX: IPX Routing	IPX Routes: Network Number, Hop Count, and Ticks IPX SAPs: Server Name, Network Number, Node Number, Socket Number Server Type, Hop Count WAN Network Number	IPX Network Number, Hop Count and Ticks for stations/ nodes beyond the remote router Information defining application services available on stations/nodes beyond the remote router Network Number for the WAN link between target and remote router
If running Bridging: Bridging	Default Destination Remote MAC address(es) Bridging On/Off Spanning Tree Protocol	Default outbound destination Remote bridging addresses to seed bridging table Enable/Disable bridging Use Spanning Tree Protocol

[a] Used only in PPP numbered mode of addressing

[b] Used only in PPP numbered mode of addressing

Basic Configuration Steps

Step 1. Log into the Target Router

Log in with the following command:

login <*password*>

where *password* is an administration password. The default password is **admin**. The password can be reset using the **system admin** command.

The login password is required if you intend to modify the router's configuration settings. This security feature prohibits unauthorized write access to the router's configuration. If you do not log in with the write enable login password, you are prevented from issuing any command that changes the router's configuration and from rebooting the router. You will receive the message 'command not authorized'.

Step 2. Set Target System Settings

Enter information about the target router you are configuring and adding to your network. This information includes the system administration password, system name, and optional message and dial authentication password.

Set the System Administration Password

If you want to change the login password from the default **admin**, enter the following command:

system admin <*password*>

where *password* is the new administration password.

Set the System Name

The system name is required. This name is sent to other routers during authentication. Set the name of the target router using the command:

system name <*name*>

where *name* is a case-sensitive character string used to identify the router. 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 type anything after **system name**, the characters will be taken as the new name. If you wish to present a different name to each remote router, use the command:

remote setOurSysName <*name*> <*remoteName*>

Set a System Message

You may enter an optional message which is saved in the router. The message is useful for specifying, for example, the name of the person configuring this router and the last changes made. Enter the command: **system msg** <*msg*>

where *msg* is a character string. Space characters are not allowed within the message; you may use underscore characters instead.   If you do not enter a message following **system msg**, the current message is displayed (underscores are converted to spaces.)

Set the Dial Authentication Password

The target router's dial authentication password is used for authentication when the target router dials out to other routers or is challenged by them. To set the password, enter:

system passwd <*password*>

where *password* is a case-sensitive character string. A new password overrides the previous one set. Existing passwords cannot be displayed.   If you wish to set a unique password used *only* when the router dials to a specific remote router, you must also use the command:

remote setOurSysName <*name*> <*remoteName* >

To list the system settings, enter the command **system list**

The following is typical output from this command:

```
GENERAL INFORMATION FOR <HQ>
 System started on.................... 6/17/1998 at 12:22
  Authentication override.............. none
  Caller ID Security selected.......... none
  Receive Data Call as Voice........... no
  WAN to WAN Forwarding................ yes
  BOOTP/DHCP Server address............ none
  Telnet Port.......................... default (23)
  SNMP Port............................ default (161)
  System message: HQ sample configured June98
```

Step 3. Set Target System ISDN Settings

CAUTION: U.S. ROUTERS ONLY
1) You MUST configure the ISDN Switch Settings parameters FIRST. 2) Then, connect the ISDN line. Failure to follow these directions will cause the Central Office switch to behave erratically for some time.

Set ISDN Switch Type

If the router supports ISDN, you must enter ISDN line information. Specify the telephone switch type your ISDN service provider is using with the following command:

isdn set switch <*switchType***>**

where *switchType* is one of the following:

NTT	Nippon Telegraph and Telephone (NTT)
KDD	Kokusai Denshin Denwa., Ltd.
5ESS	AT&T 5ESS with Custom Software
Auto-5ESS	Auto-SPID detection for that switch
DMS 100	Northern Telecom DMS-100
Auto-DMS 100	Auto-SPID detection for that switch
NI1	National ISDN-1-compliant switches
Auto-NI1	Auto-SPID detection for that switch
NET3	European ISDN/ETSI
NET3SW	Swiss NET3 variant
HSD64	64 Kb permanent connection
HSD128	128 Kb permanent connection
HSD144	144 Kb permanent connection

Set SPIDs and Directory Numbers

The service provider may assign to you none, one, or two SPIDs and/or DNs for identifying the ISDN line and devices. This varies by service provider and country. Refer to the *User Guide's Chapter 1. ISDN and Ordering Issues*.

DNs

Enter directory numbers corresponding to the ISDN B-channels with the following command:

isdn set dns [*number*] [*number*]

where the first defined <*number*>is the first or only directory number and the second defined <*number*> is the second directory number. One ISDN directory number may be assigned for each B-channel of the ISDN BRI line, one directory number may be assigned for both channels, or directory numbers may not be provided at all. The SPID may be the Directory Number extended with additional digits.

SPIDS

❖ ***Manual Configuration***

Enter ISDN Service Provider IDs (SPIDs) with the following command:

isdn set spids [*number*] [*number*]

where the first defined [*number*] is the first or only SPID number and *t*he second defined [*number*] is the second SPID number. One ISDN SPID may be assigned for each B-channel of the ISDN BRI line, one SPID may be assigned for both channels, or SPID numbers may not be provided at all.

❖ ***Auto SPID Configuration***

The router can be configured to detect the ISDN SPIDs based on the switch type and the user's Directory Numbers. It may take up to 2 minutes to determine the SPIDs. If valid SPIDs are found, they are saved to flash automatically. The following commands automatically enable Auto-SPID detection. Make sure to set your switch type to one of the "auto" types (Auto-DMS 100, Auto-NI, Auto-5ESS**).**

isdn set switch <*switchType*>

Connect your ISDN line**.**

isdn set dns [*number*] [*number*]

If SPID detection fails, SPIDs must be manually configured, as described previously.

Note: Configuration Manager and Quick Start both have dialogs to monitor Auto-SPID detection.

Allow or Exclude Outgoing and/or Incoming Data Calls

You can decide whether to allow or lock out data calls. This feature is particularly useful if your router is configured to bridge and you want to ensure that no data calls are made while you are not present. Use the command:

isdn set DataCallsAllowed <*option*> [YES|NO]

Refer to the command reference section for more details on the syntax.

Save, Reboot, and Check ISDN settings

Save the current settings to FLASH memory with the command:

save

Then reboot the router with the command:

reboot

NEVER power down immediately after using a **save** command. First type the command **sync** to synchronize the file system with the disk cache; otherwise information may be lost.

The router will go through POST and reboot the router software. Note that whenever you reboot the router, you must log in again if you wish to change the router's configuration.

Note: If you do not save the configuration to FLASH, the configuration is lost upon reboot or power down of the router.

Connect ISDN line and check ISDN status

Now that the reboot has been completed, it is safe to connect the ISDN line to the router.

To list all the current ISDN settings and check status, enter the command:

isdn list

You should receive results similar to the following:

```
# isdn list
DSL 0 is Idle
Switch type is National ISDN-I
ISDN Outgoing data calls allowed: yes
ISDN Incoming data Calls allowed: yes
Retry failed calls every 30 seconds
      CES: 1: 0555100001/5551000 TEI 76 assigned
      CES: 2: 0555300001/5553000 TEI 77 assigned
  ISDN/2               Idle ces=0 cid=-1 not assigned
  ISDN/3               Idle ces=0 cid=-1 not assigned
```

Step 4. Set Target System Ethernet LAN and DHCP Addressing

You will now initialize the router’s Ethernet LAN IP address or the Ethernet LAN IPX network number if you use IPX routing. If you are configuring the router at the office headquarters and then installing the router at a branch office, use the Ethernet LAN addressing of the LAN at the branch office. If you intend to test the router at the host site first, enter the LAN address of the host site. If you change the addresses, you must perform a Save and Reboot as shown in later steps.

Initialize Ethernet IP address

Enter the command:

eth ip addr *x.x.x.x y.y.y.y*

where *x.x.x.x* is the IP address and *y.y.y.y* the subnet mask for the router’s Ethernet LAN connection. No checking is performed on the addresses.

The command **eth list** lists the settings for the Ethernet LAN IP address and subnet mask as well as the port number.

The command also lists routing and bridging status. Following is a sample of the results of this command:

```
# eth list
GLOBAL BRIDGING/ROUTING SETTINGS:
  Bridging enabled.................... no
  IP Routing enabled.................. no
    Multicast forwarding enabled....... no
    Firewall filter enabled........... yes
    RIP Multicast address............. default
  IPX Routing enabled................. yes
ETHERNET INFORMATION FOR <ETHERNET/0>
  Hardware MAC address................ 00:20:6F:02:C5:DC
    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
  IP address/subnet mask.............. 192.168.254.254/255.255.255.0
  Static Ethernet routes defined....... none
  IPX External network number......... 00000456
  IPX Frame type...................... 802.2
```

eth list is a useful command to verify that the router’s LAN IP address and subnet mask are set correctly. Note that firewall filtering, sending and receiving RIP packets, advertising the default route and receiving the default route are set on.

Note 1: The preceding response shows you that, at present, bridging is disabled and routing from the LAN is enabled. This is the initial status when you install the router

Note 2: If another router on the local LAN has been specified as the default router, you should disable the router from advertising itself as the default router. To do this, enter:

eth ip options avdfr off

Enable Ethernet IP Routing

At this point you can enable IP routing, save the configuration, and reboot the router to test the router's local IP connectivity and the ISDN line configuration.

eth ip enable

Save the information to FLASH memory:

save

Then reboot the router with **reboot**

Note: If you do not save the configuration to FLASH, the configuration is lost upon reboot or power down of the router.

You can verify IP connectivity by running the **ping** command (an IP echo facility) to the target router. Also verify that the ISDN line is in standby status. Enter the **isdn list** command or use the command **ifs** to list the status of all interfaces. You should receive a response similar to the following:

```
# ifs
Interface   Speed  In%  Out%  Protocol   State Connection
ETHERNET/0   10mb 0%/0% 0%/0% (Ethernet) OPENED
ISDN/3        0 b             (HDLC/PPP) STANDBY
ISDN/2        0 b             (HDLC/PPP) STANDBY
ISDN-D/0     16kb 0%/0% 0%/0% (HDLC/LAPD OPENED
CONSOLE/0  9600 b 0%/0% 0%/0% (TTY)      OPENED
```

DHCP Server

The router supports DHCP and acts as a DHCP router itself, allowing hosts (i.e. PCs) to acquire initialization parameters (IP addresses, masks, domain names etc.) automatically. DHCP is initially enabled by default. Remember that DHCP must also be enabled on your PC (refer to the *User Guide* for instructions). However, if you intend to primarily configure your router for Internet access, you need to enter the DNS information provided by your Internet Service Provider. To do so, use the two following commands:

dhcp set valueoption domainnameserver <*domainnameserver*>

dhcp set valueoption domainname <*domainname*>

Examples:

```
dhcp set value DOMAINNAMESERVER 172.16.100.100 172.16.200.1
dhcp set value DOMAINNAME myisp.com
```

Note: If you need more detailed information on DHCP, refer to page 36.

CAUTION: Caution should be used when connecting the router to a company LAN which contains other DHCP servers. The router's DHCP server will turn itself off when it detects another DHCP server on the network; however, to avoid possible confusion on a network, disable the router's built-in DHCP server by entering the following commands:

```
dhcp disable all

save
```

Initialize Ethernet IPX Address

If you intend for the router to perform IPX routing, you need to set the Ethernet IPX address. Enter the command:

eth ipx addr <*network#*>

where <*network#*> is the external Network Number for the LAN segment that the router is on. No checking is performed on the network number.

You may also need to set the frame type, which is the encapsulation method used to send multiple network-level protocols over the LAN or WAN link. The default frame type is 802.2. If you need to change the frame type, enter the command:

eth ipx frame <*type*>

where <*type*> is the encapsulation method (802.2, 802.3, or dix).

Verify that you have entered your parameters correctly with the following command **eth list**.
This command lists the settings for the Ethernet LAN address as well as other Ethernet LAN information including routing and bridging status.

Save the Ethernet LAN configuration to FLASH memory with **save.**

Verify that the ISDN line is in standby status. Enter the **isdn list** command or use the command **ifs** to list the status of all interfaces.

Step 5. Add Remote Routers into Remote Router Database

You must now enter all the remote routers to which this target (local) router may connect into the remote router database and specify details about ISDN lines, bandwidth management, security, bridging, and routing. You can add a new remote router to the database, modify router information that you have already entered or delete a router.

It is recommended that you enter information for one or two remote routers, verify your settings and then test access to these remote sites. Then add other routers to the remote router database.

Add a New Remote Router

To add an entry for a remote router into the remote router database, enter the following command:

remote add <*routerName*>

Once you add a router entry, you can enter all additional data about the remote router.

Modify/Delete/Enable/Disable a Remote Router Database Entry

The following commands are used to modify, delete, enable, or disable a remote entry:

remote del <*routerName*>

remote enable <*routerName*>

remote disable <*routerName*>

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.

Step 6. Configure Dial-Up Link Information

Set Remote Router Telephone Numbers

To set ISDN phone numbers for the remote router, enter:

remote setPhone isdn *index* <*phone#*> <*remoteName*>

Specify **1** or **2** for *index* indicating the first or second ISDN B-channel, respectively, and enter the corresponding ISDN phone number. (ISDN telephone numbers can contain the numbers 0-9 and the characters * and #.)

After you have entered the phone numbers, verify your settings with the following command:

remote listPhone <*remoteName*>

Following is an example of the results of this command:

```
# remote listPhone HQ
PHONE NUMBER(s) FOR <HQ>
1. ISDN telephone number, speed auto 5552000
2. ISDN telephone number, speed auto 5554000
```

Set Disconnect Timer Value

You can alter default settings of the disconnect timer value. The disconnect timer lets you minimize dial-up costs by forcing a disconnect of the ISDN line after periods of inactivity. The default disconnect timer value is 60 seconds. To change the value, enter the following command:

remote setTimer <*timerValue*> <*remoteName*>

where <*timerValue*> is a number (in seconds). The router will disconnect the ISDN link after the number of seconds has passed since the last data transmission.

Note: If you use zero (0), the link will be disconnected before it comes up.

Step 7. Configure Bandwidth Management

Bandwidth-on-demand lets you optimize the use of the two ISDN B-channels to accommodate variation in traffic flow. You will need to observe the data traffic flow from/to your site over time to determine the most effective settings for each of the control parameters.

Set Maximum Links

To enable bandwidth-on-demand management, you need to assign two ISDN B-channels to be available for use on one connection. The maximum links parameter lets you specify the maximum number of ISDN B-channels used on the same connection. Enter the command:

remote setMaxLine <*maxLine#*> <*remoteName*>

where <*maxLine#*> is **1** or **2**. The default for an ISDN link is to have one B-channel available for use. If you specify **2**, the router can utilize up to two channels for data traffic on one connection.

Set Minimum Links

The default is to assign a B-channel *only* when data traffic occurs (minimum links = 0). You may, though, choose to have a number of B-channels (up to the maximum links) permanently reserved for a remote router connection. If you wish to reserve B-channels, enter the command:

remote setMinLine 0|1|2 <*remoteName*>

Caution: Using this command will result in a very phone bill

Set Bandwidth Threshold

If you have specified a maximum of two B-channels, you can now also set the bandwidth threshold and direction control. The bandwidth threshold determines when a second B-channel is assigned to accommodate increased traffic (or released on decreased traffic). Set the threshold using the following command:

remote setBWThresh <*threshold*> <*remoteName*>

where <*threshold*> is a percentage from 0 to 100**.** The default is 0%, meaning that the second B-channel (up to the maximum links) will be used immediately.

Set Direction Control

Bandwidth-on-Demand can apply to inbound, outbound, or both inbound and outbound traffic. Specify the direction of traffic to be managed by issuing the command:

remote setBod IN|OUT|BOTH <*remoteName*>

The default is to have bandwidth-on-Demand on both inbound and outbound traffic.

Step 8. Set Up Security

You must specify the remote router's authentication protocol and password used by the target router when communicating with the remote router.

Set Remote Router's Authentication Protocol

The authentication protocol is the minimum security level that the target router must use with the remote router and this level is checked 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.

Remember that authentication is a bi-directional process, where each end can authenticate the other using the protocol of its choice (provided the other end supports it.) The parameter in the remote router database is used for the local side of the authentication process. It is the minimum security level used by the target router when challenging or authenticating the remote router.

To set the remote router's authentication protocol, enter the following command:

remote setAuthen <*protocol*> <*remoteName*>

where *protocol* is **PAP, CHAP** or **NONE**. The default is **PAP**.

Set Remote Router's Authentication Password

The remote router's authentication password is used for validation when the remote router dials in or is challenged by the target router. The default is no password. Enter or change the remote router's password with the following command:

remote setPasswd <*password*> <*remoteName*>

Set Unique System Name and Password for Local Router

The local router uses its system name and system password as its identity when connecting to all remote routers. If the local router needs to present a different identity to each remote that it connects to, then use these commands. They will override the system name and password when connecting to the specified remote. This is useful, for example, when an Internet Service Provider assigns you a password or you want to set a password different from the system password when calling a remote location. (The default system name and password are used for authentication of the local router by the remote site in all other cases.) Refer to the section *Unique System Passwords* in the previous chapter.

To set a unique system name and password, enter:

remote setOurSysName <*systemName*> <*remoteName*>

remote setOurPasswd <*password*> <*remoteName*>

Step 9. Set Up TCP/IPRouting

TCP/IP Routing is established by entering all remote routers in the remote router database to which this router will connect. For each remote router, you enter addresses for the networks and stations that may be accessed beyond the remote router. You may set a local and/or remote WAN IP address for the WAN link. You will define a default route and set IP (RIP) protocol options.   After specifying the route addressing, you then enable IP routing across the Ethernet LAN. Be sure to review the section *Important Routing Concepts* in chapter 4 of the *User Guide*. If you do not plan to configure TCP/IP Routing, go to step 10.

Add TCP/IP Route Addresses

When IP traffic is for networks and stations beyond a remote router, the target router's routing table can be statically seeded. Static seeding ensures that the target router dials out to the appropriate remote router. After the link is established, RIP update packets will dynamically add to the target router's routing table. Seeding the routing table is not necessary when a target router never dials out; it will discover remote networks and stations beyond the calling router as soon as RIP updates arrive (provided the remote router supports RIP *and* RIP packets are allowed to flow on the WAN link).

One default route (0.0.0.0 with a mask of 255.255.255.255) is designated in the routing table for all traffic that cannot be directed to other specific routes.   Enter a distinct default route for a remote router if the target router will be placing calls to that remote router.

To seed the routing table with TCP/IP route addresses of stations or networks connected beyond a remote router, enter for each TCP/IP route address:

remote addIpRoute <*ipnet*> <*ipnetmask*> <*hops*> <*remoteName*>

ipnet is the IP address of the network/station,

ipnetmask is the network mask,

hops is a number between 1 and 15 that represents the perceived cost in reaching the remote network or station.

You can list the routes that you have added by entering:

remote listIpRoute <*remoteName*>

A sample response from this command is:

```
remote listiproute isp
IP INFORMATION FOR <isp>
  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
```

The IP route shown is the default route. Other examples of IP routes are listed in the command reference section.

Set Local and Remote WAN IP Addresses

You can specify a Remote WAN IP address and/or a Target WAN IP address for the WAN connection to the remote router depending on IP address negotiation under PPP.

Unnumbered mode: If the remote router supports unnumbered mode, the Remote and Target WAN IP addresses do not need to be specified.

Numbered mode: For numbered mode, consider the capabilities of the remote router as well as your requirements. Specify a Remote WAN IP Address if the remote router does not support IP address negotiation under PPP (i.e., does not have a pre-assigned IP address). Specify a Target WAN IP Address if the target router must be on the same subnetwork as the remote router.

Specify the Target (Src) or Remote (Rmt) WAN IP Address and Subnet Masks for the remote router using the following commands:

remote setSrcIPAddr <*ipaddr*> <*ipnetmask*> <*remoteName*>

remote setRmtIpAddr <*ipaddr*> <*ipnetmask*> <*remoteName*>

List Database Remote Router Entries

After you have entered the remote router, check the information in the remote router database with the following command:

remote list <*routerName*>

A sample response from this command is:

```
# remote list
INFORMATION FOR <HQ>
  Status............................... enabled
  Our Password used when dialing out... no
  Disconnect timeout (in seconds)...... 60
  Min/max channels..................... 0/2
  Interface in use..................... ISDN
  Authentication....................... disabled
  Authentication level required........ CHAP
  Bandwidth management criteria........ both
  Utilization threshhold............... 50%
  1. ISDN telephone number, speed auto 5552000
  2. ISDN telephone number, speed auto 5554000
  Dial Back.............................off
  Request PPP Call Back.................no
  Place ISDN Data Call as Voice Call....no
  IP address translation............... off
  Send/Receive Multicast............... off
  Compression negotiation.............. on
  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 0004 1
  Bridging enabled..................... no
    Exchange spanning tree with dest... no
    Mac addresses that dial remote..... none
INFORMATION FOR <ISP>
  Status............................... enabled
  Our Password used when dialing out... no
  Disconnect timeout (in seconds)...... 60
  Min/max channels..................... 0/2
  Interface in use..................... ISDN
  Authentication....................... disabled
  Authentication level required........ PAP
  Bandwidth management criteria........ both
  Utilization threshhold............... 50%
  1. ISDN telephone number, speed auto 18005551115
  2. ISDN telephone number, speed auto 18005551116
  Dial Back.............................off
  Request PPP Call Back.................no
  Place ISDN Data Call as Voice Call....no
  IP address translation............... on
  Send/Receive Multicast............... off
  Compression negotiation.............. on
  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... no
    Mac addresses that dial remote..... none
```

To list all remote routers, enter **remote list** without a specific router name.

Save and Test IP Routing Configuration

After you have verified that the remote router information is correct for each remote router, you can save the information to FLASH memory with the following command:

save

At this point, you can reboot the router and test the routing configuration.

Warning: If you do not save the configuration to FLASH, the configuration is lost upon reboot or power down of the router.

To test the routing configuration and the WAN link to the remote router, use the following command:

call <*remoteName*>

You can check the status of the ISDN link and channel activity by entering the **isdn list** command:
Issue the **ifs** command to verify that the line is opened or in another appropriate state.

```
# isdn list
DSL 0 is Idle
Switch type is National ISDN-I
ISDN Outgoing data calls allowed: yes
ISDN Incoming data Calls allowed: yes
Retry failed calls every 30 seconds
      CES: 1: 0555100001/5551000 TEI 76 assigned
      CES: 2: 0555300001/5553000 TEI 77 assigned
  ISDN/2               Idle ces=0 cid=-1 not assigned
  ISDN/3               Idle ces=0 cid=-1 not assigned
```

To verify IP routes, **ping** each remote station. When you enter the **ping** command from a station on the local Ethernet LAN, the router will dial out to the remote router using the ISDN link.

If the **ping** is unsuccessful, verify the TCP/IP route addresses, ISDN line information, security protocols and passwords, routing status and cables.

Issue the **ifs** command to verify that the line is opened or in another appropriate state.

You may also want to test access to both B-channels of the ISDN link. Initiate a communication (file transfer, for example) to two different remote routers concurrently.

Step 10. Set Up 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. For each remote router, you enter network addresses and services that may be accessed beyond the remote router. You also enter a network number for the WAN link. After specifying the route addressing and services, you then enable IPX routing across the Ethernet LAN. If you do not wish to configure IPX Routing, go to *Step 11. Set Up Bridging.*

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 dials out 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 dials out; 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.

Steps to Configure IPX Routing

Configuring your router for IPX routing can be rather difficult. The following table will guide you through the configuration process for IPX routing. Remember that the ISDN, dialing, and PPP Authentication settings must be configured *before* attempting the IPX routing configuration. The full router configuration for simple IPX routing includes the following elements:

- Local ISDN
- Dialing
- PPP Authentication
- IPX routing (this section)

Configuration Sample

These commands are used to configure the Target (client-side) router. Log in with the password **admin.**

Note: The Remote (server side) router (called "ipx_server" in our example) must be configured with an IPX route to the Target (client side) router's external network #.

IPX Commands with examples	Comments
eth ipx enable	Enable IPX Routing
eth ipx addr 123	Set the local 'wire' address
eth ipx frame 802.2	Set the Frame Type
remote add ipx_server	Add a connection name
remote setipxaddr 456 ipx_server	Set the WAN network # (common to both sides)
remote addipxsap SERVER2 2002 00:00:00:00:00:01 0451 4 1 ipx_server	Add a file server (SAP)
remote addipxroute 2002 1 4 ipx_server	Add a route to the server
save	Save your settings
reboot	Reboot for changes to take effect

Step 11. Set Up Bridging

Bridging is established by entering all remote routers in the remote router database to which this router will bridge traffic. The target router can bridge traffic to/from each remote router. Bridging initially defaults to 'off'. If you wish the router to bridge traffic to/from a remote router, you must enable bridging on.

You must also specify one remote router as the default bridging destination for outbound bridging if the target router is to dial out. All packets, with an unknown destination, are bridged to this default bridging destination if IP and IPX routing are disabled. If IP Routing and/or IPX routing is enabled, bridging to this destination occurs only on packets that are not routed.

Set Default Bridging Destination

Specify a default bridging destination with the following command:

remote addBridge * <*remoteName*>

The ***** indicates that all addresses on the LAN are bridged to the remote router. (This command does not enable bridging.) As a learning bridge, additional remote destinations will be added to the bridging table. You may also seed the bridging table in the router with remote destinations for individual MAC addresses by specifying a **MAC address** instead of ***** in the command.

Use Spanning Tree Protocol

This feature allows the router to check for bridging loops and communicate with other sites that support the protocol. If a remote site does support STP, you can turn the protocol on with the command:

remote setbroptions stp [ON|OFF] <*remoteName*>

The STP defaults to **off** when bridging over the ISDN WAN. This eliminates a period of about 40 seconds during which the ISDN lines are dialed and no user traffic is forwarded, while the Spanning Tree Protocol checks for and eliminates loops in the network topology. If you choose to leave STP off, this assumes that no pair of nodes on the larger network, made by joining all the local LANs that can dial each other, can be connected by more than one path. This assumption usually holds true for telecommuters and many branch office situations. If there is a possibility of redundant paths between nodes, the Spanning Tree Protocol should be turned on when dialing a site where such a loop possibility exists.

Enable Bridging

After you have set the bridging capability, enable bridging with the following command:

remote enaBridge <*remoteName*>

Check your bridging configuration with the command:

remote listBridge <*remoteName*>

Following is a sample of the results of this command:

```
# remote listbridge HQ
BRIDGING INFORMATION FOR <HQ>
Bridging enabled................... yes
Exchange spanning tree with dest... no
Mac addresses bridged.............. all
```

Save and Test the Bridging Configuration

After you have verified that the remote router bridging information is correct, save the bridging information, reboot the router, and test the bridging configuration. Remember only one remote router can be configured as the default outbound bridging destination.

Save the remote router configuration to FLASH memory with **save.**

Reboot the router to activate the bridging configuration with **reboot.**

Warning: If you do not save the configuration to FLASH, the configuration is lost upon reboot or power down of the router.

You can test your configuration and the WAN link to the remote router, by using the following command:

call <*routerName*>

You can test the bridging configuration by using any application on a locally LAN-attached system that accesses a server or disk on the remote network beyond the remote router. When you access the remote network/station, the router will dial out to the remote router using the ISDN link.

If the access is unsuccessful, verify:

- Default bridging destination
- ISDN line information
- Security protocols and passwords
- Bridging status
- Bridging has been enabled at the host router

If you have configured the router for both routing and bridging (and have not tested this configuration), test concurrent routing and bridging. Enable both routing and bridging as described in previous steps. Be sure to specifically use the remote destination for bridging by accessing a network/station beyond that remote router. Access a remote network/station using **ping** to test IP Routing.

Step 12. Save the Configuration and Reboot

When you have completed all modifications to the router's configuration, you can save the entire configuration to FLASH memory using the **save** command. (If you have performed a save during each step of the configuration process, this step is unnecessary.) Any settings that you have modified will be permanently stored in the router's configuration. Any settings you have not modified will be unchanged or default if this is your first configuration.

Warning: If you do not save the configuration to FLASH, the configuration is lost upon reboot or power down of the router.

After storing the configuration, enter the **reboot** command**.**

Step 13. Verify the Router Configuration

After saving the entire configuration and rebooting, test the ISDN line configuration, the POTS configuration, IP and/or IPX routing and bridging. Repeat any tests that you have performed in earlier steps.

Test the ISDN Line

You can test the ISDN line and configuration settings with the **call** <*remoteName*> command. The call command causes the target router to dial and connect to the remote router specified in the command.

If you cannot connect to the remote router, issue the **ifs** command to check ISDN channel status.

If ISDN channels are not in "standby" or "opened" mode, verify:

- SPIDs, DNs, and switch type
- The telephone company provisioning
- The associated equipment (NT1, etc.)

- Cables

If the ISDN line is operational, check the remote router’s telephone numbers and link parameters.

Be sure the security authentication method and password that you configured matches the remote router.

Error messages will notify you if you have a security configuration error or SPIDs/DNs negotiation error. Refer to *Chapter 5. Troubleshooting Software Problems* for more details.

Test IP Routing

♦ Test IP Routing over the Local Ethernet LAN

- Use the TCP/IP **ping** command or 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
- Check the cable connections and pinning.
- Be sure that you have saved and rebooted after setting the IP address.
- Also, check Network TCP/IP properties under Windows 95 or that you have a TCP/IP driver installed under Windows 3.1.

♦ 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 dial out to the remote router using the ISDN line.
- Verify that you configured valid remote and local (source) WAN IP addresses, if required.
- Use the **iproutes** command to check the contents of the IP routing table and that you have specified a default route as well.

♦ Test TCP/IP Routes

- Contact a station, subnetwork or host 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 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 go to the login drive (usually F:) and type **login.** Select the printer server and verify that the server you have defined is listed. When you attempt to access the server, the router will dial out to the remote router using the ISDN 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 pinning.
- 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.
- Be sure the security authentication method and password that you configured matches the remote router.

Error messages will notify you if you have a security configuration error or ISDN SPIDs/DNs negotiation error. Refer to *Chapter 5. Troubleshooting Software Problems* for more details.

Test Analog Services (for POTS routers only)

- It is recommended that you first test voice calls when no data calls are active.
- Be sure to check that you have a valid ISDN configuration, including SPIDs (if required) and central office switch setup.
- Verify that the ISDN link is operational with the **isdn list** command.

Test Outbound Voice Calls

You can test the POTS configuration by dialing any remote phone number from the attached analog telephone. The default configuration is dial mode and call preemption on both POTS interfaces.

With an operational ISDN link, you should get a dial tone when picking up the handset. If you do not get a dial tone, use the **pots list** command to verify that dial mode is configured and that you have enabled the POTS interface.

You may not get a dial-tone if a channel is temporarily unavailable (when the router cannot preempt or the other POTS line is currently dialing).

Test Inbound Voice Calls

You can test inbound voice calls to the POTS interfaces by dialing the associated phone numbers from another phone. The default configuration is answer mode and call preemption on both POTS interfaces.

If you do not get through, check that "answer" (or "both") mode is configured for the specified POTS interface and that the POTS interface is enabled. If both channels are in use, check for call preemption on inbound calls.

Enter the command **pots list** to verify the current POTS configuration.

If both data channels are active, you must also have subscribed to ‘Additional Call Offering’ through the phone company.

If you try to use one phone to call the other phone, make sure that your ISDN service is configured for two voice calls.

Step 14. Logout

After all configuration changes have been made and saved, the router has been rebooted and testing is complete, enter the command:

```
logout
```

This command reinstates administrative security on the router. Note that after a **reboot**, you are required to log in again if you wish to make any modifications to the configuration.

Sample Configuration

Scenario

In this configuration example of a hypothetical network, a small office **(SOHO)** will access a central site **(HQ)** via an ISDN link. The small office also has access to Internet through an Internet Service Provider **(ISP).** SOHO has IP routing enabled to ISP with a Class C addressing scheme, and IPX enabled to HQ.

Bandwidth-on-Demand is configured for accessing central site HQ.

Two lines are configured for calling the ISP (though two different phone numbers are defined for use).

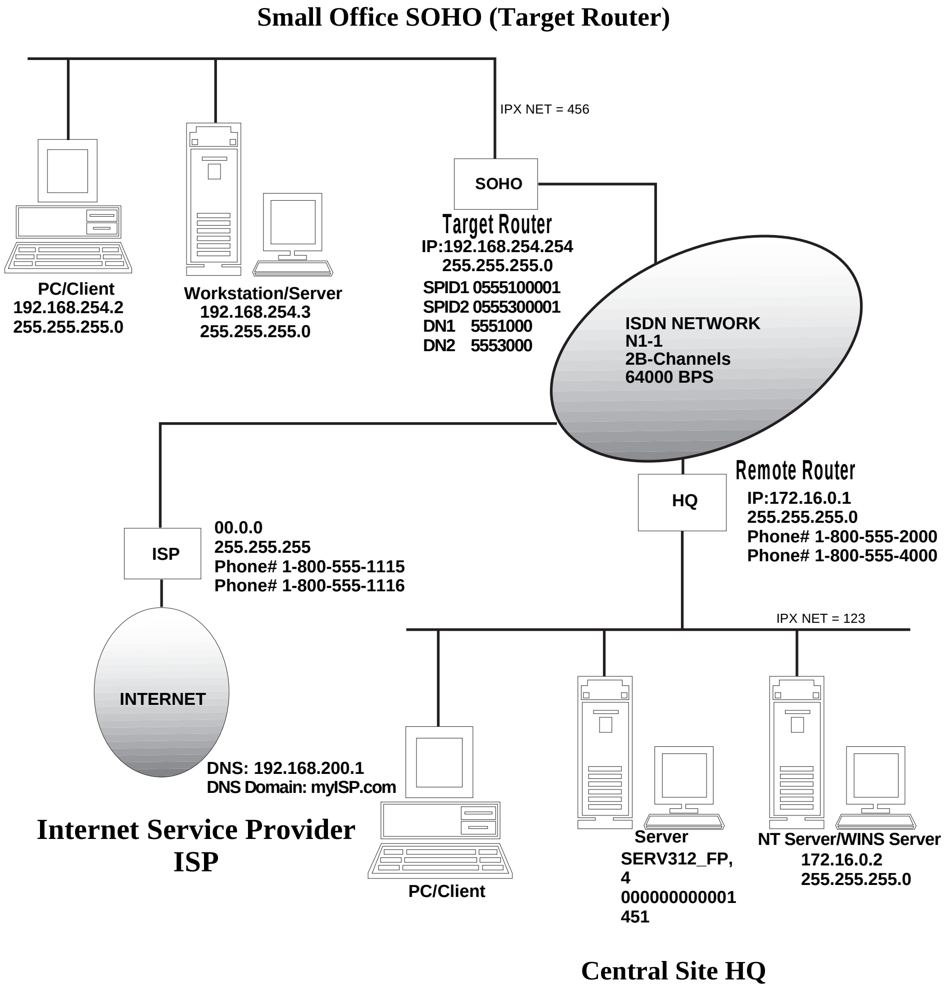
DHCP server's IP addresses are used. DHCP is set up to issue DNS information to the SOHO LAN.

Network Address Translation (NAT) is enabled to the ISP, since the ISP assigned SOHO only one IP address. HQ assumes that SOHO is on the 192.168.254.0 subnet.

The following diagram and Network Information Worksheets show configuration of router SOHO at the small office.

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

Network Diagram



Network Information Worksheets

First, the two following commands are used to log in and reset the password (optional):

```
login admin
system admin newpass
```

<table>
<tr><th colspan="2">TARGET ROUTER: SOHO</th></tr>
<tr><th>Configuration Section</th><th>Configuration Commands</th></tr>
<tr><td><b>System Settings</b>
Router Name</td><td>system name soho</td></tr>
<tr><td><b>System Settings</b>
Message</td><td>system msg configured_july98</td></tr>
<tr><td><b>System Settings</b>
Dial Authentication Password</td><td>system passwd SOHOpasswd</td></tr>
<tr><td><b>System Settings</b>
IP address and subnet mask</td><td>eth ip addr 192.168.254.254 255.255.255.0</td></tr>
<tr><td><b>System Settings</b>
DHCP Settings
DNS Domain Name
DNS Server
WINS Server address</td><td>Use defaults and add the following:
dhcp set valueoption domainname myISP.com
dhcp set valueoption domainnameserver 192.168.200.1
dhcp set valueoption winsserver 172.16.0.2</td></tr>
<tr><td><b>ISDN Settings</b>
ISDN Switch Type
ISDN SPID#1, SPID#2
ISDN Directory Number #1, #2</td><td>isdn set switch NI-1
isdn set spids 0555100001 0555300001
isdn set dns 5551000 5553000</td></tr>
<tr><td><b>Ethernet IPX Address</b></td><td>eth ipx addr 456</td></tr>
</table>

REMOTE ROUTER: HQ	
Configuration Section	**Configuration Commands**
Remote Routers Dial Settings ISDN Phone #1 (11 digits) ISDN Phone #2 (11 digits) Maximum Links	remote add hq remote setPhone 1 isdn 18005552000 hq remote setPhone 2 isdn 18005554000 hq remote setMaxLine 2 hq
Remote Routers Security Minimum Authentication Remote Router's Password	remote setAuthen CHAP hq remote setPasswd HQpasswd hq
Remote Routers Bridging Bridging On/Off: Default is off.	
Remote Routers TCP/IP Route Addresses Remote Network's IP Addresses, Subnet Masks, and Metrics	remote addIpRoute 172.16.0.0 255.255.255.0 1 HQ
IPX Route Addressing IPX routes IPX SAPS Remote WAN IPX Address	remote addipxRoute 1001 1 4 HQ remote addIpxSap SERV312_FP 1001 00:00:00:00:01 451 4 1 HQ remote setIpxAddr 789 HQ

REMOTE ROUTER: ISP	
Configuration Section	**Configuration Commands**
Remote Routers Dial Settings ISDN Phone #1 (11 digits) ISDN Phone #2 (11 digits) Maximum Links	remote add isp remote setPhone isdn 1 18005551115 isp remote setPhone isdn 2 18005551116 isp remote setMaxLine 2 ISP
Remote Routers Security Disable Authentication Remote Router's Password	remote disAuthen isp remote setPasswd isp passwdsip
Remote Routers Bridging: Default is off.	
Remote Routers TCP/IP Route Addresses Remote Network's IP Addresses, Subnet Masks, and Metrics Address Translation	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

Bridging and Routing Controls	
Configuration Section	**Configuration Commands**
Bridging / Routing TCP/IP Routing On/Off IPX Routing On/Off Internet Firewall On/Off	eth ip enable eth ipx enable eth ip firewall on

When the router's configuration has been completed, the entire configuration is saved with:

```
save
reboot
```

The following commands are then used to check your configuration:

```
ifs
isdn list
iproutes
ipxroutes
remote list
```

Names and Passwords Example

In the sample configuration provided, the small office SOHO communicates with the central site HQ and the Internet Service Provider ISP.

System Passwords

SOHO has a system password ‘SOHOpasswd’. This password is used when SOHO dials out to site 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 dials out to 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 site when the remote router dials in or is challenged by the local site. For example, SOHO has remote router entries for HQ and ISP, and defined in each 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

Chapter 3. 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.

- Bridging Filtering and IP Firewall
- ISDN Subaddressing
- CallerID Security
- Call Management
- Analog Settings
- 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 [<u>*pos*</u>] is the byte offset within a packet (number from 0-127) to a [<u>*data*</u>] (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. <u>Remember to save and reboot</u> if you alter IP routing status.

ISDN Subaddressing

ISDN subaddressing enables ISDN devices connected on an S/T interface to be addressed uniquely by an address or identifier. Subaddressing information is passed between ISDN peers during call set-up of ISDN connections and is used to target communications to a specific ISDN device (similar to a multi-point leased line capability).

Subaddressing allows you to have one telephone number for the ISDN equipment and provides an alternative to having a unique telephone number for each ISDN device. Subaddressing can be used whether one or more devices are connected to an S/T interface.

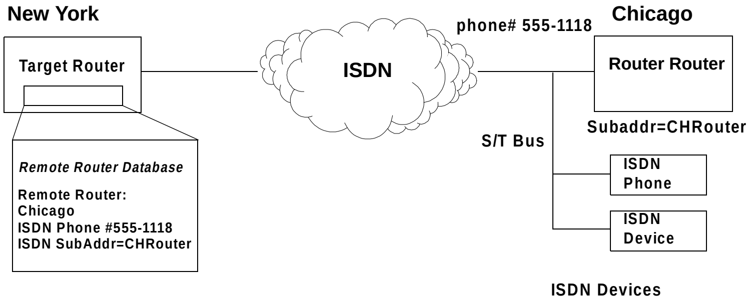
ISDN Subaddressing Configuration Settings

ISDN subaddressing configuration involves setting a subaddress for the local router and/or subaddresses for the remote routers. The subaddresses can be user-defined or network service access points (NSAPs), a format defined by the international standard Q.931.

Each device on an S/T interface "sees" the subaddress with the incoming transmission, but only the addressed device processes the packet. If a subaddress has been defined for the router, only transmissions that have a correct subaddress will be accepted, and a subaddress must be sent. If subaddressing is not defined and a transmission is received with a subaddress, the call is ignored.

Note that the router will *never* clear a call if subaddressing is incorrect; the call will be ignored.

ISDN Subaddressing



Note: The above example shows ISDN subaddressing for the remote router. If multiple devices were connected on the local (target) side with an S/T bus, then a subaddress must be defined for the target router.

Set Subaddressing

You can assign a subaddress to the router that lets remote routers uniquely identify this ISDN device during call set-up. Use the command:

isdn set subaddr u|n <*string*>
where **u** refers to a user defined subaddress and **n** refers to the international standard NSAP address. Refer to the command reference section for more details on the syntax.

Set Remote Router Subaddress

You can specify a remote subaddress to allow the router to uniquely identify the remote ISDN device. This should be used *only* if the remote device supports subaddressing. Use the command:

remote setsubaddr u|n *<subaddr> <remoteName>*
The format of the subaddress is described in the command reference section.

Caller ID Security

General Information

CallerID is an additional security feature on data calls supported by the router. CallerID allows you to verify phone numbers of the remote routers when calls come in to the local router. This feature is system-wide and you must configure the phone numbers from which a remote router can call. Any calls from other numbers will be rejected.

The allowable phone numbers must be obtained from the remote locations or your network administrator.

Define CallerID Phone Numbers and Enable

You configure the phone numbers from which a specific remote router can call and enable or disable this feature system-wide.   Any calls from other numbers will be rejected. To specify the unique numbers for the remote router, use the command:

remote addCaller <isdn *phone#> <remoteName>*
Note: the configured phone numbers must contain the actual digits passed through the switch.

Then enable/disable the CallerID feature with the command:

system callerid isdn on|off
Save the system settings and remote router database configuration. Then test CallerID. If the call is rejected by the local router, check the message displayed on the console for the actual digits received and reconfigure with the correct number. Display the status of CallerID using the **system list** command, and the **remote list** command for the remote router.

Call Management

The router supports call management features that allow you to control ISDN line usage charges. Dial-Back and PPP CallBack control whether the local or remote router are charged for the call. The “data as voice” feature allows data calls to be billed as voice calls (U.S. only) which may reduce line charges.

Configure Dial-Back

Dial-Back lets you force the router to reject an incoming call from another router and dial that router back. You can use this feature to cause ISDN phone charge billing to the local router. Dial-Back can be enabled, disabled or enabled such that Dial-Backs occur only if called by the remote router first. To add Dial-Back to a remote router, use the command:

remote setdialback on|off|only *<remoteName>*

When Dial-Back is configured, the local router's call delay timer setting must allow for disconnect and dial back; the defaults (30 seconds for the U.S. and 90 seconds for Europe, Japan) or longer should be acceptable. If you need to alter the timer setting, use the command **isdn set call delay**.

The local router searches the remote entries to find a match specifying the calling router's telephone number that should be dialed back. The calling router's telephone number is associated with a remote entry using the **remote addCaller** command.

Caution must be used when entering the calling router's telephone number since the number presented to the router may look different from what the user at the calling router's user actually sees on the telephone. For example, the ISDN network may add an area code or other prefixes. An easy way to determine the correct number is to enable callerID verification using the **system callerID** command without entering a **remote addcaller** number. The router will show that the calling telephone number xxx was rejected, since it was not found in the routers database. This is the number to use in the **remote addCaller** command.

Remember to disable the system callerID feature, if this is not desired. Note that the router will use the telephone number specified in the **remote addPhone** command to call back.

The following steps are used to set dial-back.

1/ To turn on dialback, use:

remote setdialback on|off|only <*remoteName*>

2/ To add phone numbers identifying the calling remote, use:

remote addCaller isdn <*phone#*> <*remoteName*>

Configure PPP CallBack

PPP CallBack causes the local router to request that a remote router disconnect and call the local router back. This feature results in ISDN phone charge billing to the remote router if the remote router accepts. You must specify any information obtained from the network administrator that is required by the remote end. To set PPP CallBack (with PPP user authentication), enter the command:

remote setPPPCallBack <*remoteName*>

If necessary, you can specify that a phone number, a phone number in E164 format, or a name is sent to the remote router. Refer to the command reference for a complete description of the syntax.

When CallBack is configured, the remote router's call delay timer setting must allow for disconnect and call back.

Configure Data as Voice

The "Data as Voice" feature causes data calls to be sent as voice calls over the ISDN service in the U.S. and may result in reduced line charges. You can configure a system-wide feature that allows you to receive data calls as voice calls. If you use this feature, all incoming voice calls will then be processed as data; i.e., you will not be able to use the POTS interface for incoming voice calls. Use the following command:

system dataAsVoice on|off

You can also cause data calls to a remote router to be sent as voice calls. Use the following command:

remote setDataAsVoice ON|OFF *<remoteName>*

Save the system settings and remote router database configuration. Then test call management. Display configuration status using the **system list** command, and the **remote list** command for the remote router.

Warning: This feature must be used with care. Both ends of the connection must agree to configure calls in this manner and the feature may not work depending on the central office service.

Analog Settings

The router's analog services allow for attaching analog telephones, fax machines and/or modem equipment to the POTS interfaces. This support lets you specify how phone numbers are associated with the POTS interfaces, whether the POTS interfaces can be used for dialing as well as for answering and whether voice calls have priority over data calls.

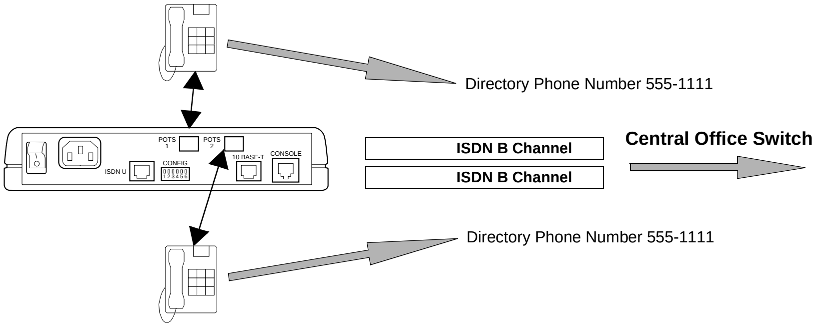
General Information

POTS Interfaces and Telephone Numbers

Your ISDN service provider has given you one or more telephone numbers that other locations or persons can dial to access the router. When you have attached analog devices, you need to associate these telephone numbers with the POTS interfaces so that an incoming voice call can be assigned to the correct analog port.

If you have a North American central office switch and have configured two SPIDS/DNs, the default configuration is DN1 is associated with POTS interface 1 and DN2 is associated with POTS interface Otherwise, the default configuration is an incoming call will ring on all available devices attached to the POTS interfaces. An outgoing call will use any available B-channel.

You may wish to assign telephone numbers to distinct analog devices. You can configure these numbers into the target router's system settings and then associate a unique telephone number with each POTS interface. You also have the option of assigning a telephone number to both POTS interfaces.



Analog Service Mode

You can designate a POTS interface to answer incoming calls and /or for dialing out. The default configuration sets both answer and dial mode for the two POTS interfaces.

Call Preemption

Call preemption allows you to give voice calls priority over data calls. Call preemption means a voice call (depending on the configuration options) will cause a disconnect of a data call on an ISDN B-channel. The default configuration is for both incoming and outgoing voice calls to preempt data, unless two data channels are in use to the same destination. A ‘no preemption’ configuration ensures that a data You can specify that incoming and/or outgoing voice calls preempt data calls or that no preemption occurs connection is maintained on at least one channel.

In all cases, a voice call will preempt one data channel if two channels are in use to the same destination. If preemption is designated for outbound calls and an outbound voice call is initiated while two data channels are in use to different destinations, the router will randomly select a B-channel to disconnect the data call. If preemption is designated for inbound calls and an inbound voice call comes in while two data channels are in use to different destinations, the router will also randomly select the line to preempt.

Call preemption does not occur on incoming calls unless a person picks up the phone or the analog equipment answers the call.

An incoming voice call may not always be forwarded from the central office if two B-channels are already in use for data calls. You must subscribe to a service called ‘Additional Call Offering’ for the voice call to be forwarded to the router.

Configure Analog Settings

Default Configuration

- Both POTS interfaces configured for both answer and dial mode.
- Voice calls will automatically preempt data calls and the POTS interfaces are enabled.
- If you have a North American central office switch and have configured two SPIDS/DNs, DN1 is associated with POTS interface 1 and DN2 is associated with POTS interface 2. Otherwise, the default configuration is for an incoming call to ring on all available devices attached to the POTS interfaces.
- An outgoing call will use any available B-channel.

Associate Phone Numbers with POTS Interfaces

If you wish to associate specific phone numbers (that have been assigned to you by the ISDN service provider) with a POTS interface, use the following command:

pots add <*pots#*> < *phone#*>

where <*pots#*> is either **1** or **2** and <*phone#*> is the associated phone number (or the least significant digits of the phone number). When you receive an incoming call for the specific phone number, the call will go to the matched POTS interface. An outgoing calls will use any available phone line.

If you want a phone number to be associated with both POTS interfaces, specify **all** instead of <*pots#*>.
If you wish to delete or disassociate a phone number with a POTS interface, use the **pots del** command.

Set POTS Interface Mode

You can set a POTS interface so that it can only answer a phone call, only dial a phone number, or be used for both answer and dial. Set the mode with the following command:

pots set line <*pots#*> **answer|dial|both**

If you want to set the same mode for both POTS interfaces, specify **all** instead of <*pots#*>.

Set POTS Interface Call Preemption

You can set a POTS interface to support call preemption in order to give a voice call priority over a data call. Set the mode with the following command:

pots set preempt <*pots#*> **in|out|both|none**

In, out, both: Call preemption will occur on data traffic when a voice call occurs inbound, outbound or both directions, if you specify **in**, **out**, or **both**, respectively.

None: If you specify **none**, call preemption will occur only when two data channels are in use to the same destination.

All: If you want call preemption to apply to both POTS interface, specify **all** instead of <*pots#*>.

The POTS interfaces default to the enabled state.   If you wish to disable or re-enable the interfaces, use the **pots ena(dis)able** command.

After you have specified the answer/dial mode, call preemption and associated phone numbers, list the POTS configuration with the following command:

pots list

```
# pots list
  pots(1).................ENABLED      state...............AVAILABLE FOR USE
          answer/dial mode....both      preempt.............incoming/outgoing
          if preempt, auto....incoming/outgoing
          last call attempt...outgoing
          last incoming call unknown
          last outgoing call unknown
          last local phone number used unknown
  pots(2).................ENABLED      state...............AVAILABLE FOR USE
          answer/dial mode....both      preempt.............incoming/outgoing
          if preempt, auto....incoming/outgoing
          last call attempt...outgoing
          last incoming call unknown
          last outgoing call unknown
          last local phone number used unknown
```

Save and Test POTS configuration

After you have verified that the POTS information is correct, save the POTS configuration with the command:

save pots

Warning: If you do not save the configuration to FLASH, the configuration is lost upon reboot or power down of the router.

You can now test the POTS configuration or continue on to the next step. To test the POTS configuration, use the attached analog phone to dial out to a remote phone number and call attached analog devices from another phone.

Refer to *Step 13. Verify the Router Configuration*. The ifs command shows the status of the ISDN channel used for the voice call.

```
# ifs
Interface  Speed  In%  Out%    Protocol    State     Connection
ETHERNET/0 10mb  0%/0% 0%/0%   (Ethernet)  OPENED
ISDN/3     0 b              (VOICE)     CONNECTED pots(1) call to #5553333
ISDN/2     64kb  83%/83% 3%/3% (HDLC/PPP)  OPENED    HQ
ISDN-D/0   16kb  0%/0% 0%/0%   (HDLC/LAPD) OPENED
CONSOLE/0 9600b  0%/0% 0%/0%   (TTY)       OPENED
```

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*>]] <*remoteName*>

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 76.

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 83)
- IP filters (For more information on this feature, refer to *IP Filtering*, page 86)
- L2TP Tunneling (For more information on this feature, refer to *L2TP Tunneling - Virtual Dial-Up*, page 89)

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

The routers SOHO (the target router) and HQ (the remote router) are configured in the same manner as shown in the section *Sample Configuration*, Chapter 2 of this manual, page 51, 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.

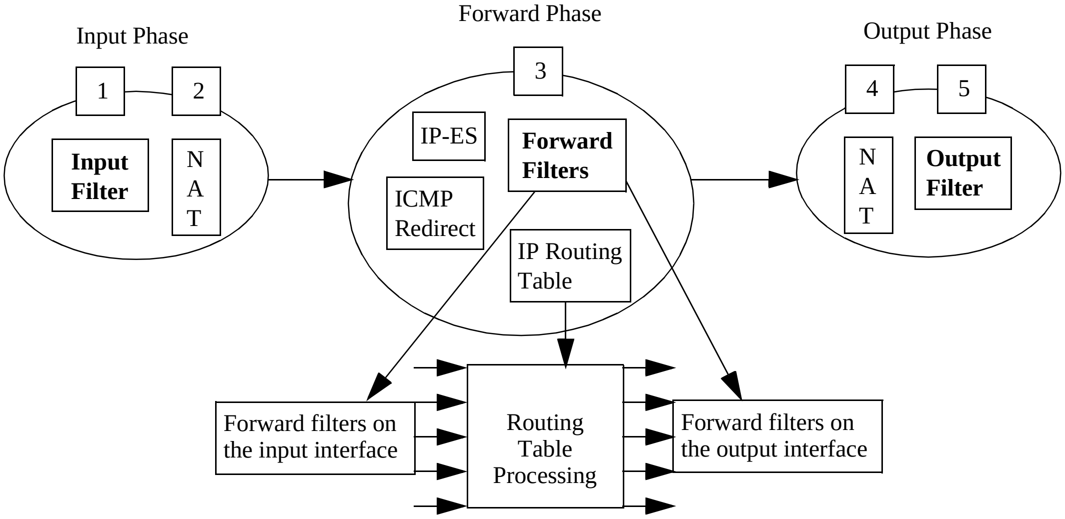
When filtering packets, the router examines information such as the source and destination address contained in the IP packet, the type of connection, etc., and then screens (filters) the packets based on this information: packets are either allowed to be forwarded from one interface to another interface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

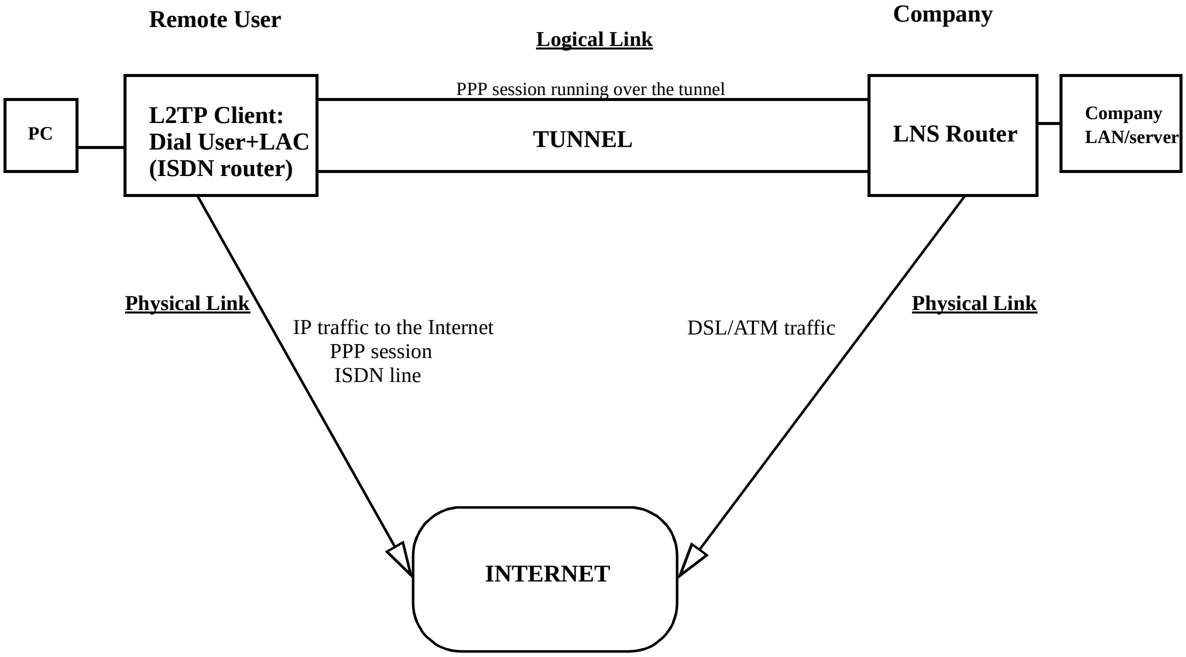
The dial user is one end of a PPP session. The LNS is the other end of the PPP session.

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.

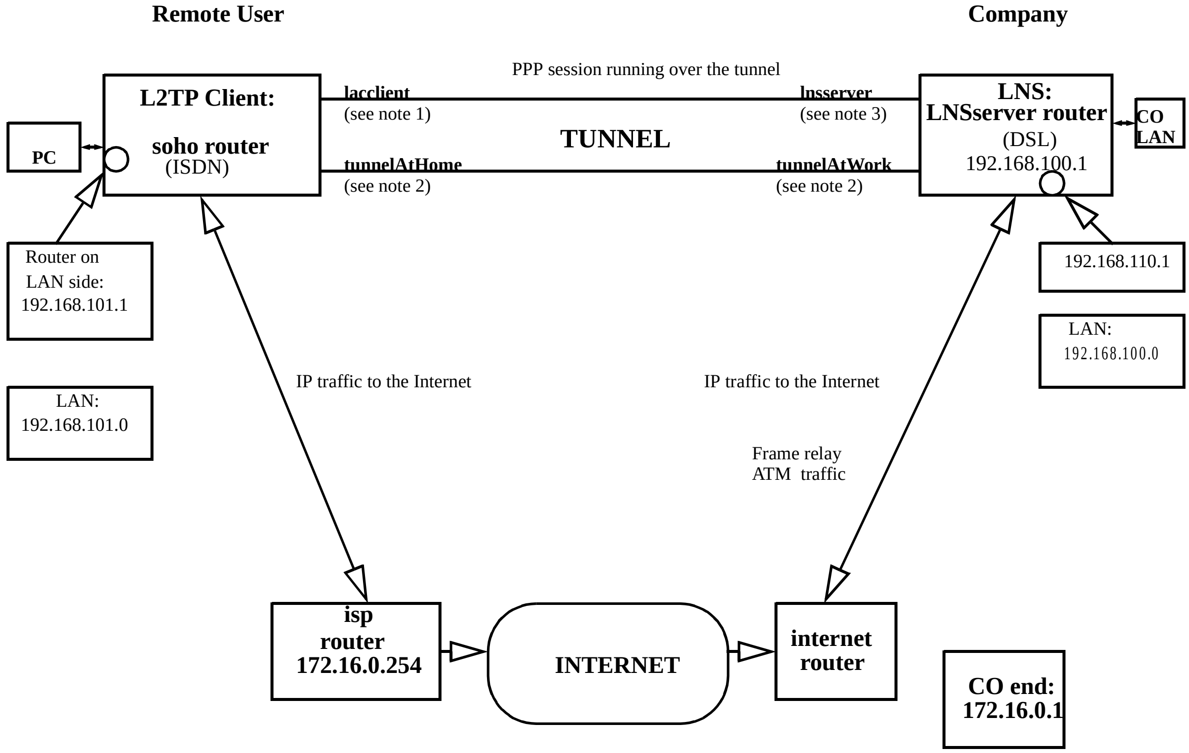
PPP Authentication

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 4. 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
  isdn
  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                  execute
format          sync                    msfs
ifs             ipifs                   iproutes
ipxroutes       ipxsaps                 bi
system          eth                     isdn
save            erase                   remote
call            ping                    pots
dhcp            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
- dial a remote router to test the ISDN line
- 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:`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: Response:

```
Interface   Speed  In%    Out%   Protocol   State     Connection
ETHERNET/0   10mb 0%/0%   0%/0% (Ethernet)  OPENED
ISDN/3      0 b           (VOICE)    CONNECTED pots(1) call to #5553333
ISDN/2       64kb 83%/83% 3%/3% (HDLC/PPP)  OPENED    HQ
ISDN-D/0     16kb 0%/0%   0%/0% (HDLC/LAPD) OPENED
CONSOLE/0  9600 b 0%/0%   0%/0% (TTY)       OPENED
```

In% is an instantaneous sample
Out% is a 5-second average bandwidth utilization
ISDN/2 and 3 refer to the two B-channels
ISDN/2 indicates a data call (protocol=HDLC/PPP) connected to remote router HQ
ISDN/3 is a voice call. For a voice call, the states are those described in the **pots list** command (dialing, ringing, etc.) and the connection indicates the direction and phone number dialed or received.

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: Response:

```
# ps
TID:          NAME              FL MK C P BOTTOM CURRENT SIZE
1:IDLE                          02 00 0 7 8ca200 8ca994 2032
3:MSFS_SYNC                     03 00 0 6 8cc100 8cc868 2032
4:SYSTEM LOGGER                 03 00 0 5 8ccb00 8cd268 2032
5:LL_PPP                        03 00 0 5 8cd400 8cdb68 2032
6:NL_BRIDGE                     03 00 0 5 8cdd00 8ce464 2032
7:NL_IP                         03 00 0 5 8ce700 8cee6c 2032
8:TL_IP_UDP                     03 00 0 3 8cf000 8cf770 2032
9:TL_IP_TCP                     03 00 0 3 8cf900 8d0070 2032
10:IP_RIP                       03 00 0 4 8d0300 8d0a68 2032
11:NL_IPX                       03 00 0 5 8d0c00 8d136c 2032
12:TELNETD                      03 00 0 5 8d1500 8d1c50 2032
13:ISDN_L1                      13 00 0 1 8d5500 8d5c68 2032
```

```
14:ISDN_ME                      13 00 0 3 8d5e00 8d6558 2032
15:ISDN_L2                      13 00 0 2 8d6700 8d6e58 2032
16:ISDN_L3                      13 00 0 3 8d7100 8d7858 2032
17:ISDN_CC                      13 00 0 3 8d7a00 8d8158 2032
18:ISDN_UL                      03 00 0 4 8d8300 8d8a54 2032
19:POTS                         13 00 0 1 8d9a00 8da164 2032
20:DUM                          03 00 0 5 8dc200 8dc968 2032
21:SNMPD                        03 00 0 5 8d3200 8d4170 4080
22:CMD                          01 00 0 6 8dd500 8de3cc 4080
```

```
 where 3: (file system synchronization)
  4: (system logging function)
  5: (PPP lower layer)
  6: (network layer bridging)
  7: (network layer IP routing)
 8: (transport layer IP routing-UDP)
  9: (transport layer IP routing-TCP)
 10: (RIP for IP)
 11: (network layer IPX routing)
 12: (TELNET Daemon)
 13: (ISDN layer 1)
 14: (ISDN management entity)
 15: (ISDN layer 2)
 16: (ISDN layer 3)
 17: (ISDN call control)
 18: (ISDN upper layer application)
 19: (POTS manager)
 20: (dial up manager)
 21: (SNMP daemon)
 22: (command processor)
```

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:

```
.# vers
FlowPoint/128 ISDN Access Node (ISDN v5.2)
FlowPoint-100 BOOT/POST V2.2.0 (04-13-98 16:27)
Software version fp100/fp128 (v3.0.1. built Tue Jan 26 14:43:40 PDT 1999
18:36:15 PST 1999
Maximum users: unlimited
Options: ISDN-EU, NET3, NI1, 5ESS, DMS100, HSD64, HSD128, HSD144, ~POTS, IP,
~IP FILTERING, IP TRANS, HOST MAPPING, DHCP, L2TP, ~ENCRYPT, BRIDGE

Up for 0 days 0 hours 20 minutes (started 1/7/1999 at 13:28)
```

Note: 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.

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
- **isdn:** Target router ISDN commands
- **eth ip**: Ethernet IP routing commands
- **remote**: remote router database commands
- **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
- CallerID feature activation
- Data as voice on inbound calls

SYSTEM ?

Lists the supported keywords.

system ?

Response:

```
System commands:
?                    msg                  name
passwd               authen               community
callerID             list                 admin
DataAsVoice          history              log
addHostMapping       delHostMapping       addServer
delServer            bootpServer          supportTrace
telnetport           snmpport             addTelnetFilter
delTelnetFilter      addSNMPFilter        delSNMPFilter
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 <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 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 <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 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 CALLERID ISDN

Enables or disables CallerID system-wide. Caller ID is an additional data call security feature that allows verification of a remote router’s phone number when the remote router dials in. Phone numbers are entered using the **remote addCaller** command.

system callerID isdn <on|off>

on When CallerID is set on, any calls from phone numbers other than those specified for the remote router, will be rejected.

off The default; no CallerID checking.

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 81.

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 DATAASVOICE

This command causes the router to receive data calls as voice calls. If you use this feature, all incoming voice calls will then be processed as data; i.e., you will not be able to use the POTS interface for incoming voice calls.

Warning: This feature must be used with care. Both ends of the connection must agree to configure calls in this manner and the feature may not work depending on the central office service.

system dataAsVoice <on|off>

on When ‘Data as Voice’ is set on, all incoming voice calls are received as data.

off This is the default; the feature is inactive.

Example: `system dataAsVoice on`

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 3.

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
- isdn list
- pots list (if this is a POTS router)
- bi (if bridging is enabled)
- 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 ISDN Configuration Settings (ISDN)

The following commands allow you to:

- set ISDN SPIDs, directory numbers, and switch type
- set ISDN subaddress and a call delay
- activate and reset the ISDN link
- list the current ISDN settings
- get the status of the ISDN link
- set line speed
- lock out data calls

Refer to chapter 1, *ISDN and Ordering Issues* in the *User Guide* for more information on ISDN.

ISDN ?

Lists the supported keywords.

isdn ?

Response:`Response:`

```
ISDN commands:
?                   help                set
save                list                reset
activate            net3L1test
```

ISDN ACTIVATE

Activates the ISDN line. You do not normally need to enter this command.

isdn activate

Example: `isdn activate`

ISDN LIST

Displays the target router ISDN SPIDs, directory numbers, switch type, and the operational status of the B- and D-channels.

isdn list

Example: `isdn list`

Response:
```
Response:
   DSL 0 is Idle
   Switch type is National ISDN-I
   ISDN Outgoing data calls allowed: yes
   ISDN Incoming data Calls allowed: yes
   Retry failed calls every 30 seconds
         CES: 1: 0555100001/5551000 TEI 76 assigned
         CES: 2: 0555300001/5553000 TEI 77 assigned
   ISDN/2               Idle ces=0 cid=-1 not assigned
  ISDN/3                Idle ces=0 cid=-1 not assigned
```

ISDN RESET

Resets the ISDN software, re-initializing the ISDN connection. Use only when you are experiencing severe problems on the ISDN connection.

isdn reset

Example: `isdn reset`

Response: 12/11/1995-21:39.15: ISDN: SPID/DN Accepted for chan 1
12/11/1995-21:45.14: ISDN: SPID/DN Accepted for chan 2

ISDN SET CALL_DELAY

Sets the time delay after a call made to the network fails, before the target router retries the call.

Note: These settings are saved across reboots.

isdn set call_delay [*number*]

number Number of seconds before retrying a call to the network. A value of zero resets the call delay time to the default value for a particular switch The default is 30 seconds in the U.S. and 90 seconds in Europe and Japan.

Example: `isdn set call_delay 60`

ISDN SET DATACALLSALLOWED

You can decide whether to allow or lock out data calls. This feature is particularly useful if your router is configured to bridge and you want to ensure that no data calls are made or received by your POTS lines.

isdn set DataCallsAllowed in|out|both yes|no

in Selects incoming data calls.

out Selects outgoing data calls.

both Selects both inbound and outgoing data calls.

yes *Yes* to a selected option above will allow it.

no *No* to a selected option above will lock it out.

Example: isdn set DataCallsAllowed in no
(ISDN incoming data calls are locked out)

isdn set DataCallsAllowed out yes
(ISDN outgoing data calls are allowed)

ISDN SET DNS

Sets the target router’s ISDN Directory Numbers provided by the ISDN service provider. DNs can be entered for European and Japanese switches.

isdn set dns [*number*] [*number*]

number Directory number. If no number is entered in the command, the directory numbers are cleared. If one directory number is entered, both numbers are set to the same number.
The incoming call number must be a subset of the entered DN number in order to be accepted by the router, or the call will be ignored.

Example:

```
isdn set dns
isdn set dns 5551111
isdn set dns 5551111 5551112
```

Response*:

```
   ISDN: SPID/DN Accepted for chan 1
   ISDN: SPID/DN Accepted for chan 2
```

* The response indicated may not be displayed immediately. The message appears only when the ISDN line is plugged in and after the new SPIDs and DNs are registered with the network. This message only appears in North America.

ISDN SET SPEED

When this setting is active, the speed of all calls made and received by the router is 56 kilobits per seconds, regardless of the speed setting in the remote database. This feature should only be used where a network, which operates at 56,000 bits per second, actually signals calls at 64,000 bits per second.

isdn set speed [auto|56000]

56000 Locks the speed at 56,000 bits per second

auto Any override of the default line speed is done by the **remote setSpeed** command.

Example:

```
isdn set speed 56000
```

ISDN SET SPIDS

Sets the target router’s ISDN SPIDs, provided by the ISDN service provider.

isdn set spids [*number*] [*number*]

number SPID number. If SPID numbers are not entered in the command, the SPID numbers are cleared. If one SPID number is entered, SPID#1 and SPID#2 are set to the same number.

Example: `isdn set spids 4085551111 4085551112`

Response*:

```
ISDN: SPID/DN Accepted for chan 1
ISDN: SPID/DN Accepted for chan 2
```

The response indicated may not be displayed immediately. The message appears only when the ISDN line is plugged in and after the new SPIDs and DNs are registered with the network. This message is only displayed for North American switches.

ISDN SET SUBADDR

Sets the target router's ISDN subaddress to identify the device to callers. If no address is entered, the subaddress is reset.

isdn set subaddr [u|n <*string*>]

u|n Specify **u** for user defined subaddress. Specify **n** for network service access point format.

string If **u** is specified, the subaddress can be a character string or a series of hexadecimal digits. If **n** is specified, the subaddress can be a string of up to 20 characters or a series of up to 40 digits. If **n** is specified, an even number of digits must be specified. The hexadecimal string must be preceded with a '/'.

Example:

```
isdn set subaddr u address10
isdn set subaddr n /1f2abcd3
isdn set subaddr u /12579a
```

ISDN SET SWITCH

Sets the target router Telco switch type your ISDN service provider is using. If a switch type is not entered in the command, the list of switch types is displayed.

Auto-SPID Detection: The router can be configured to detect the ISDN SPIDs are based on the switch type and the Directory Numbers. It may take up to 2 minutes to detect the SPIDs. If SPID detection fails, SPIDs must be manually configured. If valid SPIDs are found, they are saved to flash automatically.

Note 1: In countries where lease line ISDN is available, HSD64 or HSD128 switches allow the router to work at 64 Kbps or 128 Kbps. In HSD mode, only one remote entry is allowed since it is a permanent link. You must reboot to go into or go out of HSD mode. POTS are not available in this mode.

isdn set switch 5ESS|AUTO-5ESS | DMS100 | AUTO-DMS | NI1 | AUTO-NI1 | KDD | NTT | NET3 | NET3SW | HSD64 | HSD128 | HSD144

switchType Type of Telco switch. Can be one of the following:

5ESS	AT&T 5ESS
AUTO-5ESS	Auto-SPID detection for that switch
DMS100	Northern Telecom DMS-100

AUTO-DMS100	Auto-SPID detection for that switch
NI1	National ISDN NI-1 standard
AUTO-NI1	Auto-SPID detection for that switch
KDD	Kokusai Denshin Denwa., Ltd.
NTT	Nippon Telegraph and Telephone
NET3	European ISDN
NET3SW	NET3 Swiss variant
HSD64	64 Kb permanent connection
HSD128	128 Kb permanent connection
HSD144	144 Kb permanent connection

Example:

```
isdn set switch DMS100
```

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 directedbcast on
```

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 3.*

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 addr 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
```

Target Router Analog Services (POTS)

The following commands allow you to:

- associate phone numbers with POTS interfaces
- set answer and/or dial mode
- set call preemption
- enable or disable POTS interfaces
- list current POTS configuration settings

POTS ?

Lists the supported keywords.

pots ?

```
Response:Response:
POTS sub-commands?
add             del                       disable
enable          list                      set
```

POTS ADD

Associates a phone number with a POTS interface on incoming calls. The router matches the least significant digits of the assigned phone number with the incoming called number. An outgoing call will select any available phone line on which to place the call. This command adds phone numbers cumulatively; to delete any existing phone numbers, use **pots del**.

pots add <*pots#*> <*phone#*>

pots# **1, 2,** or **all** for POTS interfaces 1, 2, or both interfaces, respectively

phone# Phone number associated with the specified POTS interface or with both POTS 1 & 2 if all was specified.

Example:

```
pots add 1 5551212
pots add 2 1212
```

The first example results in POTS interface 1 to be allocated to incoming calls to phone number 5551212. The second example results in incoming calls with the last digits 1212 to be allocated to POTS interface 2.

POTS DEL

Disassociates a phone number from a POTS interface.

pots del <*pots#*>| <*phone#*>

pots# **1, 2,** or **all** for POTS interfaces 1, 2, or both interfaces, respectively.

phone# Phone number associated with the specified POTS interface or with both POTS 1 and 2 if all was specified.

Example: `pots del 1 5551212`

POTS DISABLE

Disables a POTS interface.

pots disable <*pots#*>

pots# **1, 2,** or **all** for POTS interfaces 1, 2, or both interfaces, respectively.

Examples:

```
pots disable 1
pots disable all
```

POTS ENABLE

Enables a POTS interface.

pots enable <*pots#*>

pots# **1, 2,** or **all** for POTS interfaces 1, 2, or both interfaces, respectively.

Example:

```
pots enable 1
pots enable all
```

POTS LIST

Lists the POTS interfaces configuration, including answer/dial mode, call preemption, and telephone number assignments. The status of the POTS interface is also displayed. Status can be out-of-service, available for use, dial-tone, ringing, dialing, not configured for dialing, waiting for dial-tone, connected, and disconnected.

pots list

Example: `pots list`

Response:

```
Response:
pots(1)................ENABLED
          state...............CONNECTED
          answer/dial mode....both
          preempt.............incoming/outgoing
          if preempt, auto....incoming/outgoing
                last call attempt...outgoing
                last incoming call unknown
                last outgoing call unknown
                last local phone number used unknown

pots(2)................ENABLED
          state...............AVAILABLE FOR USE
          answer/dial mode....both
          preempt.............incoming/outgoing
          if preempt, auto....incoming/outgoing
          last call attempt...outgoing
          last incoming call unknown
          last outgoing call unknown
          last local phone number used unknown
```

POTS SET AUTO

Enables automatic bumping of data calls for the router’s POTS interface when answering a call, dialing a call, or both answering and dialing a call on a-per-line basis.

pots set auto <*pots#*> answer|dial|both

pots# **1, 2,** or **all** for POTS interfaces 1, 2, or both interfaces, respectively.

answer Sets a POTS interface to only answer a phone call.

dial Sets a POTS interface to only dial a phone number.

both Sets a POTS interface to both answer and dial.

Example:

```
pots set auto 2 answer
```

POTS SET LINE

Specifies the answering and/or dialing capability for the POTS interfaces.

pots set line <*pots#*> answer|dial|both

pots# **1, 2,** or **all** for POTS interfaces 1, 2, or both interfaces, respectively.

answer Sets a POTS interface to only answer a phone call.

dial Sets a POTS interface to only dial a phone number.

both Sets a POTS interface to both answer and dial.

Example:

```
pots set line 1 answer
pots set line 2 both
```

The first example dedicates POTS interface 1 for answer mode. The second example allows both incoming and outgoing voice calls on POTS interface 2.

POTS SET PREEMPT

Specifies the priority of a voice call over a data call. You can specify that an incoming and/or an outgoing analog call has priority over a data call or that no preemption occurs unless two data channels are in use to the same destination. No preemption ensures that a data connection is maintained on at least one channel.

Note that an incoming voice call can only be received when both data channels are active if the ‘Additional Call Offering’ service is subscribed to through the phone company.

pots set preempt <*pots#*> in|out|both|none

pots# **1, 2,** or **all** for POTS interfaces 1, 2, or both interfaces

In or **both** If **in** or **both** is specified, an inbound voice call will preempt a data call; the data call is preempted onlywhen the voice call is answered.

Out or **both** If **out** or **both** is specified, an outbound voice call will preempt a data call; one data channel is randomly selected to preempt if the two data channels are connected to different destinations.

None If **none** is specified, no calls will preempt data channels unless the two data channels are connected to the same destination.

Examples:

```
pots set preempt 1 both
pots set preempt all out
```

The first example allows call preemption on POTS interface 1 in both inbound and outbound directions. The second example allows call preemption on both POTS interfaces on outgoing calls.

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:

- Phone numbers
- CallerID phone numbers
- Call management
- Bandwidth management
- 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          setPhone            delPhone
setBWThresh         setBod              addCaller
delCaller           setMaxLine          setMinLine
setTimer            setSpeed            delSubAddr
setSubAddr          setDialBack         setPPPCallBack
delPPPCallBack      setDataAsVoice      addHostMapping
delHostMapping      addServer           delServer
setIPTranslate      setCompression      stats
statsclear          setSrcIpAddr        setRmtIpAddr
addIproute          delIproute          setIpOptions
```

```
listIproutes        setIpxaddr          addIpxroute
delIpxroute         listIpxroutes       addIpxsap
delIpxsap           listIpxsaps         addBridge
delBridge           listBridge          enaBridge
disBridge           setBrOptions        setEncryption
delEncryption      setl2tpclient         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 ADDBRIDGE

Adds bridging capability from the target router to the remote router. Defines a default destination for bridging or defines specific remote bridging addresses. When a default destination is specified (and enabled), all outbound bridging is sent to the default destination. When specific remote bridge addresses are defined, the router places these addresses into the static bridging table.

remote addBridge * | <*mac_addr***>**<*remoteName*>

***** Indicates a default destination and all traffic from the local LAN is bridged to this remote router.

mac_addr MAC address specified as xx:xx:xx:xx:xx:xx (6-byte address).

remoteName Name of the remote router (character string)

Examples:

```
remote addBridge * HQ
remote addBridge 01:08:03:0a:0b:0c HQ
```

REMOTE ADDCALLER

Adds acceptable phone numbers from which a remote router can call the local router. If the CallerID feature is enabled using the **system callerID** command, the caller's phone number is verified when a call comes in from the remote router. The call is rejected if the phone number is not in the remote router database.

remote addCaller isdn <*phone#*> <*remoteName*>

phone# Caller's phone number

remoteName Name of the remote router (character string)

Example:

```
remote addCaller isdn 5556666 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 DELBRIDGE

Deletes bridging capability from the target router to the remote router. Specific local addresses or all local addresses can be deleted.

remote delBridge *|<*mac_addr***>** <*remoteName*>

* Deletes the remote router as the default bridging destination

mac_addr MAC address specified as xx:xx:xx:xx:xx:xx (6-byte address). Deletes the MAC address from the bridging table. Dashes can also be used: xx-xx-xx-xx-xx-xx-xx.

remoteName Name of the remote router (character string)

Example:
```
remote delBridge * HQ
```

REMOTE DELCALLER

Deletes phone numbers from which a remote router can call the local router.

remote delCaller isdn <*phone#*|all> <*remoteName*>

phone# Caller’s phone number is deleted. If **all** is specified, all phone numbers are deleted.

remoteName Name of the remote router (character string)

Examples:
```
remote delCaller 5556666 HQ
remote delCaller all 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 DELPHONE

Deletes phone numbers that were established with the command `remote setPhone,` which is used when dialing out to the remote router.

remote delPhone async | isdn <*index*> <*phone#*> <*remoteName*>

index 1 or 2, for the first or second ISDN channel.

phone# Decimal number representing the exact digits to be dialed to access the remote router. Digits, asterisk, and # are accepted.

remoteName Name of the remote router (character string)

```
Examples:   remote delPhone isdn 1 5551111 HQ
            remote delPhone isdn 2 5551112 HQ
```

REMOTE DELPPPCALLBACK

The router will no longer request PPP CallBack with the remote router. PPP CallBack causes the local router to request that the remote router disconnect and call the local router back.

remote delPPPCallBack <*remoteName*>

remoteName Name of the remote router (character string)

```
Example:    remote delPPPCallBack 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 76, 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 DELSUBADDR

Deletes the subaddress for the remote router. The subaddress, passed during call setup, uniquely identifies the remote ISDN device (remote router).

remote delSubAddr <*remoteName*>

remoteName Name of the remote router (character string)

Example: `remote delSubAddr HQ`

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 86.

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)

```
remote list HQ
```

```
INFORMATION FOR <hq>
  Status.............................. enabled
  Our Password used when dialing out... no
  Disconnect timeout (in seconds)...... 60
  Min/max channels.................... 0/2
  Interface in use.................... ISDN
  Authentication...................... disabled
  Authentication level required........ CHAP
  Bandwidth management criteria........ both
  Utilization threshhold.............. 50%
  1. ISDN telephone number, speed auto 5552000
  2. ISDN telephone number, speed auto 5554000
  Dial Back...........................off
  Request PPP Call Back................no
  Place ISDN Data Call as Voice Call....no
  IP address translation.............. off
  Send/Receive Multicast.............. off
  Compression negotiation............. on
  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 0004 1
        Bridging enabled.................... no
          Exchange spanning tree with dest... no
          Mac addresses that dial remote..... none
```

In this example, the target router can connect to the remote router HQ through a maximum of two ISDN B-channels. The security authentication protocol is CHAP. After 120 seconds of inactivity, the ISDN line will be disconnected.

A remote IP network can be accessed through a default route. No bridging is defined for any MAC addresses on the Ethernet LAN and bridging is disabled. A remote IPX network (external) is defined.

If an entry is not in the remote router database for HQ, the response is:

```
Unknown remote name HQ
```

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: Response:

```
PHONE NUMBER(s) FOR <HQ>
1. ISDN telephone number, speed auto 5552000
2. ISDN telephone number, speed auto 5554000
```

Note: If the remote name is not specified, a list of phone numbers is displayed for each remote router in the database.

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 SETBOD

Sets bandwidth on demand management to incoming, outgoing, or both incoming and outgoing traffic. The bandwidth on demand threshold set using the **remote setbwthresh** command applies to the direction of traffic specified by this command.

remote setBod in|out|both <*remoteName*>

in|out|both Used for bandwidth management on incoming, outgoing or both incoming and outgoing traffic. The default is **both**.

remoteName Name of the remote router (character string)

Example: `remote setBod both 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 SETBWTHRESH

Sets the bandwidth threshold used in bandwidth on demand management. Initially a call is activated on one B-channel. When bandwidth utilization reaches the bandwidth, the second B-channel is activated (if maximum links have been set to 2). An additional channel is available for use if the **remote setmaxline** command has been used to set the maximum number of links to 2.

Both channels are utilized until the bandwidth utilization drops below the threshold. The default is 0% utilization which means that both channels are used for data transmission.

remote setBwThresh <*threshold*> <*remoteName*>

threshold Number from 0 to 100 representing the% bandwidth utilization. The default is 0%; meaning the maximum number of links is allocated when data transmission occurs.

remoteName Name of the remote router (character string)

Example:

```
remote setBwThresh 75 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 SETDATAASVOICE

Causes the router to send data calls as voice calls. This may be used to reduce phone charges.
Warning: This feature must be used with care. Both ends of the connection must agree to configure calls in this manner and the feature may not work depending on the central office service.

remote setDataAsVoice [on|off] <*remoteName*>

on Data calls are sent as voice calls to the remote router

off The feature is inactive

remoteName Name of the remote router (character string)

Example:

```
remote setDataAsVoice on HQ
```

REMOTE SETDIALBACK

Controls the Dial-Back feature. Dial-Back causes the router to reject an incoming data call from another router and dial that router back using a configured phone number. This can be used to force phone charge billing to the local router. Dial-Back can be enabled, disabled or enabled such that Dial-Backs occur only if called by the remote router first.

remote setDialBack on | off | only <*remoteName*>

on	Dial-Back feature is on for the remote router
off	Dial-Back feature is off for the remote router
only	Disconnects the caller and calls back only if called first
remoteName	Name of the remote router (character string)

Example:

```
remote setDialBack on HQ
```

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 178, 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 184, for more usage information.

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

REMOTE SETMAXLINE

Sets the maximum number of links to be used during remote data transmission. You can set one or two B-channels for ISDN traffic. If you set the maximum number of links to **2** for ISDN traffic, bandwidth on demand management will determine the usage of B-channels for data transmission.

remote setMaxLine <*maxLine#*> <*remoteName*>

maxLine# Up to **1** or **2** links can be used for data transmission. The default is **1** channel.

remoteName Name of the remote router (character string)

Example: `remote setMaxLine 2 HQ`

REMOTE SETMINLINE

Sets the minimum number of links to be used for remote data transmission. You can specify a number of B-channels (up to the maximum links setting) to be permanently allocated for the remote site connection or specify that a channel is allocated only as required. **0** is the default indicating a channel is allocated when needed; specify **1** or **2**, for permanent allocation of one or two B-channels.

remote setMinLine <*minLine#*> <*remoteName*>

minLine# A minimum of **0, 1,** or **2** B-channels are used for data transmission. The default is **0**.

remoteName Name of the remote router (character string)

Example: `remote setMinLine 1 HQ`

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 SETPHONE

Specifies the phone number to be used when dialing out to the remote router.

remote setPhone async | isdn <*index*> <*phone#*> <*remoteName*>

index 1 or 2, for the first or second ISDN channel.

phone# Decimal number representing the exact digits to be dialed to access the remote router. Digits, asterisk, and # are accepted.

remoteName Name of the remote router (character string)

Examples:

```
remote setPhone isdn 1 5551111 HQ
remote setPhone isdn 2 5551112 HQ
```

REMOTE SETPPPCALLBACK

Causes the local router to request that the remote router disconnect and call the local router back. If accepted, this results in ISDN phone charge billing to the remote router. Any additional information specified must be that which is required by the remote end and should be obtained from a network administrator. If only the remote router name is specified, PPP user authentication is performed for the CallBack action.
Note: Caller ID must be disabled.

remote setPPPCallBack dialnum | E164 | name [number|string] <*remoteName*>

dialnum Local router's phone number to be used by the remote end.

E164 Refers to a phone number in the E164 standard format.

name A character string used by the remote end for locating dialing information.

remoteName Name of the remote router (character string)

Examples:

```
remote setPPPCallBack dialnum 5551111 HQ
remote setPPPCallBack E164 xxxxxx HQ
remote setPPPCallBack name NewYork 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 SETSPEED

Sets the speed of the outgoing call on a per remote basis. for the remote router.

remote setSpeed 56K | 64K | auto isdn <*index*> <*remoteName*>

56K This locks the speed at 56 Kb per second.

64K This locks the speed at 64 Kb per second.

auto The router will attempt to negotiate the speed (56 Kb or 64 Kb) with the remote router. This is the default setting.

index 1 or 2, for the first or second ISDN channel.

remoteName Name of the remote router (character string)

Example:

```
remote setSpeed 56K isdn 2 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 SETSUBADDR

Sets the subaddress for the remote router. The subaddress, passed during call set-up, uniquely identifies the remote ISDN device (remote router).

remote setSubAddr <u|n *subaddr*> <*remoteName*>

u|n **u** indicates user-defined subaddress; **n** indicates network service access point (nsap) international standard format.

subaddr If **u** is specified, the subaddress can be a character string or a series of hexadecimal digits.

If **n** is specified, the subaddress can be a string of up to 20 characters or a series of up to 40 digits. If **n** is specified, an even number of digits must be specified. The hexadecimal string must be preceded with a ‘/’. (BCD is a subset of hexadecimal and can also be specified.)

remoteName Name of the remote router (character string).

Example:

```
remote setsubaddr u HQ1 add HQ
remote setsubaddr n /1f2abcd3 SanFran
remote setsubaddr u /12579a NewYork
```

REMOTE SETTIMER

Sets the timer value used for disconnecting the ISDN dial-up communications line when no traffic is occurring.

remote setTimer <*timerValue*> <*remoteName*>

timerValue Number in seconds representing the time after which the communication line is disconnected (when there has not been any traffic). The time is a decimal number. The default is 60 seconds.

remoteName Name of the remote router (character string)

Example: `remote setTimer 180 HQ`

REMOTE STATS

Shows the current status of the connection to the remote router including the bandwidth, data transfer rate, and call details.

remote stats [<*remoteName*>]

remoteName Name of the remote router (character string)

Example: `remote stats HQ`

Response:
```
Response:
STATISTICS FOR <HQ>:
  Current state...................... currently connected
  Current output bandwidth........... 72000 bps
  Current input bandwidth............ 72000 bps
  Current bandwidth allocated........ 128000 bps
  Current bandwidth state............ increasing
  Current number of channels......... 2
  Total connect time................. 0+00:01:08
  Total bytes out.................... 88643
  Total bytes in..................... 283
  Total calls made................... 1
  Total call retries made............ 1
```
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`

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 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 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 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
- ISDN settings

SAVE ALL

Saves the configuration settings for the system, Ethernet LAN, ISDN 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 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 ISDN

Saves the configuration settings for ISDN into FLASH memory.

save isdn

Example: `save isdn`

SAVE POTS

Saves the configuration settings into FLASH memory.

save pots

Example: `save pots`

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
- ISDN 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, ISDN 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 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 ISDN

Erases the configuration settings for ISDN from FLASH memory.

erase isdn

Example: `erase isdn`

ERASE POTS

Erases the configuration settings for POTS from FLASH memory.

erase pots

Example: `erase pots`

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.fp1
Response:Response:kernel.fp1 deleted.
```

DIR

Displays the directory of the file system. The size of each file is listed (bytes).

dir

```
Example:    dir
Response:   KERNEL   FP1  352852
            SYSTEM   CNF  4864
            ISDN     DAT  244
            DHCP     DAT  1792
            FILTER    DAT  1284
```

The following list explains each file:

KERNEL	Router Software
SYSTEM	System Configuration Settings (Remote router database, system settings, and Ethernet LAN configuration)
ISDN	ISDN Configuration Settings
DHCP	DHCP Configuration Settings
FILTER	Filter database

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 **<u>only</u>** 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: Response:

```
Filesystem 0, size=1536k :
Checking filesystem...
Checking file entries...
ISDN     DAT ...    128 bytes .. ok.
SYSTEM   CNF ...   1792 bytes .. ok.
FILTER   DAT ...      0 bytes .. ok.
KERNEL   FP1 ... 315392 bytes .. ok.
1349 fat(s) used, 0 fat(s) unused, 0 fat(s) unref, 1696 fat(s) free
690688 bytes used by files, 13824 bytes by tables, 868352 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 5. 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 122.

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 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.

Manual Boot Menu

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).

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.FP1 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:

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**[2], **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 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.

[2] To reset any parameter, press **`enter`** when prompted.

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.

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.FP1 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:
   192.168.254.254
     Enter server IP address:
   192.168.254.2
     Enter load address [80100]:
     Enter file name: kernel.fp1
   ```

   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.fp1`
   sync

   where xxx.xxx.xxx.xxx is the TFTP server IP address, SFILENAME is the server filename of the kernel and KERNEL.FP1 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 ISDN 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.FP1` 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.fp1
   sync

   where xxx.xxx.xxx.xxx is the TFTP server IP address, `sfilename` is the server filename of the kernel and **kernel.fp1** 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, ISDN.DAT, DHCP.DAT, and FILTER.DAT in this subdirectory. The files can be empty or not, but should be writeable by everyone.

   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**.

3. To backup a copy of configuration files, enter:

   ```
   copy isdn.dat tftp@xxx.xxx.xxx.xxx:myname/isdn.dat
   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.

   To restore the configuration files, enter:

   ```
   copy tftp@xxx.xxx.xxx.xxx:myname/isdn.dat isdn.dat
   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

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 202. Then, while logged in, enter the following commands:

   ```
   format disk
   save
   copy tftp@xxx.xxx.xxx.xxx:kernel.fp1 kernel.fp1
   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 200*.

   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.fp1 kernel.fp1

      This assumes that TCP/IP routing is enabled and that an IP address has been assigned to the Ethernet interface.

Recovering Passwords and IP Addresses

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**.**

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 6. Troubleshooting Software Problems

Software problems usually occur when the router's software configuration contains incomplete or incorrect information. This chapter lists symptoms of software configuration problems, recommends actions for you to take and also lists system messages. It also describes *History Log*, a convenient troubleshooting tool. If you suspect a hardware configuration problem, refer to *Troubleshooting* in the *User Guide*.

Problems and Recommendations

Login Password is Invalid

You have entered the **login passwd** command and received an error message:
"*Re-enter the correct password and hit enter*."

If you have forgotten the password, you must reset the login password. Refer to the *Changing Configuration Switches* Appendix in the *User Guide* and perform the following procedure:

1. Move **switches 5** and **6 DOWN**.

2. Re-enter the **login passwd** command with any password. Password checking is overridden.

3. Move **switch 5** and **6.**

4. Complete any configuration update.

5. Set your login password to a new password using the **system admin** command.

6. Save 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 cycle power on the rear panel of the router.

IP Routing Problems

- Check that Ethernet LAN TCP/IP Routing has been enabled (**eth list** command).
- 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, if required, the source and remote WAN IP addresses are on the same subnetwork.
- 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 a proper Ethernet cable (crossover or straight through).
- 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).

ISDN Problems

ISDN Configuration

1. Connect the ISDN cable from the wall jack to the ISDN U interface.
2. Configure the ISDN Switch Type, the SPIDS, and the Directory Numbers (DNs) as below:

ISDN Switch Type	SPIDs	DNs
DMS-100	**2**	**2**
NI-1	**2**	**2**
AT&T 5ESS **Point-to-Multipoint** **Point-to-Point**	 **2** **0**	 **2** **0**

ISDN Commands

isdn set switch <*switch type*>
isdn set spids [*spid#1*] [*spid#2*]
isdn list
isdn reset
isdn set debug 3
ifs

Ready State

When in ready state, the lights on the router should be as follows:

PWR = ON

LINE = ON

NT1 = ON

And you should get a similar message:

```
10/28/1996-13:27:33:ISDN: SPID/DN Accepted for chan 1
10/28/1996-13:27:33:ISDN: SPID/DN Accepted for chan 1
```

Problems and Troubleshooting

Lights are not in ready state

We assume that the power light is on. If it is not, turn the router ON.

❖ *NT1 light is flashing*

- This flashing light means that there is a problem with the physical connection from the telephone company to the router. Check the connection from the wall to the router’s U interface.
- You may have the wrong cable**.** Check that you have an ISDN cable where only the center 2 wires are used; this is what is required to connect to the U interface. Note that pins #4 and #5 (center pins) of the RJ45 connector are the only 2 pins out of the eight that are used.
- If you are using an external NT1, check the two following cables:

  The cable from the wall to the NT1 (U interface) uses the center two wires (as described above).

  The cable from the NT1 to the router (S/T interface) uses the center 4 wires (pins 3, 4, 5, and 6).
- Also make sure that:
  - you are using straight-through cables, **not** crossover cables.
  - the necessary pins are wired.
  - the connectors are functioning properly.

❖ *NT1 light is solid and the LINE line is flashing*

- Try to reset the ISDN interface by using the command **isdn reset.**
- Use the command **isdn list** to check that the Switch Type, SPIDs, and DNs are correct.
- If the Switch Type is AT&T 5ESS with only 1 DN and 1 SPID.

  Do **NOT** enter them into the router. You are dealing with a Point-to-Point configuration that does not require any SPIDs and DNs.

  Note: For all other configurations, 2 SPIDs and 2 DNs will be needed.
- Try to change the Switch Type to NI1, using the following commands:

  ```
  isdn set switch ni1
  isdn reset
  ```

The lights are in ready state but you did not get the “spids accepted” message.

- If you are using an AT&T 5EES switch in a Point-to-Point configuration, you will **not** get this “SPIDs accepted” message.
- Check the status of the ISDN interface using the **ifs** command. The status should reflect STANDBY mode for both ISDN/2 and ISDN/3.

Your router is in “ready” state but cannot connect to the host site

❖ ***Cause value indicated on the error message:***

1. Check the reference in the “isdn Q.931 cause values” pages.
2. Use the **dod dial** <*isdn number*> **2** command to test the ISDN number.
3. Try dialing your other B channel number . If this does not work, contact the phone company.
4. If it is a local call, use the 7-digit phone number.
5. If this does not work either, try the 11-digit number (1 + area code + number).
6. If this does not work either, try forcing the speed to **56 Kb** and try both #1 and #2 again. Use the command: **isdn set speed 56000**. You can set the speed back to default with **isdn set speed auto.**
7. Try dialing a 9 or 8 if the ISDN is on a Centrex.
8. Try the AT&T access code 10288 in the following format:
   10288 (area code) (phone #)

❖ ***Use the following commands to further test ISDN, if required:***

• **isdn list**	It will show the ISDN configuration of the router.
• **isdn reset**	It will reset the ISDN interface of the router.
• **isdn set debug 3**	It will turn on a level-3 trace of the call setup information.
• **dod dial** <phone # > **2**	The router dials the phone number specified on the B1 channel (the 2 on the end of the command represents channel B1, a 3 represents channel B2).

If no cause value is indicated, then check for authentication problems.

Bridging Problems

- 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).
- Check that the bridged MAC addresses are set to “**all**” (use the **remote listbridge** command).

- Check that ISDN SPIDs are accepted and in “standby” mode with the **ifs** command.
- Verify that the authentication passwords are correct.
- Reboot your PC if you have Windows for Workgroups.
- 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.

IPX Routing Problems

- 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 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 controls have 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.
- If IPX WAN cannot fill the 128 Kb ISDN bandwidth:

  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.

No Dial Tone

- Check the line status for “standby” with the **ifs** command.
- Check that the ISDN line is communicating with the central office. Verify that the ISDN SPIDS (if required) and the central office switch are configured correctly. Check wiring on the ISDN line.
- Check analog telephone device wiring.
- Check that you have configured the correct phone numbers for the POTS interface, the mode is dial and the interface is enabled.
- Check that call preemption is enabled and/or one channel is available or two data channels go to the same destination.
- Check that voice service has been requested from the ISDN service provider.
- Check that the second POTS interface is not initiating a call simultaneously.
- Check that the router has been powered on.

Remote Router Won’t Dial

- Verify that the remote router has a default IP router or a default bridge.
- Check that the remote router is not disabled or in Dial Back mode only, and that it has a phone number.
- Check that it is in ISDN “standby” mode.

Cannot Receive Analog Calls

- Check that the ISDN line is communicating with the central office and correctly configured. Check the wiring on the ISDN line.
- Check the analog telephone device wiring; also check that the ringer is set on.
- Be sure that you have configured the correct phone numbers for the POTS interface, the mode is answer and the interface is enabled.
- Check that Additional Call Offering has been subscribed to from the ISDN service provider and that voice channels have been requested.
- Check if call preemption is disabled and two channels are in use to different destinations.

"Strange" Dial Tone

- You may be in “Set IP Address” mode. You need to hang up and try again.
- You may be in call preemption manual mode. Press * or # to get a dial tone.

Cannot Access the Router via Telnet

- Ensure that the router has a valid IP address.
- Check that the Ethernet cable is plugged in.
- Reboot your PC.

Cannot Download 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.
- Use the console and “ping” the TFTP server address.

How to Obtain Technical Support

Before you contact Technical Support, please have the following information ready:

- Router model number
- Router serial number
- Router software version
- Date of purchase
- Type of Operating System (Windows 95, Windows 98, Windows for Workgroups, or Windows NT)
- Description of the problem
- List of other equipment such as personal computers, modems, etc. and third party software you are using, including revision levels.
- Use the **system supporttrace** command. This command output is used by Technical Support to diagnose problems.

Note: If you can connect with the GUI, gather the Technical Support information from the router with the **Help** and **Technical Support Data Menu** commands.

.Technical support, repair services, and spare parts are available through your FlowPoint Distributor. Otherwise,

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

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.

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 issuing the message including:

- DOD Dial-on-Demand software
- ISDN ISDN link layer software
- PPP PPP software

message message

The following are some examples of the messages:

```
10/31/95+11:24:30.93: DOD: connecting to rt1 @ 5551110 over ISDN/2
10/31/95+12:24:31.07: ISDN: call 32774 proceeding on B1
10/31/95+12:24:34.20: DOD: link to rt1 over ISDN/2 is now up
10/31/95+13:12:50.51: DOD: closed connection(s) to rt1
10/31/95+13:38:29.34: PPP: call from rt1 accepted via CHAP on ISDN/2
10/31/95+13:48:01.23: ISDN: Call disconnected (cause 16)
```

Time-Stamped Messages

Authorization failed

Explanation: PAP authentication cannot be negotiated**.**

BNCP not being negotiated right now

Explanation: Bridging is off on the other end of the WAN link.

Call disconnected (cause {ISDN cause number})

Explanation: The call has been disconnected. Refer to the ISDN cause number and description in the section, *ISDN Q.931 Cause Values*.

Call from {router/user} accepted via [PAP|CHAP] on {link/number}

Explanation: A call from the specified remote destination has been accepted on the indicated ISDN channel after successful PAP|CHAP authentication.

Call {link/number} proceeding (unknown B-channel)

Explanation: The call is proceeding on an unidentified channel.

Call {phone number} proceeding on B- {channel number} channel

Explanation: The call is proceeding on the specified B-channel.

Call to {phone number] {router/user} never completed (no answer)

Explanation: The remote destination did not answer the call.

Call to {router/user} on {link/number} failed

Explanation: The call to the specified remote destination has failed.

Calling remote {name} back

Explanation: Informative message.

Cannot agree with {remote name} 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.

Cannot obtain an IP address from {remote router}: one is needed in single user mode

Explanation: Informative message

Cannot complete call to {phone number} {router/user} at {link speed} (cause {ISDN cause number})

Explanation: The call cannot be completed. Refer to the ISDN cause number and description in the section, *ISDN Q.931 Cause Values on page 221*.

Cannot supply an IP address to {remote router}

Explanation: The remote end requests an IP address from the local end, which cannot supply it.

Closed all connections to {router/user}

Explanation: The last link is down to the specified remote destination.

Closed connection(s) to {router/user}

Explanation: The connection to the specified remote destination has been closed.

Connected to {remote} with no authentication

Explanation: Informative message.

Connecting to {router/user} @ {phone number} over {link/number}

Explanation: Trying to connect to the specified remote destination.

Did not call remote {name} back - Security problem

Explanation: The information provided by the remote router to be called back is incompatible with the actual entries in the local router's database.

Having trouble negotiating with network

Explanation: System is not able to communicate with the ISDN switch.

Ignoring incoming call - can't understand call type

Explanation: Received a voice call and this router model does not support POTS.

Ignoring incoming call {phone number} (no callerID found)

Explanation: Received a voice call while callerID feature is active and this phone number is not in the remote router database.

IP is not configured to operate in unnumbered mode with {name of router}

Explanation: On one end, remote entries have been configured for numbered mode. On the other end, remote entries have been configured for unnumbered mode. Both end cannot communicate with each other.

IPCP not being negotiated right now

Explanation: IP routing is off on the local end of the WAN link.

PXCP not being negotiated right now

Explanation: IPX routing is off on the local end of the WAN link.

Link on {target name} to {remote name} did not negotiate required options

Explanation: Each end is using network control protocols that are disabled at the other end.

Link to {router/user} over {link/number} is now up

Explanation: The connection to the specified remote destination is now up.

Network connection broken

Explanation: The ISDN connection to the switch has been broken. Check the ISDN cable.

Network connection up

Explanation: The ISDN network connection is in idle state.

No bearer info!

Explanation: The ISDN call did not specify the bearer type.

No free channels for incoming call

Explanation: The ISDN channels of the router’s WAN link are in use and unavailable for an incoming call

No free outgoing lines to connect to {router/user}

Explanation: No channels are available to perform a call request to the specified remote destination.

No system name is known for us - using defaults

Explanation: The router does not have a system name. For PAP/CHAP security negotiation, the router will use a default name and password.

Peer not negotiating CCP right now

Explanation: The remote has compression disabled- Informative message.
Non fatal - Link will work but compression is off.

Peer not negotiating - IPCP right now

Explanation: The remote router has IP disabled- Informative message.
Can be fatal if IP routing is required.

Peer not negotiating IPXCP right now

Explanation: IPX routing cannot be negotiated. IPX routing may not be enabled on the remote router, the remote router may have no route back to the local router, or the WAN network numbers have not been specified.

POTS call rejected

Explanation: The voice call is rejected because the POTS interface is disabled or a line is unavailable (not configured for preemption).

POTS line connected

Explanation: A voice call is connected; both ends of the voice call are off-hook.

POTS line disconnected

Explanation: A voice call has been disconnected. The analog equipment has hung up or receiver is on-hook.

Rejecting call - cannot understand call type

Explanation: The ISDN call has been rejected because the call type requested from the remote destination is unknown.

Remote did not authenticate in time

Explanation: The response to authentication did not happen within 30 seconds.

Remote did not negotiate our IP address correctly

Explanation: The remote router did not negotiate the IP address options as was expected by the local router.

Remote on {router/user} {link/number} refuses to authenticate us

Explanation: The remote destination refuses to participate in the PAP/CHAP security authentication process.

Remote on {router/user} {link/number} rejected our password with PAP

Explanation: The remote destination rejected our PAP password during the PAP security authentication process. “**dial authentication passwd**” was not accepted.

Remote {router/user} did not accept our CHAP password

Explanation: The router attempted CHAP security authentication but the remote destination rejected the password. “**dial authentication passwd**” was not accepted.

Remote {router/user} tried to use PAP when CHAP was expected.

Explanation: The remote destination negotiated PAP while its minimum security level in the remote database was set to CHAP.

Remote {router/user} used wrong password [PAP|CHAP]

Explanation: The remote destination has used an invalid password during PAP|CHAP security authentication.

Resetting ISDN interface

Explanation: Reinitializing the ISDN interface due to error.

Retrying call to {phone number} {router/user} at {link speed}

Explanation: The call is being retried to the remote destination.

SPID/DNS accepted for channel {link/number}

Explanation: The SPID and/or DN numbers have been accepted during the call process for the channel specified.

TELNETD: connection accepted

Explanation: A remote configuration session has been established.

TELNETD: connection disconnected

Explanation: A remote configuration session has been disconnected.

Unsupported information transfer capability

Explanation: Received a voice call and this router model does not support POTS.

User {router/user} is disabled in remote database

Explanation: Received a call from a disabled router.

User {router/user} not found in remote database [PAP|CHAP]

Explanation: The remote destination was not found in the remote router database for the PAP|CHAP security authentication process.

Voice call {phone number} proceeding on B {channel number} channel-call in progress

Explanation: The voice call is proceeding on the specified B-channel. The phone number is the actual digits entered on an outbound call or any received digits on an inbound call.

We dialed {router/user} and got {router/user} according to [PAP|CHAP]

Explanation: During the PAP|CHAP security authentication process, a call to one remote destination resulted in a response from another remote destination.

ISDN Q.931 Cause Values

ISDN link level error messages include the Q.931 cause value. The cause value displayed is the cause number exactly or the number +128. The following table is a reference list of the Q.931 cause values.

Cause No.	Cause Name
1	Unassigned (unallocated) number
2	No route to specified transit network
3	No route to destination
6	Channel unacceptable
7	Call awarded and being delivered in an established channel
16	Normal call clearing
17	User busy
18	No user responding
19	User alerting, no answer
21	Call rejected
22	Number changed
26	Non-selected user clearing
27	Destination out of order
28	Invalid number format (incomplete number)
29	Facility rejected
30	Response to STATUS INQUIRY
31	Normal, unspecified
34	No circuit/channel available
38	Network out-of-order
41	Temporary failure
42	Switching equipment congestion
43	Access information discarded
44	Requested circuit/channel not available
47	Resource unavailable, unspecified
49	Quality of service unavailable
50	Requested facility not subscribed

57	Bearer capability not authorized
58	Bearer capability not presently available
63	Service or option not available, unspecified
65	Bearer capability not implemented
66	Channel type not implemented
69	Requested facility not implemented
70	Only restricted digital information bearer capability is available
79	Service or option not implemented, unspecified
81	Invalid call reference value
82	Identified channel does not exist
83	A suspended call exists, but this call identity does not
84	Call identify in use
85	No call suspended
86	Call having the requested call identity has been cleared
88	Incompatible destination
91	Invalid transit network selection
95	Invalid message, unspecified
96	Mandatory information element is missing
97	Message type non-existent or not implemented
98	Message not compatible with call state or message type non-existent or not implemented
99	Information element non-existent or not implemented
100	Invalid information element contents
101	Message not compatible with call state
102	Recovery on timer expiration
111	Protocol error, unspecified
127	Internetworking, unspecified

History Log

History Log is a troubleshooting tool which displays the router’s recent activity. It can be accessed from a terminal emulation session (including 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 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. Use the command **system history** to view the buffer contents.

Other logging commands:

- If you wish to monitor your router activity at all time, enter the command **system log start** to view a continuous log of new events**.** 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.**
- The command **system supporttrace** will provide status and configuration information for Technical Support.

When you exit Telnet, you automatically stop any logging programs running in that session.

Note: The History Log is preserved across reboots but not across power outages or power down.

Appendix A. Network Information Worksheets

TARGET ROUTER:		
Command	**Item**	**Setting**
system name	Router Name (Req)	
system msg	Message	
system authen	Dial Authentication Protocol forced PAP/CHAP/NONE	
system passwd	Dial Authentication Password/ Secret(Req)	
system callerid	CallerID Feature ON/OFF	
dhcp set valueoption domainname **dhcp set valueoption omainnameserver** **dhcp set valueoption winsserver**	DNS Domain Name DNS Server WINS Server address	
isdn set spids	ISDN SPID#1 ISDN SPID#2	
isdn set dns	ISDN Directory Number #1 ISDN Directory Number #2	
isdn set switch	ISDN Switch Type (Req)	
eth ip addr	Ethernet IP Address and Subnet Mask	
eth ip ena/dis	TCP/IP Routing On/Off	
eth ipx addr	Ethernet IPX Address	
eth ipx ena/dis	NetWare IPX Routing On/OFF	

Note: Underlined words are defaults.

REMOTE ROUTER:

Command	Item	Setting
remote setPhone	ISDN Phone #1 ISDN Phone #2	
remote setMaxLine	Max/Min Links (1/0)	
remote setAuthen **remote setPasswd** **remote setOurSysName** **remote setOurPasswd** **remote addCallerID**	Minimum Authentication CHAP|PAP|NONE Remote’s Password System Name Override System Password Override CallerID Phone Numbers	
remote setDialBack **remote setPPPCallBack**	Dial-Back (On/Off/Only) PPP CallBack (On/Off)	
remote addIpRoute	Remote Network’s IP Addresses, Subnet Masks, and Metrics	
setIpOptions	IP RIP protocol options	
setSrcIpAddr **setRmtIpAddr**	Remote WAN IP Address and Subnet Mask[a] Source WAN IP Address and Subnet Mask[b]	
addIpxRoute	IPX Routes: Network Number, Hop Count, Ticks	
addIpxSap	IPX SAPs: Server Name, Network#, Node#, Socket# Server Type and Hop Count	
setIpxAddr	Remote WAN IPX addr	
addBridge	Default Bridging Dest (*) Remote MAC address(es)	
ena/disBridge	Bridging (On/Off)	
setBrOptions	Spanning Tree Protocol (On|Off)	

[a] PPP addressing Numbered Mode only
[b] PPP addressing Numbered Mode only

Note: One chart for each remote router in the remote router database

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:

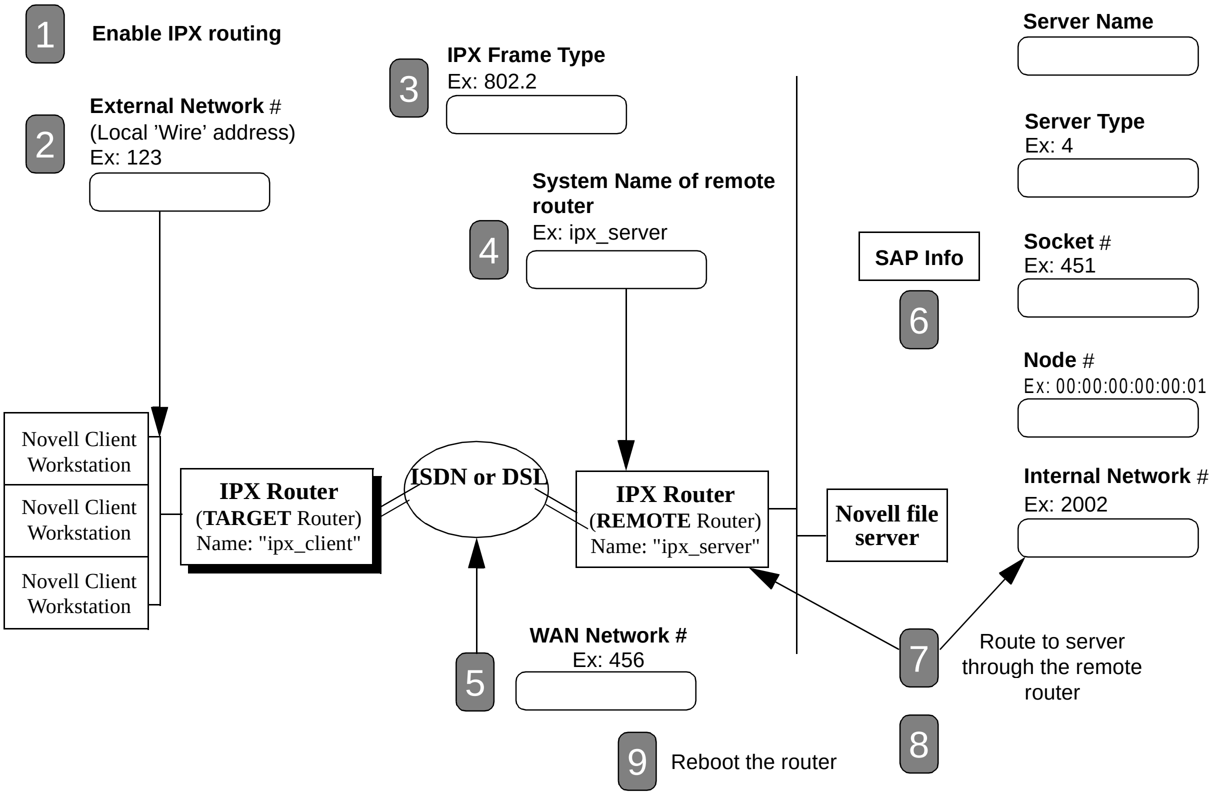
- PPP Authentication
- IPX routing (this section)

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

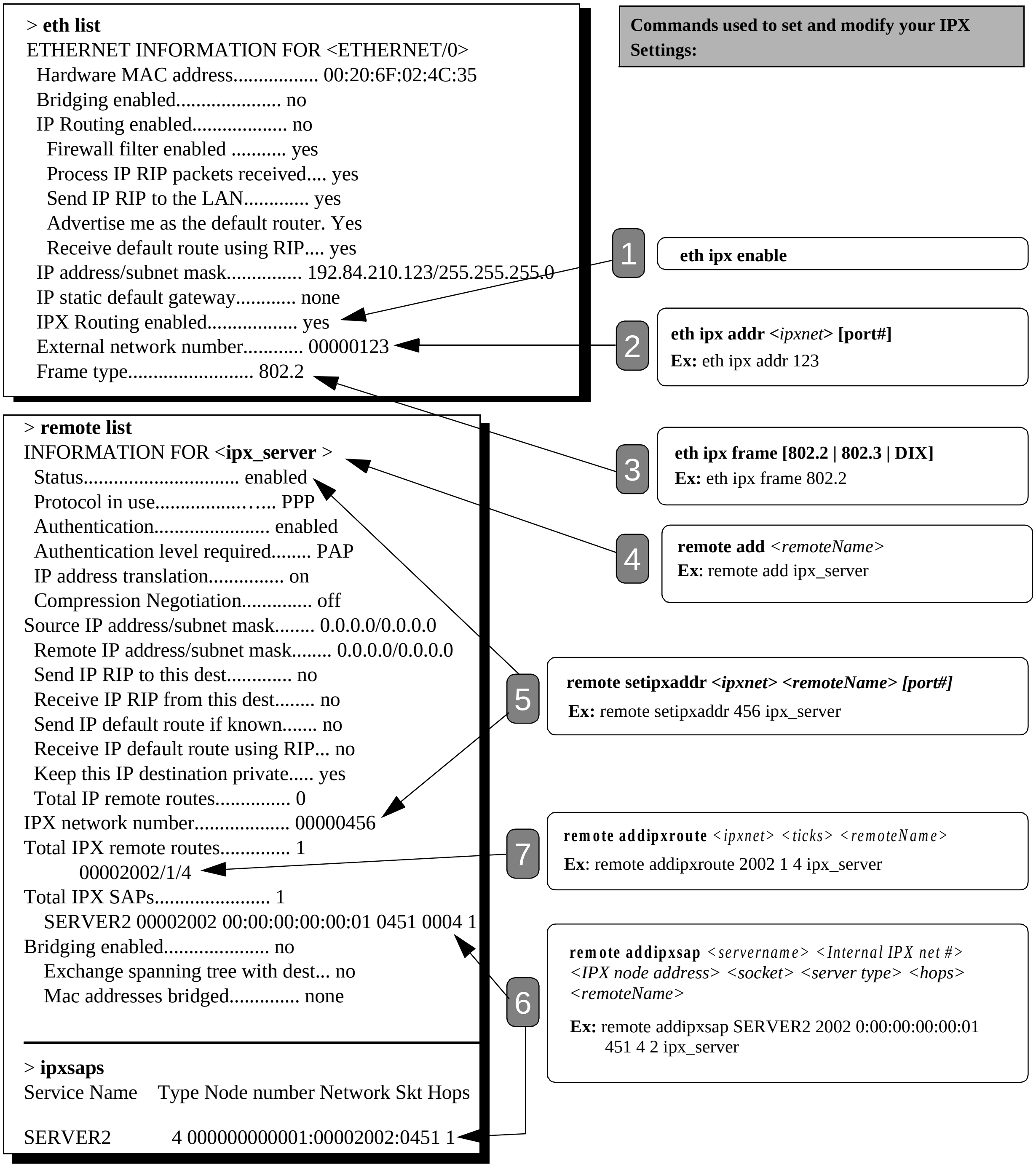
These commands are used to configure the Target (client-side) router (**ipx_client**).Log in with the password **admin.**

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**



```
> eth list
ETHERNET INFORMATION FOR <ETHERNET/0>
  Hardware MAC address................ 00:20:6F:02:4C:35
  Bridging enabled..................... no
  IP Routing enabled................... no
    Firewall filter enabled ........... yes
    Process IP RIP packets received.... yes
    Send IP RIP to the LAN............. yes
    Advertise me as the default router. Yes
    Receive default route using RIP.... yes
  IP address/subnet mask.............. 192.84.210.123/255.255.255.0
  IP static default gateway............ none
  IPX Routing enabled.................. yes
  External network number............ 00000123
  Frame type.......................... 802.2
```

```
> remote list
INFORMATION FOR <ipx_server >
  Status............................... enabled
  Protocol in use.................….. PPP
  Authentication....................... enabled
  Authentication level required........ PAP
  IP address translation............... on
  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
  Receive IP RIP from this dest........ no
  Send IP default route if known....... no
  Receive IP default route using RIP... no
  Keep this IP destination private..... yes
  Total IP remote routes............... 0
IPX network number................... 00000456
Total IPX remote routes.............. 1
          00002002/1/4
Total IPX SAPs....................... 1
    SERVER2 00002002 00:00:00:00:00:01 0451 0004 1
Bridging enabled..................... no
    Exchange spanning tree with dest... no
    Mac addresses bridged.............. none
```

```
> ipxsaps
Service Name    Type Node number Network Skt Hops

SERVER2          4 000000000001:00002002:0451 1
```

Commands used to set and modify your IPX Settings:

1 **eth ipx enable**

2 **eth ipx addr <*ipxnet*> [port#]**
Ex: eth ipx addr 123

3 **eth ipx frame [802.2 | 802.3 | DIX]**
Ex: eth ipx frame 802.2

4 **remote add** <*remoteName*>
Ex: remote add ipx_server

5 **remote setipxaddr <*ipxnet*> <*remoteName*> *[port#]***
Ex: remote setipxaddr 456 ipx_server

7 **remote addipxroute** <*ipxnet*> <*ticks*> <*remoteName*>
Ex: remote addipxroute 2002 1 4 ipx_server

6 **remote addipxsap** <*servername*> <*Internal IPX net #*> <*IPX node address*> <*socket*> <*server type*> <*hops*> <*remoteName*>
Ex: remote addipxsap SERVER2 2002 0:00:00:00:00:01 451 4 2 ipx_server

Index